RESEARCH CENTER

FIELD
Algorithmics, Programming, Software and Architecture

Activity Report 2014

Section New Results

Edition: 2015-03-24

ARIC Project-Team

6. New Results

6.1. Arithmetic operators

6.1.1. A table-based method to evaluate trigonometric functions

Linear (order-one) function evaluation schemes, such as bipartite and multipartite tables, are usually effective for low precision approximations. For high output precision, the lookup table size is often too large for practical use. Dong Wang and Milos Ercegovac (UC Los Angeles) and Nicolas Brisebarre and Jean-Michel Muller investigate the so-called (M, p, k) scheme that reduces the range of input argument to a very small interval so that trigonometric functions can be approximated with very small lookup tables and a few additions/subtractions. An optimized hardware architecture is proposed and implemented in both FPGA device and standard cell based technology. Experimental results show that the proposed scheme achieves more than 50% reduction in total chip area compared with the best linear approach for 24-bit evaluation [14].

6.2. Floating-Point arithmetic

6.2.1. On the computation of the reciprocal of floating point expansions using an adapted Newton-Raphson iteration

Many numerical problems require a higher computing precision than that offered by common floating point (FP) formats. One common way of extending the precision is to represent numbers in a *multiple component* format. With so-called *floating point expansions*, numbers are represented as the unevaluated sum of standard machine precision FP numbers. This format offers the simplicity of using directly available and highly optimized FP operations and is used by multiple-precisions libraries such as Bailey's 'Q'D or the analogue Graphics Processing Units tuned version, GQD. Mioara Joldes (LAAS), Jean-Michel Muller, and Valentina Popescu introduced a new algorithm for computing the reciprocal FP expansion a^{-1} of a FP expansion a. Their algorithm is based on using an adapted Newton-Raphson iteration where "truncated" operations (additions, multiplications) involving FP expansions are used. The error analysis given shows that their algorithm allows for computations of very accurate quotients. Precisely, after $i \geq 0$ iterations, the computed FP expansion $x = x_0 + \cdots + x_{2^i-1}$ satisfies the relative error bound $|\frac{x-a^{-1}}{a^{-1}}| \leq 2^{-2^i(p-3)-1}$, where $p > 4$ is the precision of the FP representation used ($p = 24$ for single precision and $p = 53$ for double precision) [19].

6.2.2. Error bounds on complex floating-point multiplication with a fused-multiply add

The accuracy analysis of complex floating-point multiplication done by Brent, Percival, and Zimmermann [*Math. Comp.*, 76:1469–1481, 2007] is extended by Peter Kornerup (Odense Univ. Denmark), Claude-Pierre Jeannerod, Nicolas Louvet, and Jean-Michel Muller [42] to the case where a fused multiply-add (FMA) operation is available. Considering floating-point arithmetic with rounding to nearest and unit roundoff u, they show that the bound $\sqrt{5}\,u$ on the normwise relative error $|\widehat{z}/z - 1|$ of a complex product z can be decreased further to $2u$ when using the FMA in the most naive way. Furthermore, they prove that the term $2u$ is asymptotically optimal not only for this naive FMA-based algorithm, but also for two other algorithms, which use the FMA operation as an efficient way of implementing rounding error compensation. Thus, although highly accurate in the componentwise sense, these two compensated algorithms bring no improvement to the normwise accuracy $2u$ already achieved using the FMA naively. Asymptotic optimality is established for each algorithm thanks to the explicit construction of floating-point inputs for which it is proven that the normwise relative error then generated satisfies $|\widehat{z}/z - 1| \to 2u$ as $u \to 0$. All these results hold for IEEE floating-point arithmetic, with radix $\beta \geq 2$, precision $p \geq 2$, and rounding to nearest; it is only assumed that underflows and overflows do not occur and, when bounding errors from below, that $\beta^{p-1} \geq 12$.

6.2.3. Refined error analysis of the Cornea-Harrison-Tang method for $ab + cd$

In their book *Scientific Computing on Itanium-based Systems*, Cornea, Harrison, and Tang introduced an accurate algorithm for evaluating expressions of the form $ab + cd$ in binary floating-point arithmetic, assuming a fused-multiply add instruction is available. They showed that if p is the precision of the floating-point format and if $u = 2^{-p}$, the relative error of the result is of order u. Jean-Michel Muller improved their proof to show that the relative error is bounded by $2u + 7u^2 + 6u^3$. Furthermore, by building an example for which the relative error is asymptotically (as $p \to \infty$ or, equivalently, as $u \to 0$) equivalent to $2u$, he proved that this error bound is asymptotically optimal [11]. Claude-Pierre Jeannerod then showed in [41] that an error bound of the form $2u + 2u^2 + O(u^3)$ in fact holds for any radix $\beta \geq 2$, with $u = \frac{1}{2}\beta^{1-p}$. He also showed that the possibility of removing the $O(u^2)$ term from this bound depends on the radix parity and the tie-breaking strategy used for rounding: if β is odd or rounding is *to nearest even* then the simpler bound $2u$ is obtained, while if β is even and rounding is *to nearest away*, then there exist floating-point inputs a, b, c, d that lead to a relative error larger than $2u + \frac{1}{\beta}u^2$.

6.2.4. On the maximum relative error when computing integer powers by iterated multiplications in floating-point arithmetic

Stef Graillat (Paris 6 University), Vincent Lefèvre and Jean-Michel Muller improved the usual relative error bound for the computation of x^n through iterated multiplications by x in binary floating-point arithmetic. The obtained error bound is only slightly better than the usual one, but it is simpler. They also discussed the more general problem of computing the product of n terms [7].

6.2.5. Improved error bounds for numerical linear algebra

When computing matrix factorizations and solving linear systems in floating-point arithmetic, classical rounding error analyses provide backward error bounds whose leading terms have the form $\gamma_n = nu/(1 - nu)$ for suitable values of n and with u the unit roundoff. With Siegfried M. Rump (Hamburg University of Technology), Claude-Pierre Jeannerod showed in [13] that for LU and Cholesky factorizations as well as for triangular system solving, γ_n can be replaced by the $O(u^2)$-free and unconditional constant nu. To get these new bounds the main ingredient is a general framework for bounding expressions of the form $|\rho - s|$, where s is the exact sum of a floating-point number and $n - 1$ real numbers, and where ρ is a real number approximating the computed sum $\widehat{s}$.

6.2.6. On relative errors of floating-point operations

Rounding error analyses of numerical algorithms are most often carried out via repeated applications of the so-called standard models of floating-point arithmetic. Given a round-to-nearest function RN and barring underflow and overflow, such models bound the relative errors $E_1(t) = |t - \mathrm{RN}(t)|/|t|$ and $E_2(t) = |t - \mathrm{RN}(t)|/|\mathrm{RN}(t)|$ by the unit roundoff u. In [34] Claude-Pierre Jeannerod and Siegfried M. Rump (Hamburg University of Technology) investigated the possibility of refining these bounds, both in the case of an arbitrary real t and in the case where t is the exact result of an arithmetic operation on some floating-point numbers. They provided explicit and attainable bounds on $E_1(t)$, which are all less than or equal to $u/(1 + u)$ and, therefore, smaller than u. For $E_2(t)$ the bound u is attainable whenever $t = x \pm y$ or $t = xy$ or, in base > 2, $t = x/y$ with x, y two floating-point numbers. However, for division in base 2 as well as for square root, smaller bounds are derived, which are also shown to be attainable. This set of sharp bounds was then applied to the rounding error analysis of various numerical algorithms: in all cases, they obtained either much shorter proofs of the best-known error bounds for such algorithms, or improvements on these bounds themselves.

6.2.7. Comparison between binary and decimal floating-point numbers

In collaboration with Christoph Lauter and Marc Mezzarobba (LIP6 laboratory, Paris), Nicolas Brisebarre and Jean-Michel Muller introduce an algorithm to compare a binary floating-point (FP) number and a decimal FP number, assuming the "binary encoding" of the decimal formats is used, and with a special emphasis on the basic interchange formats specified by the IEEE 754-2008 standard for FP arithmetic. It is a two-step algorithm: a first pass, based on the exponents only, quickly eliminates most cases, then, when the first pass

does not suffice, a more accurate second pass is performed. They provide an implementation of several variants of our algorithm, and compare them [37].

6.2.8. Correctly rounded sum of floating-point numbers in GNU MPFR

Vincent Lefèvre has designed a new algorithm to compute the correctly rounded sum of several floating-point numbers, each having its own precision and the output having its own precision, as in GNU MPFR. At the same time, the `mpfr_sum` function is being reimplemented (not finished yet). While the old algorithm was just an application of Ziv's method, thus with exponential time and memory complexity in the worst case such as the sum of a huge number and a tiny number, the new algorithm does the sum by blocks (reiterations being needed only in case of cancellations), taking such holes between numbers into account.

6.3. Certified computing and computer algebra

6.3.1. Standardization of interval arithmetic

The IEEE 1788 working group is devoted to the standardization of interval arithmetic. V. Lefèvre and N. Revol are very active in this group. This year has been devoted to a ballot on the whole text of the standard [28], and to editorial work to make it compliant with IEEE rules. The final, remaining step, is the so-called "Sponsor ballot" and it should be completed in 2015.

6.3.2. Interval linear algebra on multi-core processors

For the product of matrices with interval coefficients, fast approximate algorithms have been developed by Philippe Théveny: they compute an enclosure of the exact product. These algorithms rely on the representation of intervals by their midpoints and radii. This representation allows one to use optimized routines for the multiplication of matrices with floating-point coefficients. In [4], the quality of the approximation of several algorithms is established, which accounts for roundoff errors and not only method's errors. A new algorithm is proposed, which requires even less (only 2) calls to a floating-point routine and still offers a good approximation quality, for a well specified type of input matrices. Three of the studied algorithms are implemented on a multi-core architecture. To avoid problems listed in [12] and to offer good performances, Philippe Théveny developed optimizations. The resulting implementations exhibit good performances: guaranteed results are obtained with an overhead less than 3, high numerical intensity and good scalability.

6.3.3. Numerical reproducibility

What is called *numerical reproducibility* is the problem of getting the same result when the scientific computation is run several times, either on the same machine or on different machines. In [12], the focus is on interval computations using floating-point arithmetic: Nathalie Revol and Philippe Théveny identified implementation issues that may invalidate the inclusion property, and presented several ways to preserve this inclusion property. This work has also been replaced in the larger context of numerical validation [15].

6.3.4. Faster multivariate interpolation with multiplicities

Muhammad Chowdhury (U. Western Ontario), Claude-Pierre Jeannerod, Vincent Neiger (ENS de Lyon), Éric Schost (U. Western Ontario), and Gilles Villard proposed in [38] a fast algorithm for interpolating multivariate polynomials with multiplicities. This algorithm relies on the reduction to a problem of simultaneous polynomial approximations, which is then solved using fast structured linear algebra techniques. This algorithm leads to the best known complexity bounds for the interpolation step of the list-decoding of Reed-Solomon codes, Parvaresh-Vardy codes or folded Reed-Solomon codes. In the special case of Reed-Solomon codes, it allows to accelerate the interpolation step of Guruswami and Sudan's list-decoding by a factor (list size)/(multiplicity).

6.3.5. Polynomial system solving

M. Bardet (U. Rouen), J.-C. Faugère (PolSys team) and B. Salvy studied the complexity of Gröbner bases computation, in particular in the generic situation where the variables are in simultaneous Noether position with respect to the system. They gave a bound on the number of polynomials of each degree in a Gröbner basis computed by Faugère's F_5 algorithm in this generic case for the grevlex ordering (which is also a bound on the number of polynomials for a reduced Gröbner basis, independently of the algorithm used) and used it to bound the complexity of the F_5 algorithm [5].

6.3.6. Linear differential equations

In [6], A. Bostan (SpecFun team), K. Raschel (U. Tours) and B. Salvy proved that the sequence $(e_n^{\mathfrak{S}})_{n\geq 0}$ of excursions in the quarter plane corresponding to a nonsingular step set $\mathfrak{S} \subseteq \{0, \pm 1\}^2$ with infinite group does not satisfy any nontrivial linear recurrence with polynomial coefficients. Accordingly, in those cases, the trivariate generating function of the numbers of walks with given length and prescribed ending point is not D-finite. Moreover, they displayed the asymptotics of $e_n^{\mathfrak{S}}$. This completes the classification of these walks.

Colleagues from the LAAS (Toulouse) and B. Salvy provided a new method for computing the probability of collision between two spherical space objects involved in a short-term encounter. In this specific framework of conjunction, classical assumptions reduce the probability of collision to the integral of a 2-D normal distribution over a disk shifted from the peak of the corresponding Gaussian function. Both integrand and domain of integration directly depend on the nature of the short-term encounter. Thus the inputs are the combined sphere radius, the mean relative position in the encounter plane at reference time as well as the relative position covariance matrix representing the uncertainties. The method they presented is based on an analytical expression for the integral. It has the form of a convergent power series whose coefficients verify a linear recurrence. It is derived using Laplace transform and properties of D-finite functions. The new method has been intensively tested on a series of test-cases and compares favorably to other existing works [29].

6.4. Lattices and cryptography

6.4.1. Worst-Case to Average-Case Reductions for Module Lattices

Most lattice-based cryptographic schemes are built upon the assumed hardness of the Short Integer Solution (SIS) and Learning With Errors (LWE) problems. Their efficiencies can be drastically improved by switching the hardness assumptions to the more compact Ring-SIS and RingLWE problems. However, this change of hardness assumptions comes along with a possible security weakening: SIS and LWE are known to be at least as hard as standard (worst-case) problems on euclidean lattices, whereas Ring-SIS and Ring-LWE are only known to be as hard as their restrictions to special classes of ideal lattices, corresponding to ideals of some polynomial rings. Adeline Langlois and Damien Stehlé defined the Module-SIS and Module-LWE problems, which bridge SIS with Ring-SIS, and LWE with Ring-LWE, respectively. They proved that these average-case problems are at least as hard as standard lattice problems restricted to module lattices (which themselves bridge arbitrary and ideal lattices). As these new problems enlarge the toolbox of the lattice-based cryptographer, they could prove useful for designing new schemes. Importantly, the worst-case to average-case reductions for the module problems are (qualitatively) sharp, in the sense that there exist converse reductions. This property is not known to hold in the context of Ring-SIS/Ring-LWE: Ideal lattice problems could reveal easy without impacting the hardness of Ring-SIS/Ring-LWE [8].

6.4.2. Semantically Secure Lattice Codes for the Gaussian Wiretap Channel

Cong Ling (Imperial College, UK), Laura Luzzi (ENSEA), Jean-Claude Belfiore (Telecom ParisTech) and Damien Stehlé proposed a new scheme of wiretap lattice coding that achieves semantic security and strong secrecy over the Gaussian wiretap channel. The key tool in their security proof is the flatness factor which characterizes the convergence of the conditional output distributions corresponding to different messages and leads to an upper bound on the information leakage. They not only introduced the notion of secrecy-good lattices, but also proposed the flatness factor as a design criterion of such lattices. Both the modulo-lattice

Gaussian channel and the genuine Gaussian channel are considered. In the latter case, they proposed a novel secrecy coding scheme based on the discrete Gaussian distribution over a lattice, which achieves the secrecy capacity to within a half nat under mild conditions. No a priori distribution of the message is assumed, and no dither is used in their proposed schemes [9].

6.4.3. GGHLite: More Efficient Multilinear Maps from Ideal Lattices

The Garg-Gentry-Halevi (GGH) Graded Encoding Scheme, based on ideal lattices, is the first plausible approximation to a cryptographic multilinear map. Unfortunately, the scheme requires very large parameters to provide security for its underlying encoding re-randomization process. Adeline Langlois, Damien Stehlé and Ron Steinfeld (Monash University, Australia) formalized, simplified and improved the efficiency and the security analysis of the re-randomization process in the GGH construction. This results in a new construction that they called GGHLite. In particular, they first lowered the size of a standard deviation parameter of the GGH re-randomization process from exponential to polynomial in the security parameter. This first improvement is obtained via a finer security analysis of the so-called drowning step of re-randomization, in which they applied the Rényi divergence instead of the conventional statistical distance as a measure of distance between distributions. Their second improvement is to reduce the number of randomizers needed to 2, independently of the dimension of the underlying ideal lattices. These two contributions allowed them to decrease the bit size of the public parameters to $O(\lambda \log^2 \lambda)$ in GGHLite, with respect to the security parameter λ (for a constant multilinearity parameter κ) [22].

6.4.4. LLL reducing with the most significant bits

Let B be a basis of a Euclidean lattice, and $\widetilde{B}$ an approximation thereof. Saruchi (IIT Delhi, India), Ivan Morel, Damien Stehlé and Gilles Villard gave a sufficient condition on the closeness between $\widetilde{B}$ and B so that an LLL-reducing transformation U for $\widetilde{B}$ remains valid for B. Further, they analysed an efficient reduction algorithm when B is itself a small deformation of an LLL-reduced basis. Applications include speeding-up reduction by keeping only the most significant bits of B, reducing a basis that is only approximately known, and efficiently batching LLL reductions for closely related inputs [30].

6.4.5. Hardness of k-LWE and Applications in Traitor Tracing

San Ling (NTU, Singapore), Duong Hieu Phan (LAGA), Damien Stehlé and Ron Steinfeld (Monash University, Australia) introduced the k-LWE problem, a Learning With Errors variant of the k-SIS problem. The Boneh-Freeman reduction from SIS to k-SIS suffers from an exponential loss in k. Ling *et al.* improved and extended it to an LWE to k-LWE reduction with a polynomial loss in k, by relying on a new technique involving trapdoors for random integer kernel lattices. Based on this hardness result, they presented the first algebraic construction of a traitor tracing scheme whose security relies on the worstcase hardness of standard lattice problems. The proposed LWE traitor tracing is almost as efficient as the LWE encryption. Further, it achieves public traceability, i.e., allows the authority to delegate the tracing capability to untrusted parties. To this aim, Ling *et al.* introduced the notion of projective sampling family in which each sampling function is keyed and, with a projection of the key on a well chosen space, one can simulate the sampling function in a computationally indistinguishable way. The construction of a projective sampling family from k-LWE allows us to achieve public traceability, by publishing the projected keys of the users [27].

6.4.6. Lattice-Based Group Signatures Scheme with Verifier-local Revocation

Support of membership revocation is a desirable functionality for any group signature scheme. Among the known revocation approaches, verifier-local revocation (VLR) seems to be the most flexible one, because it only requires the verifiers to possess some up-to-date revocation information, but not the signers. All of the contemporary VLR group signatures operate in the bilinear map setting, and all of them will be insecure once quantum computers become a reality. Adeline Langlois, San Ling, Khoa Nguyen and Huaxiong Wang (NTU, Singapore) introduced the first lattice-based VLR group signature [21], and thus, the first such scheme that is believed to be quantum-resistant. In comparison with existing lattice-based group signatures, this scheme has several noticeable advantages: support of membership revocation, logarithmic-size signatures, and weaker security assumption. In the random oracle model, our scheme is proved to be secure based on the hardness of

the Shortest Independent Vector Problem with approximation factor $\gamma = \widetilde{O}(n^{1.5})$ - an assumption that is as weak as those of state-of-the-art lattice-based standard signatures. Moreover, this construction works without relying on encryption schemes, which is an intriguing feature for group signatures.

6.4.7. Proxy Re-Encryption Scheme Supporting a Selection of Delegatees

Julien Devigne (Orange Labs), Eleonora Guerrini (Univ. Montpellier 2, LIRMM) and Fabien Laguillaumie adapt the primitive of proxy re-encryption which allows a user to decide that in case of unavailability, one (or several) particular user, the delegatee, will be able to read his confidential messages. They modify it so that a sender can choose who among many potential delegatees will be able to decrypt his messages, and propose a simple and efficient scheme which is secure under chosen plaintext attack under standard algorithmic assumption in a bilinear setting. They also investigate the possibility to add a traceability of the proxy so that one can detect if it has leaked some re-encryption keys [17].

6.4.8. Practical validation of several fault attacks against the Miller algorithm

Ronan Lashermes (SAS-ENSMSE, PRISM), Marie Paindavoine, Nadia El Mrabet (Univ. P8, LIASD), Jacques Fournier (SAS-ENSMSE) and Louis Goubin (UVSQ, PRISM) describe practical implementations of fault attacks against the Miller algorithm, which computes pairing evaluations on algebraic curves. These implementations validate common fault models used against pairings. In the light of the implemented fault attacks, they show that some blinding techniques proposed to protect the algorithm against Side-Channels Analyses cannot be used as countermeasures against the implemented fault attacks [23].

6.4.9. Non-Malleability from Malleability: Simulation-Sound Quasi-Adaptive NIZK Proofs and CCA2-Secure Encryption from Homomorphic Signatures

Verifiability is central to building protocols and systems with integrity. Initially, efficient methods employed the Fiat-Shamir heuristics. Since 2008, the Groth-Sahai techniques have been the most efficient in constructing non-interactive witness indistinguishable and zero-knowledge proofs for algebraic relations in the standard model. For the important task of proving membership in linear subspaces, Jutla and Roy (Asiacrypt 2013) gave significantly more efficient proofs in the quasi-adaptive setting (QA-NIZK). For membership of the row space of a $t \times n$ matrix, their QA-NIZK proofs save $\Omega(t)$ group elements compared to Groth-Sahai. In [26], Benoît Libert, Thomas Peters (UCL, Belgique), Marc Joye (Technicolor, USA) and Moti Yung (Google and Columbia U, USA) gave QA-NIZK proofs made of a *constant* number group elements – regardless of the number of equations or the number of variables – and additionally proved them *unbounded* simulation-sound. Unlike previous unbounded simulation-sound Groth-Sahai-based proofs, their construction does not involve quadratic pairing product equations and does not rely on a chosen-ciphertext-secure encryption scheme. Instead, they built on structure-preserving signatures with homomorphic properties. They applied their methods to design new and improved CCA2-secure encryption schemes. In particular, they built the first efficient threshold CCA-secure keyed-homomorphic encryption scheme (*i.e.*, where homomorphic operations can only be carried out using a dedicated evaluation key) with publicly verifiable ciphertexts.

6.4.10. Born and Raised Distributively: Fully Distributed Non-Interactive Adaptively-Secure Threshold Signatures with Short Shares

Threshold cryptography is a fundamental distributed computational paradigm for enhancing the availability and the security of cryptographic public-key schemes. It does it by dividing private keys into n shares handed out to distinct servers. In threshold signature schemes, a set of at least $t + 1 \leq n$ servers is needed to produce a valid digital signature. Availability is assured by the fact that any subset of $t + 1$ servers can produce a signature when authorized. At the same time, the scheme should remain robust (in the fault tolerance sense) and unforgeable (cryptographically) against up to t corrupted servers; *i.e.*, it adds quorum control to traditional cryptographic services and introduces redundancy. Originally, most practical threshold signatures have a number of demerits: They have been analyzed in a static corruption model (where the set of corrupted servers is fixed at the very beginning of the attack), they require interaction, they assume a trusted dealer in the key generation phase (so that the system is not fully distributed), or they suffer from certain overheads in terms of storage (large share sizes).

In [24], Benoît Libert, Marc Joye (Technicolor, USA) and Moti Yung (Google and Columbia U, USA) constructed practical *fully distributed* (the private key is born distributed), non-interactive schemes – where the servers can compute their partial signatures without communication with other servers – with adaptive security (*i.e.*, the adversary corrupts servers dynamically based on its full view of the history of the system). Their schemes are very efficient in terms of computation, communication, and scalable storage (with private key shares of size $O(1)$, where certain solutions incur $O(n)$ storage costs at each server). Unlike other adaptively secure schemes, their schemes are erasure-free (reliable erasure is a hard to assure and hard to administer property in actual systems). Such a fully distributed highly constrained scheme has been an open problem in the area. In particular, and of special interest, is the fact that Pedersen's traditional distributed key generation (DKG) protocol can be safely employed in the initial key generation phase when the system is born – although it is well-known not to ensure uniformly distributed public keys. An advantage of this is that this protocol only takes one round optimistically (in the absence of faulty player).

6.4.11. Concise Multi-challenge CCA-Secure Encryption and Signatures with Almost Tight Security

To gain strong confidence in the security of a public-key scheme, it is most desirable for the security proof to feature a tight reduction between the adversary and the algorithm solving the under-lying hard problem. Recently, Chen and Wee (Crypto '13) described the first Identity-Based Encryption scheme with almost tight security under a standard assumption. Here, "almost tight" means that the security reduction only loses a factor $O(\lambda)$ – where λ is the security parameter – instead of a factor proportional to the number of adversarial queries. Chen and Wee also gave the shortest signatures whose security almost tightly relates to a simple assumption in the standard model. Also recently, Hofheinz and Jager (Crypto '12) constructed the first CCA-secure public-key encryption scheme in the multi-user setting with tight security. These constructions give schemes that are significantly less efficient in length (and thus, processing) when compared with the earlier schemes with loose reductions in their proof of security. Hofheinz and Jager's scheme has a ciphertext of a few hundreds of group elements, and they left open the problem of finding truly efficient constructions. Likewise, Chen and Wee's signatures and IBE schemes are somewhat less efficient than previous constructions with loose reductions from the same assumptions.

In [25], Benoît Libert, Thomas Peters (UCL, Belgique), Marc Joye (Technicolor, USA) and Moti Yung (Google and Columbia U, USA) considered space-efficient schemes with security almost tightly related to standard assumptions. As a step in solving the open question by Hofheinz and Jager, they constructed an efficient CCA-secure public-key encryption scheme whose chosen-ciphertext security in the multi-challenge, multi-user setting almost tightly relates to the DLIN assumption (in the standard model). Quite remarkably, the ciphertext size decreases to 69 group elements under the DLIN assumption whereas the best previous solution required about 400 group elements. Their scheme is obtained by taking advantage of a new almost tightly secure signature scheme (in the standard model) they developed and which is based on the recent concise proofs of linear subspace membership in the quasi-adaptive non-interactive zero-knowledge setting (QA-NIZK) defined by Jutla and Roy (Asiacrypt '13). The new signature scheme reduces the length of the previous such signatures (by Chen and Wee) by 37% under the Decision Linear assumption, by almost 50% under the K-LIN assumption, and it becomes only 3 group elements long under the Symmetric eXternal Diffie-Hellman assumption. Our signatures are obtained by carefully combining the proof technique of Chen and Wee and the above mentioned QA-NIZK proofs.

CARAMEL Project-Team

6. New Results

6.1. Highlights of the Year

Razvan Barbulescu, ex-PhD student in the team, has received the award "Prix Le Monde de la recherche universitaire", as one of the top-5 PhD thesis in exact science in 2014.

Emmanuel Thomé has received the "Prix Régional du Chercheur" of the Région Lorraine.

Emmanuel Thomé has received the "Prix de l'Association des Amis de l'Université de Lorraine".
BEST PAPER AWARD :
[17] **Eurocrypt 2014**. R. BARBULESCU, P. GAUDRY, A. JOUX, E. THOMÉ.

6.2. Discrete logarithm computation in a prime finite field of 180 decimal digits

Participants: Cyril Bouvier, Pierrick Gaudry, Hamza Jeljeli, Emmanuel Thomé [contact].

In the context of the CATREL ANR project, we performed a new computation of a discrete logarithm modulo a 180 digit (596-bit) prime using the number field sieve algorithm. Previous records were 135-digit (448 bits, done in 2006) and 160-digit (530-bit, done in 2007) primes. This is, to date, the largest computation in a prime field. In total, this took the equivalent of 130 years on one CPU core.

6.3. Discrete logarithm in finite fields of small extension degree

Participant: Pierrick Gaudry [contact].

Together with Razvan Barbulescu (CNRS, IMJ-PRG), Aurore Guillevic and François Morain (GRACE project-team), we investigated the discrete logarithm problem in the case of finite fields of the form $\mathbb{F}_{p^n}$, where $n > 1$ is a small integer. We proposed in a preprint — a part of which was accepted to Eurocrypt 2015 — various theoretical and practical improvements [25]:

- new methods for selecting polynomials,
- better (heuristic) asymptotic complexity in the case where $n \approx \log p$, and
- use of algebraic number theory to show that in some cases we can skip the Schirokauer maps.

We have adapted CADO-NFS in order to perform a record computation in a field of the form $\mathbb{F}_{p^2}$, where p^2 has 180 digits. To our knowledge, this is the first time that the number field sieve algorithm is used in practice for record-size computations in this type of fields.

6.4. Igusa class polynomials computation for class number 20,016

Participant: Emmanuel Thomé [contact].

In collaboration with the LFANT project-team, Emmanuel Thomé and Andreas Enge completed the computation of Igusa class polynomials for a quartic CM field whose Igusa class number is 20,016. That is more than 20 times more than the previous state of the art. This has been made possible with the CMH software, which corresponds to the article [10].

6.5. Isogeny graphs for curves with maximal real multiplication

Participant: Emmanuel Thomé [contact].

Emmanuel Thomé and Sorina Ionica (currently with the LFANT project-team) worked on a new algorithm for computing isogeny graphs for Jacobians of curves having the special property that the intersection of their endomorphism ring with its real subfield is maximal. The resulting algorithm is the first depth-first algorithm for this task. This work has been submitted [29].

6.6. Polynomial selection for the Number Field Sieve

Participants: Cyril Bouvier, Nicholas Coxon, Alexander Kruppa, Paul Zimmermann [contact].

A new polynomial selection algorithm for GNFS (General Number Field Sieve) has been described in a preprint [24] and implemented in CADO-NFS. We demonstrate the efficiency of this algorithm by exhibiting a better polynomial than the one used for the factorization of RSA-768, and a polynomial for RSA-1024 that outperforms the best published one.

Montgomery's method of polynomial selection for GNFS has been analysed in a preprint [27]. Criteria for the selection of good parameters for Montgomery's method are given, and the existence of the modular geometric progressions used in the method is considered.

6.7. Beyond double precision

Participant: Paul Zimmermann [contact].

A project entitled "Beyond Double Precision" (BeDoP) has been submitted to the European Research Council (ERC) for funding (advanced grant category). The BeDoP project will (i) demonstrate the limits of double precision on large-scale applications, (ii) make multiple-precision tools easier to use in modern computer languages, and (iii) improve the efficiency and robustness of those tools, in particular by using formal proof techniques. Our dream with the BeDoP project is that scientific computations on exascale computers will no longer give very fast and very wrong results, but instead give very fast and very accurate results.

6.8. Gröbner bases for sparse algebraic systems

Participant: Pierre-Jean Spaenlehauer [contact].

In collaboration with Jean-Charles Faugère and Jules Svartz (POLSYS project-team), new Gröbner bases algorithms have been proposed in [20] to solve efficiently sparse systems of multivariate polynomial equations. Moreover, new complexity bounds have also been proved; they extend in a unified way previous bounds known for solving multi-homogeneous systems with Gröbner bases. For such systems, a proof-of-concept prototype implementation of these algorithms achieves large speed-ups compared to state-of-the-art optimized Gröbner bases algorithms.

6.9. Faster index calculus in algebraic curves

Participant: Maike Massierer [contact].

A possible application of the new ideas speeding up the function field sieve algorithm to index calculus in Jacobians of algebraic curves of large genus has been studied in [30]. Based on a number of practical experiments as well as theoretical considerations, a conjecture has been formulated. It implies that the new ideas only apply to curves which are not interesting in the context of the discrete logarithm problem.

6.10. FFS factory

Participant: Jérémie Detrey [contact].

An extension of Coppersmith's "factorization factory" and Barbulescu's "discrete logarithm factory" to the Function Field Sieve was proposed, dubbed the "FFS factory" [28]. The idea is to batch discrete logarithm computations in finite fields of different extension degrees, sharing the sieving step on the algebraic side between all these finite fields. A careful analysis proved that this approach can be used to lower the overall asymptotic complexity. This was also illustrated with a practical experiment in which the discrete logarithm problem was solved for all of the 50 binary fields of the form $\mathbb{F}_{2^n}$ with n odd ranging from 601 to 699.

CASCADE Project-Team

5. New Results

5.1. Results

All the results of the team have been published in journals or conferences (see the list of publications). They are all related with the research program (see before) and the research projects (see after):

- More efficient constructions with lattices
- New constructions from pairings
- Delegation of computations
- Analysis of pseudo-random generators
- Advanced primitives for the privacy in the cloud
- Cryptanalysis of symmetric primitives
- New leakage-resilient primitives
- Stronger security with related-key security

CRYPT Team

4. New Results

4.1. Highlights of the Year

The team published [20] improved single-key attacks on reduced-round AES: AES is currently the most widespread block cipher standard, it is implemented in Intel processors.

The team also showed [18] how to speed-up a well-known public-key cryptanalysis technique: finding small roots of univariate polynomial congruences. This technique is used to break special cases of the RSA cryptosystem.

Phong Nguyen was Program co-Chair of the 33rd IACR Eurocrypt Conference (EUROCRYPT 2014) [22].

GALAAD2 Team

6. New Results

6.1. Algebraic representations for geometric modeling

6.1.1. A comparison of different notions of ranks of symmetric tensors

Participants: Alessandra Bernardi, Jérome Brachat, Bernard Mourrain.

In [2], we introduce various notions of rank for a symmetric tensor, namely: rank, border rank, catalecticant rank, generalized rank, scheme length, border scheme length, extension rank and smoothable rank. We analyze the stratification induced by these ranks. The mutual relations between these stratifications, allow us to describe the hierarchy among all the ranks. We show that strict inequalities are possible between rank, border rank, extension rank and catalecticant rank. Moreover we show that scheme length, generalized rank and extension rank coincide.

6.1.2. Dimensions and bases of hierarchical tensor-product splines

Participant: Bernard Mourrain.

In [1], we prove that the dimension of trivariate tensor-product spline space of tri-degree (m,m,m) with maximal order of smoothness over a three-dimensional domain coincides with the number of tensor-product B-spline basis functions acting effectively on the domain considered. A domain is required to belong to a certain class. This enables us to show that, for a certain assumption about the configuration of a hierarchical mesh, hierarchical B-splines span the spline space. This paper presents an extension to three-dimensional hierarchical meshes of results proposed recently by Giannelli and Jüttler for two-dimensional hierarchical meshes.

Joint work with Dmitry Berdinsky, Taiwan Kim, Oh Min-Jae, Sutipong Kiatpanichgij (Department of Naval Architecture and Ocean Engineering, Seoul, South Korea), Cesare Bracco (Dipartimento di Matematica "Giuseppe Peano", Torino, Italy), Durkbin Cho (Department of Mathematics, Dongguk, South Korea).

6.1.3. Bounds on the dimension of trivariate spline spaces: A homological approach

Participant: Bernard Mourrain.

In [8], we consider the vector space of globally differentiable piecewise polynomial functions defined on a three-dimensional polyhedral domain partitioned into tetrahedra. We prove new lower and upper bounds on the dimension of this space by applying homological techniques. We give an insight of different ways of approaching this problem by exploring its connections with the Hilbert series of ideals generated by powers of linear forms, fat points, the so-called Fröberg–Iarrobino conjecture, and the weak Lefschetz property.

Joint work with Nelly Villamizar (RICAM - Johann Radon Institute for Computational and Applied Mathematics, Linz, Austria)

6.1.4. High-quality construction of analysis-suitable trivariate NURBS solids by reparameterization methods

Participants: André Galligo, Bernard Mourrain.

High-quality volumetric parameterization of computational domain plays an important role in three-dimensional isogeometric analysis. Reparameterization techniques can improve the distribution of isoparametric curves/surfaces without changing the geometry. In [10], using the reparameterization method, we investigate the high-quality construction of analysis-suitable NURBS volumetric parameterization. Firstly, we introduce the concept of volumetric reparameterization, and propose an optimal Möbius transformation to improve the quality of the isoparametric structure based on a new uniformity metric. Secondly, from given boundary NURBS surfaces, we present a two-stage scheme to construct the analysis-suitable volumetric parameterization: in the first step, uniformity-improved reparameterization is performed on the boundary surfaces to achieve high-quality isoparametric structure without changing the shape; in the second step, from a new variational harmonic metric and the reparameterized boundary surfaces, we construct the optimal inner control points and weights to achieve an analysis-suitable NURBS solid. Several examples with complicated geometry are presented to illustrate the effectiveness of proposed methods.

Joint work with Gang Xu (College of computer - Hangzhou Dianzi University, China), Timon Rabczuk (Bauhaus-Universität Weimar, Germany).

6.1.5. Spline Spaces over Quadrangle Meshes with Complex Topologies

Participants: André Galligo, Bernard Mourrain, Meng Wu.

Motivated by Magneto Hydrodynamic (MHD) simulation with isoparametric elements method, we pursue our work on new types of spline functions defined over a quadrangular mesh, that can follow isobaric curves with node singularities. The practicability of these splines is analyzed for different geometries related to MHD simulation.

This work is done in collaboration with Boniface Nkonga (Inria, EPI CASTOR and University of Nice).

6.1.6. Parametric modeling for ship hull deformation

Participant: Elisa Berrini.

The objective of the work is to develop a parametric modeler tool, allowing consistent ship hull deformations with respect to classic naval architecture design constraints. This work will be applied in automatic shape optimization process. Two scientific problematics are addressed : 1) The parametrization of the hull: the numerical representation of the shape from a defined set of parameters; 2) The deformations of curves and surfaces: getting a new shape by modifying chosen parameters from the parameterization set. The consistency with naval architecture constraints is essential.

To produce realistic models, we want to use methods similar to naval architects' ones. The approach under development is based on the extraction and deformation of skeletons curves.

6.2. Algebraic algorithms for geometric computing

6.2.1. Resultant of an equivariant polynomial system with respect to the symmetric group

Participants: Laurent Busé, Anna Karasoulou.

Given a system of n homogeneous polynomials in n variables which is equivariant with respect to the canonical actions of the symmetric group of n symbols on the variables and on the polynomials, it is proved that its resultant can be decomposed into a product of several smaller resultants that are given in terms of some divided differences. As an application, we obtain a decomposition formula for the discriminant of a multivariate homogeneous symmetric polynomial.

This work is submitted for publication [14].

6.2.2. Delaunay Mesh Generation of NURBS Surfaces

Participant: Laurent Busé.

We introduce a method for isotropic triangle meshing of NURBS surfaces. Based on Delaunay filtering and refinement, our approach departs from previous work by meshing in embedding space instead of parametric space. The meshing engine relies upon a novel line/surface intersection test, based on the matrix-based implicit representation of NURBS surfaces and numerical methods in linear algebra such as singular value and eigenvalue decompositions. A careful treatment of degenerate cases makes our approach robust to intersection points with multiple pre-images. In addition to ensure both approximation accuracy and mesh quality, our approach is seamless as it does not depend on the initial decomposition into NURBS patches, and is oblivious to the parameterization of the patches. Removing such dependencies provides us with a means to reliably mesh across patches with greater control over mesh sizing and shape of the elements.

This work was done in collaboration with Jingjing Shen and Neil Dodgson from Cambridge University and Pierre Alliez from TITANE.

6.2.3. *Toric Border Basis*

Participant: Bernard Mourrain.

In [11], we extend the theory and the algorithms of Border bases to systems of Laurent polynomial equations, defining "toric" roots. Instead of introducing new variables and new relations to saturate by the variable inverses, we propose a more efficient approach which works directly with the variables and their inverse. We show that the commutation relations and the inversion relations characterize toric border bases. We explicitly describe the first syzygy module associated to a toric border basis in terms of these relations. Finally, a new border basis algorithm for Laurent polynomials is described and a proof of its termination is given for zero-dimensional toric ideals.

Joint work with Philippe Trébuchet (LIP6 - UPMC).

6.2.4. *Border Basis relaxation for polynomial optimization*

Participants: Marta Abril-Bucero, Bernard Mourrain.

A relaxation method based on border basis reduction which improves the efficiency of Lasserre's approach is proposed to compute the optimum of a polynomial function on a basic closed semi-algebraic set. A new stopping criterion is given to detect when the relaxation sequence reaches the minimum, using a sparse flat extension criterion. We also provide a new algorithm to reconstruct a finite sum of weighted Dirac measures from a truncated sequence of moments, which can be applied to other sparse reconstruction problems. As an application, we obtain a new algorithm to compute zero-dimensional minimizer ideals and the minimizer points or zero-dimensional G-radical ideals. Experimentations show the impact of this new method on significant benchmarks. See [12].

6.2.5. *Flat extensions in $*$-algebra*

Participant: Bernard Mourrain.

The objective of this work is to develop a flat extension characterization on moment matrices in the non-commutative case. We give a flat extension theorem for positive linear functionals on $_*$-algebras. The theorem is applied to truncated moment problems on cylinder sets, on matrices of polynomials and on enveloping algebras of Lie algebras. See [17].

Joint work with Konrad Schmüdgen, University of Leipzig, Germany.

6.3. Symbolic-Numeric Analysis

6.3.1. *Cubatures, and related problems, with symmetry*

Participants: Mathieu Collowald, Evelyne Hubert.

We address the computation of cubature formulae as a moment problem. Symmetry by finite groups arise naturally for cubatures. We developed the algebraic results to use the symmetry in order to reduce the number of parameters and the size of the matrices involved in the flat extension.

6.3.2. Quantitative Equidistribution for the Solutions of Systems of Sparse Polynomial Equations

Participant: André Galligo.

For a system of Laurent polynomials $f_1, ..., f_n \in \mathbb{C}[x_1^{\pm 1}, ..., x_n^{\pm 1}]$ whose coefficients are not too big with respect to its directional resultants, we show in [6] that the solutions in the algebraic torus $(\mathbb{C}^*)^n$ of the system of equations $f_1 = \cdots = f_n = 0$, are approximately equidistributed near the unit polycircle. This generalizes to the multivariate case a classical result due to Erdös and Turán on the distribution of the arguments of the roots of a univariate polynomial. We apply this result to bound the number of real roots of a system of Laurent polynomials, and to study the asymptotic distribution of the roots of systems of Laurent polynomials over $\mathbb{Z}$ and of random systems of Laurent polynomials over $\mathbb{C}$.

Joint work with Carlos D'Andrea (DM-UBA - Departamento de Matemática, Spain), Martin Sombra (ICREA & Universitat de Barcelona, Spain).

GEOMETRICA Project-Team

6. New Results

6.1. Highlights of the Year

[10] was elected among the notable articles of 2013 by ACM and Computing Reviews (see http://computingreviews.com/recommend/bestof/notableitems_2013.cfm).

6.2. Mesh Generation and Geometry Processing

6.2.1. A Surface Reconstruction Method for In-Detail Underwater 3D Optical Mapping

Participant: Mariette Yvinec.

In collaboration with Pierre Alliez (EPI Titane), Ricard Campos (University of Girona), Raphael Garcia (University of Girona)

Underwater range scanning techniques are starting to gain interest in underwater exploration, providing new tools to represent the seafloor. These scans (often) acquired by underwater robots usually result in an unstructured point cloud, but given the common downward-looking or forward-looking configuration of these sensors with respect to the scene, the problem of recovering a piecewise linear approximation representing the scene is normally solved by approximating these 3D points using a heightmap (2.5D). Nevertheless, this representation is not able to correctly represent complex structures, especially those presenting arbitrary concavities normally exhibited in underwater objects. We present a method devoted to full 3D surface reconstruction that does not assume any specific sensor configuration. The method presented is robust to common defects in raw scanned data such as outliers and noise often present in extreme environments such as underwater, both for sonar and optical surveys. Moreover, the proposed method does not need a manual preprocessing step. It is also generic as it does not need any information other than the points themselves to work. This property leads to its wide application to any kind of range scanning technologies and we demonstrate its versatility by using it on synthetic data, controlled laser-scans, and multibeam sonar surveys. Finally, and given the unbeatable level of detail that optical methods can provide, we analyze the application of this method on optical datasets related to biology, geology and archeology. [23]

6.2.2. A Transfer Principle and Applications to Eigenvalue Estimates for Graphs

Participant: David Cohen-Steiner.

In collaboration with Omid Amini (ENS),

In this paper, we prove a variant of the Burger-Brooks transfer principle which, combined with recent eigenvalue bounds for surfaces, allows to obtain upper bounds on the eigenvalues of graphs as a function of their genus. More precisely, we show the existence of a universal constants C such that the k-th eigenvalue λ_k of the normalized Laplacian of a graph G of (geometric) genus g on n vertices satisfies $\lambda_k \leq C d_{max}(g+k)/n$ where d_max denotes the maximum valence of vertices of the graph. This result is tight up to a change in the value of the constant C. We also use our transfer theorem to relate eigenvalues of the Laplacian on a metric graph to the eigenvalues of its simple graph models, and discuss an application to the mesh partitioning problem. [44]

6.3. Topological and Geometric Inference

6.3.1. Only distances are required to reconstruct submanifolds

Participants: Jean-Daniel Boissonnat, Steve Oudot.

In collaboration with Ramsay Dyer (Johann Bernouilli Institute, University of Groningen, Pays Bas) and Arijit Ghosh (Max-Planck-Institut für Informatik, Saarbrücken, Germany).

In [45], we give the first algorithm that outputs a faithful reconstruction of a submanifold of Euclidean space without maintaining or even constructing complicated data structures such as Voronoi diagrams or Delaunay complexes. Our algorithm uses the witness complex and relies on the stability of *power protection*, a notion introduced in this paper. The complexity of the algorithm depends exponentially on the intrinsic dimension of the manifold, rather than the dimension of ambient space, and linearly on the dimension of the ambient space. Another interesting feature of this work is that no explicit coordinates of the points in the point sample is needed. The algorithm only needs the *distance matrix* as input, i.e., only distance between points in the point sample as input.

6.3.2. Computing Persistent Homology with Various Coefficient Fields in a Single Pass

Participants: Jean-Daniel Boissonnat, Clément Maria.

In [32], we introduce an algorithm to compute the persistent homology of a filtered complex with various coefficient fields in a single matrix reduction. The algorithm is output-sensitive in the total number of distinct persistent homological features in the diagrams for the different coefficient fields. This computation allows us to infer the prime divisors of the torsion coefficients of the integral homology groups of the topological space at any scale, hence furnishing a more informative description of topology than persistence in a single coefficient field. We provide theoretical complexity analysis as well as detailed experimental results.

6.3.3. Recognizing shrinkable complexes is NP-complete

Participants: Olivier Devillers, Marc Glisse.

In collaboration with Dominique Attali (Gipsa-lab, Grenoble), Sylvain Lazard (Inria Nancy - Grand Est)

We say that a simplicial complex is shrinkable if there exists a sequence of admissible edge contractions that reduces the complex to a single vertex. We prove [31] that it is NP-complete to decide whether a (three-dimensional) simplicial complex is shrinkable. Along the way, we describe examples of contractible complexes which are not shrinkable.

6.3.4. Zigzag Zoology: Rips Zigzags for Homology Inference

Participant: Steve Oudot.

In collaboration with Donald Sheehy (University of Connecticut)

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 [28] 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.

6.3.5. Zigzag Persistence via Reflections and Transpositions

Participants: Clément Maria, Steve Oudot.

We introduce [40] a simple algorithm for computing zigzag persistence, designed in the same spirit as the standard persistence algorithm. Our algorithm reduces a single matrix, maintains an explicit set of chains encoding the persistent homology of the current zigzag, and updates it under simplex insertions and removals. The total worst-case running time matches the usual cubic bound. A noticeable difference with the standard persistence algorithm is that we do not insert or remove new simplices "at the end" of the zigzag, but rather "in the middle". To do so, we use arrow reflections and transpositions, in the same spirit as reflection functors in quiver theory. Our analysis introduces a new kind of reflection called the "weak-diamond", for which we are able to predict the changes in the interval decomposition and associated compatible bases. Arrow transpositions have been studied previously in the context of standard persistent homology, and we extend the study to the context of zigzag persistence. For both types of transformations, we provide simple procedures to update the interval decomposition and associated compatible homology basis.

6.3.6. Topological analysis of scalar fields with outliers

Participants: Mickaël Buchet, Frédéric Chazal, Steve Oudot.

In collaboration with Tamal K. Dey (University of Ohio) Fengtao Fan (University of Ohio) Yusu Wang (University of Ohio)

We extend [57] the notion of the distance to a measure from Euclidean space to probability measures on general metric spaces as a way to do topological data analysis in a way that is robust to noise and outliers. We then give an efficient way to approximate the sub-level sets of this function by a union of metric balls and extend previous results on sparse Rips filtrations to this setting. This robust and efficient approach to topological data analysis is illustrated with several examples from an implementation.

6.3.7. Efficient and Robust Persistent Homology for Measures.

Participants: Mickaël Buchet, Frédéric Chazal, Steve Oudot.

In collaboration with Donald Sheehy (University of Connecticut)

In [34], we extend the notion of the distance to a measure from Euclidean space to probability measures on general metric spaces as a way to do topological data analysis in a way that is robust to noise and outliers. We then give an efficient way to approximate the sub-level sets of this function by a union of metric balls and extend previous results on sparse Rips filtrations to this setting. This robust and efficient approach to topological data analysis is illustrated with several examples from an implementation.

6.3.8. Persistence-based Structural Recognition

Participants: Frédéric Chazal, Maksims Ovsjanikovs.

In collaboration with Chunyuan Li (former intern in Saclay in 2013)

In [39] we present a framework for object recognition using topological persistence. In particular, we show that the so-called persistence diagrams built from functions defined on the objects can serve as compact and informative descriptors for images and shapes. Complementary to the bag-of-features representation, which captures the distribution of values of a given function, persistence diagrams can be used to characterize its structural properties, reflecting spatial information in an invariant way. In practice, the choice of function is simple: each dimension of the feature vector can be viewed as a function. The proposed method is general: it can work on various multimedia data, including 2D shapes, textures and triangle meshes. Extensive experiments on 3D shape retrieval, hand gesture recognition and texture classification demonstrate the performance of the proposed method in comparison with state-of-the-art methods. Additionally, our approach yields higher recognition accuracy when used in conjunction with the bag-offeatures.

6.3.9. Convergence rates for persistence diagram estimation in Topological Data Analysis

Participants: Frédéric Chazal, Marc Glisse, Bertrand Michel.

In collaboration with Catherine Labruère (University of Burgundy)

Computational topology has recently known an important development toward data analysis, giving birth to the field of topological data analysis. Topological persistence, or persistent homology, appears as a fundamental tool in this field. In [36], we study topological persistence in general metric spaces, with a statistical approach. We show that the use of persistent homology can be naturally considered in general statistical frameworks and persistence diagrams can be used as statistics with interesting convergence properties. Some numerical experiments are performed in various contexts to illustrate our results.

6.3.10. Stochastic Convergence of Persistence Landscapes and Silhouettes

Participant: Frédéric Chazal.

In collaboration with Brittany Fasy (Tulane University) Fabrizio Lecci (Carnegie Mellon University) Alessandro Rinaldo (Carnegie Mellon University) Larry Wasserman (Carnegie Mellon University)

Persistent homology is a widely used tool in Topological Data Analysis that encodes multiscale topological information as a multi-set of points in the plane called a persistence diagram. It is difficult to apply statistical theory directly to a random sample of diagrams. Instead, we can summarize the persistent homology with the persistence landscape, introduced by Bubenik, which converts a diagram into a well-behaved real-valued function. In [35], we investigate the statistical properties of landscapes, such as weak convergence of the average landscapes and convergence of the bootstrap. In addition, we introduce an alternate functional summary of persistent homology, which we call the silhouette, and derive an analogous statistical theory.

6.3.11. Subsampling Methods for Persistent Homology

Participants: Frédéric Chazal, Bertrand Michel.

In collaboration with Brittany Fasy (Tulane University) Fabrizio Lecci (Carnegie Mellon University) Alessandro Rinaldo (Carnegie Mellon University) Larry Wasserman (Carnegie Mellon University)

Persistent homology is a multiscale method for analyzing the shape of sets and functions from point cloud data arising from an unknown distribution supported on those sets. When the size of the sample is large, direct computation of the persistent homology is prohibitive due to the combinatorial nature of the existing algorithms. We propose to compute the persistent homology of several subsamples of the data and then combine the resulting estimates. We study the risk of two estimators and we prove that the subsampling approach carries stable topological information while achieving a great reduction in computational complexity.

6.3.12. The observable structure of persistence modules

Participant: Frédéric Chazal.

In collaboration with Vin de Silva (Pomona College) William Crawley-Boevey (University of Leeds)

In persistent topology, q-tame modules appear as a natural and large class of persistence modules indexed over the real line for which a persistence diagram is de- finable. However, unlike persistence modules indexed over a totally ordered finite set or the natural numbers, such diagrams do not provide a complete invariant of q-tame modules. The purpose of [59] is to show that the category of persistence modules can be adjusted to overcome this issue. We introduce the observable category of persis- tence modules: a localization of the usual category, in which the classical properties of q-tame modules still hold but where the persistence diagram is a complete isomorphism invariant and all q-tame modules admit an interval decomposition.

6.4. Data Structures and Robust Geometric Computation

6.4.1. Efficiently Navigating a Random Delaunay Triangulation

Participants: Olivier Devillers, Ross Hemsley.

In collaboration with Nicolas Broutin (EPI RAP*)*

Planar graph navigation is an important problem with significant implications to both point location in geometric data structures and routing in networks. Whilst many algorithms have been proposed, very little theoretical analysis is available for the properties of the paths generated or the computational resources required to generate them. In this work, we propose and analyse a new planar navigation algorithm for the Delaunay triangulation. We then demonstrate a number of strong theoretical guarantees for the algorithm when it is applied to a random set of points in a convex region [33]. In a side result, we give a new polylogarithmic bound on the maximum degree of a random Delaunay triangulation in a smooth convex, that holds with probability one as the number of points goes to infinity. In particular, our new bound holds even for points arbitrarily close to the boundary of the domain. [56]

6.4.2. A chaotic random convex hull

Participants: Olivier Devillers, Marc Glisse, Rémy Thomasse.

The asymptotic behavior of the expected size of the convex hull of uniformly random points in a convex body in $\mathcal{R}^d$ is polynomial for a smooth body and polylogarithmic for a polytope. We construct a body whose expected size of the convex hull oscillates between these two behaviors when the number of points increases [62]

6.4.3. A generator of random convex polygons in a disc

Participants: Olivier Devillers, Rémy Thomasse.

In collaboration with Philippe Duchon (LABRI)

Let $\mathcal{D}$ a disc in $\mathbb{R}^2$ with radius 1 centered at $\mathfrak{o}$, and $(x_1, \cdots, x_n)$ a sample of n points uniformly and independently distributed in $\mathcal{D}$. Let's define the polygon P_n as the convex hull of $(x_1, \cdots, x_n)$, and $f_0(P_n)$ its number of vertices. This kind of polygon has been well studied, and it is known, see [65], that

$$\mathbb{E}f_0(P_n) = c\ n^{\frac{1}{3}} + o(n^{\frac{1}{3}})$$

where $c > 0$ is constant. To generate such a polygon, one can explicitly generate n points uniformly in $\mathcal{D}$ and compute the convex hull. For a very large quantity of points, it could be interesting to generate less points to get the same polygon, for example to have some estimations on asymptotic properties, such as the distribution of the size of the edges.We propose an algorithm that generate far less points at random in order to get P_n, so that the time and the memory needed is reduced for n large. Namely [61], we generate a number of points of the same order of magnitude than the final hull, up to a polylogarithmic factor

6.4.4. On the complexity of the representation of simplicial complexes by trees

Participants: Jean-Daniel Boissonnat, Dorian Mazauric.

In [46], we investigate the problem of the representation of simplicial complexes by trees. We introduce and analyze local and global tree representations. We prove that the global tree representation is more efficient in terms of time complexity for searching a given simplex and we show that the local tree representation is more ecient in terms of size of the structure. The simplicial complexes are modeled by hypergraphs. We then prove that the associated combinatorial optimization problems are very dicult to solve and to approximate even if the set of maximal simplices induces a cubic graph, a planar graph, or a bounded degree hypergraph. However, we prove polynomial time algorithms that compute constant factor approximations and optimal solutions for some classes of instances.

6.4.5. Building Efficient and Compact Data Structures for Simplicial Complexes

Participant: Jean-Daniel Boissonnat.

In collaboration with Karthik C.S (Weizmann Institute of Science, Israël) and Sébastien Tavenas (Max-Planck-Institut für Informatik, Saarbrücken, Germany).

The Simplex Tree is a recently introduced data structure that can represent abstract simplicial complexes of any dimension and allows to efficiently implement a large range of basic operations on simplicial complexes. In this paper, we show how to optimally compress the simplex tree while retaining its functionalities. In addition, we propose two new data structures called Maximal Simplex Tree and Compact Simplex Tree. We analyze the Compressed Simplex Tree, the Maximal Simplex Tree and the Compact Simplex Tree under various settings.

6.4.6. Delaunay triangulations over finite universes

Participant: Jean-Daniel Boissonnat.

In collaboration with Ramsay Dyer (Johann Bernouilli Institute, University of Groningen, Pays Bas) and Arijit Ghosh (Max-Planck-Institut für Informatik, Saarbrücken, Germany).

The witness complex was introduced by Carlsson and de Silva as a weak form of the Delaunay complex that is suitable for finite metric spaces and is computed using only distance comparisons. The witness complex $\mathrm{Wit}(L, W)$ is defined from two sets L and W in some metric space X: a finite set of points L on which the complex is built, and a set W of witnesses that serves as an approximation of X. A fundamental result of de Silva states that $\mathrm{Wit}(L, W) = \mathrm{Del}(L)$ if $W = X = \mathbb{R}^d$. In this paper we give conditions on L that ensure that the witness complex and the Delaunay triangulation coincide when $W \subset \mathbb{R}^d$ is a finite set, and we introduce a new perturbation scheme to compute a perturbed set L' close to L such that $\mathrm{Del}(L') = \mathrm{Wit}(L', W)$. The algorithm constructs $\mathrm{Wit}(L', W)$ in time sublinear in $|W|$.

The only numerical operations used by our algorithms are (squared) distance comparisons (i.e., predicates of degree 2). In particular, we do not use orientation or in-sphere predicates, whose degree depends on the dimension d, and are difficult to implement robustly in higher dimensions. Although the algorithm does not compute any measure of simplex quality, a lower bound on the thickness of the output simplices can be guaranteed. Another novelty in the analysis is the use of the Moser-Tardos constructive proof of the general Lovász local lemma.

GRACE Project-Team

6. New Results

6.1. Highlights of the Year

- F. Morain and A. Guillevic (with their co-authors R. Barbulescu and P. Gaudry) broke the discrete logarithm world record for finite fields of the form $GF(p^2)$ with a prime p of 80 decimal digits. The new techniques form the preprint [31].
- D. Augot and M. Finiasz received the best paper award at FSE 2014 [17]. FSE is the most important conference devoted to symmetric cryptography. Grace contribution is to propose a mathematical construction which enables direct construction of so-called diffusion layers in block ciphers.
- A. Zeh, former Grace PhD student, received the special Prize of the Université Franco-Allemande (UFA) Jury 2014 at the French Ambassy in Berlin, on November 21st.

BEST PAPER AWARD :
[17] **21st International Workshop on Fast Software Encryption, FSE 2014**. D. AUGOT, M. FINIASZ.

6.2. Diffusion layers for block ciphers

MDS matrices allow the construction of optimal linear diffusion layers in block ciphers. However, MDS matrices usually have a large description (for example, they can never be sparse), and this results in costly software/hardware implementations. We can solve this problem using *recursive MDS matrices*, which can be computed as a power of a simple companion matrix—and thus have a compact description suitable for constrained environments. Until now, finding recursive MDS matrices required an exhaustive search on families of companion matrices; this clearly limited the size of MDS matrices that one could look for. We have found a new direct construction, based on shortened BCH codes, which allows us to efficiently construct these matrices for arbitrary parameter sizes [17]. D. Augot and M. Finiasz received the best paper award at FSE 2014, and were invited to submit an extended journal version to *Journal of Cryptology*.

P. Karpman started to study sub-optimal diffusion layers, which can be built using algebraic geometry codes with a large automorphism group. Preliminary work has been done, leading to promising results [18]. To properly assert the cryptanalytic properties of these codes, V. Ducet is starting to implement a method for computing efficiently the weight distribution of AG codes.

6.3. Rank metric codes over infinite fields

Rank metric and Gabidulin codes over the rationals promise interesting applications to space-time coding. We have constructed optimal codes, similar to Gabidulin codes, in the case of infinite fields. We use algebraic extensions, and we have determined the condition on the considered extension to enable this construction. For example: we can design codes with complex coefficients, using number fields and Galois automorphisms. Then, in the rank metric setting, codewords can be seen as matrices. In this setting, a channel introduces errors (a matrix of small rank r added to the codeword) and erasures (s_r rows and s_c columns of the matrix are erased). We have developed an algorithm (adapted from the Welch–Berlekamp algorithm) to recover the right codeword in the presence of an error of rank weight up to $r + s_c + s_r \leqslant d - 1$, where d is the minimal distance of the code. As opposed to the finite field case, we are confronted by coefficient size growth. We solve this problem by computing modulo prime ideals. Using these codes we can completely bypass intermediate constructions using finite fields, which were the stumbling-block in classic constructions.

We also have used this framework to build rank-metric codes over the field of rational functions, using algebraic function fields with cyclic Galois group (Kummer and Artin extensions). These codes can be seen as a generator of infinitely many convolutional codes [25].

6.4. Tensor rank of multiplication over finite fields

Determining the tensor rank of multiplication over finite fields is a problem of great interest in algebraic complexity theory, but it also has practical importance: it allows us to obtain multiplication algorithms with a low bilinear complexity, which are of crucial significance in cryptography. In collaboration with S. Ballet and J. Chaumine [35], J. Pieltant obtained new asymptotic bounds for the symmetric tensor rank of multiplication in finite extensions of finite fields $\mathbb{F}_q$. In the more general (not-necessarily-symmetric) case, J. Pieltant and H. Randriam obtained new uniform upper bounds for multiplication in extensions of $\mathbb{F}_q$. They also gave purely asymptotic bounds substantially improving those coming from uniform bounds, by using a family of Shimura curves defined over $\mathbb{F}_q$. This work will appear in Mathematics of Computation [15].

6.5. Filtration Attacks against McEliece Cryptosystem

The McEliece encryption scheme based on binary Goppa codes was one of the first public-key encryption schemes [39]. Its security rests on the difficulty of decoding an arbitrary code. The original proposal uses classical Goppa codes, and while it still remains unbroken, it requires a huge size of key. On the other hand, many derivative systems based on other families of algebraic codes have been subject to key recovery attacks. Up to now, key recovery attacks were based either on a variant of Sidelnikov and Shestakov's attack [40], where the first step involves the computation of minimum-weight codewords, or on the resolution of a system of polynomial equations using Gröbner bases.

In [10], A. Couvreur, P. Gaborit, V. Gauthier, A. Otmani and J.-P. Tillich introduced a new paradigm of attack called *filtration attacks*. The general principle decomposes in two steps:

1. **Distinguishing** the public code from a random one using the square code operation.
2. **Computing a filtration** of the public code using the distinguisher, and deriving from this filtration an efficient decoding algorithm for the public code.

This new style of attack allowed A. Couvreur, A. Otmani and J.-P. Tillich to break (in polynomial time) McEliece based on wild Goppa codes over quadratic extensions [23]; and A. Couvreur, I. Márquez-Corbella, and R. Pellikaan to break McEliece based on algebraic geometry codes from curves of arbitrary genus [22], [26].

6.6. A new bound on the number of rational points of arbitrary projective varieties

In [38], the authors asked for a general upper bound on the number of rational points of a (possibly reducible) equidimensional variety $X \subseteq \mathbf{P}^n$ of dimension d and degree δ. They conjectured that

$$|X(\mathbf{F}_q)| \leq \delta(\pi_d - \pi_{2d-n}) + \pi_{2d_n}, \tag{1}$$

where for all positive integer ℓ, π_ℓ is defined as the number of rational points of the projective space of dimension ℓ over $\mathbf{F}_q$. That is to say, $\pi_\ell = \frac{q^{\ell+1}-1}{q-1}$.

By combining algebraic geometric methods with a combinatorial method of double counting, A. Couvreur proved this conjecture [32] and got a more general upper bound on the number of rational points of arbitrary varieties (possibly non-equidimensional). In addition, he proved that (1) is sharp by providing examples of varieties reaching this bound.

6.7. New families of fast elliptic curves

B. Smith has pioneered the use of mod-p reductions of Q-curves to produce elliptic curves with efficient scalar multiplication algorithms—which translates into faster encryption, decryption, signing, and signature verification operations on these curves. A theoretical article was presented at ASIACRYPT 2013 [7], and a longer version was submitted (upon invitation) to the Journal of Cryptology. The theory was put into practice in collaboration with Craig Costello (Microsoft Research) and Huseyin Hisil (Yasar University). Their resulting publicly available implementation, which represents the state of the art in constant-time (side-channel conscious) elliptic curve scalar multiplication on 64-bit Intel platforms at the 128-bit security level, can carry out a constant-time scalar multiplication in 145k cycles on Ivy Bridge architectures. This work appeared in EUROCRYPT 2014 [21].

6.8. New results for solving the discrete logarithm problem

Recent results of R. Barbulescu, P. Gaudry, A. Joux, and E. Thomé seem to indicate that solving the discrete logarithm problem over finite fields of small characteristic is easier than was precedently thought. F. Morain and A. Guillevic, joined by R. Barbulescu and P. Gaudry, embarked on an attempt to assess the security of the discrete logarithm problem in a closely related context: that of finite fields with large characteristic and small degree. Improving on the methods of A. Joux, R. Lercier and others, they found new algorithms to select polynomials for the Number Field Sieve – the algorithm of choice in this setting. Moreover, a clever study of the algebraic properties of the fields used (e.g., algebraic units), enabled them to break the world record for the case of $GF(p^2)$, soon to be followed by new cases. This work is described in [31], and part of it is currently submitted.

6.9. Quantum Integer Factorization

Together with two researchers in quantum physics (F. Grosshans and T. Lawson), F. Morain and B. Smith have been working on the number theoretical postprocessing in Shor's algorithm. A preprint is being written.

LFANT Project-Team

5. New Results

5.1. Highlights of the Year

Aurel Page has defended his PhD thesis on *Méthodes explicites pour les groupes arithmétiques* [12] in July 2014. Nicolas Mascot has defended his PhD thesis on *Computing modular Galois representations* [11], in July 2014.

5.2. Class groups and other invariants of number fields

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

In [21], P. Lezowski describes the explicit computation of the Euclidean minimum of a number field. It has been published in Mathematics of Computation.

Ohno and Nakagawa have proved, relations between the counting functions of certain cubic fields. These relations may be viewed as complements to the Scholz reflection principle, and Ohno and Nakagawa deduced them as consequences of 'extra functional equations' involving the Shintani zeta functions associated to the prehomogeneous vector space of binary cubic forms. In [26], Henri Cohen, Simon Rubinstein-Salzedo and Frank Thorne generalize their result by proving a similar identity relating certain degree fields with Galois groups D and F respectively, for any odd prime, and in particular we give another proof of the Ohno–Nakagawa relation without appealing to binary cubic forms.

The article [16] by H. Cohen and F. Thorne, H. Cohen on Dirichlet series associated to cubic fields with given resolvent has been published. This article gives an explicit formula for the Dirichlet series $\sum_K |\Delta(K)|^{-s}$, where the sum is over isomorphism classes of all cubic fields whose quadratic resolvent field is isomorphic to a fixed quadratic field k.

This work is extended in [15] where H. Cohen give efficient numerical methods for counting exactly the number of D_ℓ number fields of degree ℓ with given quadratic resolvent, for calculating the constants occurring in their asymptotic expansions, and give tables for typical cases.

5.3. Number and function fields

Participants: Jean-Marc Couveignes, Karim Belabas.

In the article [29], J. Brau study the growth of the Galois invariants of the p-Selmer group of an elliptic curve in a degree p Galois extension. He shows that this growth is determined by certain local cohomology groups and determine necessary and sufficient conditions for these groups to be trivial.

In the article [30] written with J. Nathan, J. Brau study the modular curve $X'(6)$ of level 6 defined over $\mathbb{Q}$ whose $\mathbb{Q}$-rational points correspond to j-invariants of elliptic curves E over $\mathbb{Q}$ for which $\mathbb{Q}(E[2])$ is a subfield of $\mathbb{Q}(E[3])$. They characterize the j-invariants of elliptic curves with this property by exhibiting an explicit model of $X'(6)$. $X'(6)(\mathbb{Q})$ gives an infinite family of examples of elliptic curves with non-abelian "entanglement fields," which is relevant to the systematic study of correction factors of various conjectural constants for elliptic curves over $\mathbb{Q}$.

5.4. Quaternion algebras

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

In the article [14] written with J. Chaubert, J.-P. Cerri and P. Lezowski study totally indefinite Euclidean quaternion fields over a number field K, that is to say where no infinite place is ramified. Relying on some generalisation of Hasse–Schilling–Maaß Norm Theorem, they prove that the Euclidean property of K implies the Euclidean property of any totally indefinite Euclidean quaternion field over K. Conversely, they provide the complete list of norm-Euclidean and totally indefinite quaternion fields over an imaginary quadratic number field. In particular, the article exhibits a totally indefinite and norm-Euclidean quaternion field over a non-Euclidean number field. This provides an answer to a question by Eichler. The proofs are both theoretic and algorithmic. The article has been published in Acta Arithmetica.

Deciding whether an ideal of a number field is principal and finding a generator is a fundamental problem with many applications in computational number theory. In the article [25] gives a an algorithm for indefinite quaternion algebras by reducing the decision problem to that in the underlying number field. It also gives an heuristically subexponential algorithm for finding a generator.

5.5. Complex multiplication and modularity

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

A. Enge and E. Thomé describe in [20] a quasi-linear algorithm for computing Igusa class polynomials of Jacobians of genus 2 curves via complex floating-point approximations of their roots. After providing an explicit treatment of the computations in quartic CM fields and their Galois closures, they pursue an approach due to Dupont for evaluating θ-constants in quasi-linear time using Newton iterations on the Borchardt mean. They report on experiments with the implementation CMH and present an example with class number 20016.

In [34] E. Milio explains how to generalise the work of Régis Dupont for computing modular polynomials in dimension 2 to invariants derived from theta constants. Modular polynomials have many applications. In particular, they could speed up the CRT-algorithm to compute class fields of degree 4 CM-fields which would lead to faster algorithms to construct cryptographically secure Jacobians of hyperelliptic curves. They are also used to compute graphs of isogenies. This paper presents how to compute modular polynomials and the polynomials computed and then it proves some of their properties.

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 [19]. The result is a new infinite family of generators for ring class fields, usable to determine complex multiplication curves. They 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.

N. Mascot has continued his work on computing Galois representations attached to Jacobians of modular curves. He has given tables of modular Galois representations in [33] obtained using the algorithm of [39]. He has computed Galois representations modulo primes up to 31 for the first time. In particular, he has computed the representations attached to a newform with non-rational (but of course algebraic) coefficients, which had never been done before. These computations take place in the Jacobians of modular curves of genus up to 26.

5.6. Elliptic curve and Abelian varieties cryptology

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

In [27] J.-M. Couveignes and T. Ezome show how to efficiently evaluate functions, including Weil functions and canonical Theta functions, on Jacobian varieties and their quotients. They deduce a quasi-optimal algorithm to compute (l, l) isogenies between Jacobians of genus two curves, using a compact representation and differential characterisation of isogenies in this context. This work has been submitted to the LMS Journal of Computation and Mathematics.

The paper [18] by J.-M. Couveignes and R. Lercier describing the problem of parameterisations by radicals of low genus algebraic curves has been accepted in *Advances in mathematics of communications*.

In [31] D. Lubicz and D. Robert explain how to improve the arithmetic of Abelian and Kummer varieties. The speed of the arithmetic is a crucial factor in the performance of abelian varieties based cryptosystem. Depending on the cryptographic application, the speed record holder are elliptic curves (in the Edwards model) or the Kummer surface of an hyperelliptic curves of genus 2 (in the level 2 theta model). One drawback of the Kummer surface is that only scalar multiplications are available, which may be a drawback in certain cryptographic protocols. The previous known models to work on the Jacobian rather than the Kummer surface (Mumford coordinates or theta model of level 4) are too slow and not competitive with Elliptic Curves. This paper explains how to use geometric properties (like projective normality) to speed up the arithmetic. In particular it introduces a novel addition algorithm on Kummer varieties (compatible additions), and use it to enhance multi-exponentiations in Kummer varieties and to obtain new models of abelian surfaces where the scalar multiplication is as fast as on the Kummer surface.

In [32] (which has been accepted at LMS Journal of Computation and Mathematics), D. Lubicz and D. Robert explain how to compute isogenies between abelian varieties given algebraic equation of the kernel. The previous algorithms to compute isogenies between abelian varieties needed the coordinates of generators of the kernel. One drawback was that even if the kernel is rational, these generators may live in extension of large degree, especially for Abelian varieties defined over a number field rather than a finite field. This paper combines the use of formal coordinates together with a normalisation alongs linear subspaces of the kernel rather than the whole kernel to derive an algorithm which is quasi-optimal if the degree of the isogeny is ℓ^g, for ℓ congruent to 1 modulo 4.

This article expands the article [17] by D. Cosset and D. Robert about the computation of (ℓ, ℓ)-isogenies in dimension 2 which has been published in Mathematics of Computation.

5.7. Pairings

Participants: Andreas Enge, Damien Robert.

The article [22] by D. Lubicz and D. Robert explaining how to compute optimal pairings on abelian varieties described by their theta models has been accepted for publication at Journal of Symbolic Computation.

In [24], A. Enge and J. Milan report on the APIP implementation of cryptographic pairings on elliptic curves in PARI/GP. For security levels equivalent to the different AES flavours, they exhibit suitable curves in parametric families and show that optimal ate and twisted ate pairings exist and can be efficiently evaluated. They provide a correct description of Miller's algorithm for signed binary expansions such as the NAF and extend a recent variant due to Boxall et al. to addition-subtraction chains. They analyse and compare several algorithms proposed in the literature for the final exponentiation. Finally, they give recommendations on which curve and pairing to choose at each security level.

POLSYS Project-Team

6. New Results

6.1. Highlights of the Year

Jointly with Univ. Of Kaiserslautern (C. Eder), we have released a new open source C library for linear algebra dedicated to Gröbner bases computations (see http://www-polsys.lip6.fr/~jcf/Software/index.html). This new library opens the door to high performance applications

- The library is specialized in reducing matrices generated during Gröbner bases computations. Optimizing this reduction step is crucial for the overall computation.
- Our approach takes even more advantage of the very special structure (quasi unit-triangular sparse matrices with patterns in the data)
- We also reduce the number of operations, in a parallel friendly fashion, by changing the order of the operations in the elimination.
- We present experimental results for sequential and parallel computations on NUMA architectures. We also get good scaling up until 32 (non hyper-threaded) cores: we have speed-ups around 14 or 16.

6.2. Fundamental algorithms and structured polynomial systems

6.2.1. Sparse Gröbner Bases

Sparse elimination theory is a framework developped during the last decades to exploit monomial structures in systems of Laurent polynomials. Roughly speaking, this amounts to computing in a *semigroup algebra*, *i.e.* an algebra generated by a subset of Laurent monomials. In order to solve symbolically sparse systems, we introduce *sparse Gröbner bases*, an analog of classical Gröbner bases for semigroup algebras, and we propose sparse variants of the F_5 and FGLM algorithms to compute them.

In the case where the generating subset of monomials corresponds to the points with integer coordinates in a normal lattice polytope $\mathcal{P} \subset \mathbb{R}^n$ and under regularity assumptions, we prove in [19] complexity bounds which depend on the combinatorial properties of $\mathcal{P}$. These bounds yield new estimates on the complexity of solving 0-dim systems where all polynomials share the same Newton polytope (*unmixed case*). For instance, we generalize the bound $\min(n_1, n_2) + 1$ on the maximal degree in a Gröbner basis of a 0-dim. Bilinear system with blocks of variables of sizes (n_1, n_2) to the multihomogeneous case: $n + 2 - \max_i(\lceil (n_i+1)/d_i \rceil)$. We also propose a variant of Fröberg's conjecture which allows us to estimate the complexity of solving overdetermined sparse systems.

Moreover, our prototype "proof-of-concept" implementation shows large speed-ups (more than 100 for some examples) compared to optimized (classical) Gröbner bases software.

6.2.2. Gröbner bases for weighted homogeneous systems

Solving polynomial systems arising from applications is frequently made easier by the structure of the systems. Weighted homogeneity (or quasi-homogeneity) is one example of such a structure: given a system of weights $W = (w_1, \cdots, w_n)$, W-homogeneous polynomials are polynomials which are homogeneous w.r.t the weighted degree $\deg_W(X_1^{\alpha_1}, \cdots, X_n^{\alpha_n}) = \sum w_i \alpha_i$.

Gröbner bases for weighted homogeneous systems can be computed by adapting existing algorithms for homogeneous systems to the weighted homogeneous case. In [29], we show that in this case, the complexity estimate for Algorithm F5 $\left(\binom{n+d_{\max}-1}{d_{\max}}^\omega\right)$ can be divided by a factor $(\prod w_i)^\omega$. For zero-dimensional systems, the complexity of Algorithm FGLMnD^ω (where D is the number of solutions of the system) can be divided by the same factor $(\prod w_i)^\omega$. Under genericity assumptions, for zero-dimensional weighted homogeneous systems of W-degree $(d_1, \cdots, d_n)$, these complexity estimates are polynomial in the weighted Bézout bound $\prod_{i=1}^n d_i / \prod_{i=1}^n w_i$.

Furthermore, the maximum degree reached in a run of Algorithm F5 is bounded by the weighted Macaulay bound $\sum (d_i - w_i) + w_n$, and this bound is sharp if we can order the weights so that $w_n = 1$. For overdetermined semi-regular systems, estimates from the homogeneous case can be adapted to the weighted case.

We provide some experimental results based on systems arising from a cryptography problem and from polynomial inversion problems. They show that taking advantage of the weighted homogeneous structure yields substantial speed-ups, and allows us to solve systems which were otherwise out of reach.

6.2.3. Computing necessary integrability conditions for planar parametrized homogeneous potentials

Let $V \in \mathbb{Q}(i)(\mathbf{a}_1, \cdots, \mathbf{a}_n)(\mathbf{q}_1, \mathbf{q}_2)$ be a rationally parametrized planar homogeneous potential of homogeneity degree $k \neq -2, 0, 2$. In [12], we design an algorithm that computes polynomial *necessary* conditions on the parameters $(\mathbf{a}_1, \cdots, \mathbf{a}_n)$ such that the dynamical system associated to the potential V is integrable. These conditions originate from those of the Morales-Ramis-Simó integrability criterion near all Darboux points and make use of Gröbner bases algorithms. The implementation of the algorithm allows to treat applications that were out of reach before, for instance concerning the non-integrability of polynomial potentials up to degree 9. Another striking application is the first complete proof of the non-integrability of the *collinear three body problem*.

6.3. Solving Polynomial Systems over the Reals and Applications

6.3.1. Exact algorithms for polynomial optimization

Let $f, f_1, ..., f_s$ be n-variate polynomials with rational coefficients of maximum degree D and let V be the set of common complex solutions of $\mathbf{F} = (f_1, ..., f_s)$. In [7], we give an algorithm which, up to some regularity assumptions on $\mathbf{F}$, computes an *exact* representation of the global infimum $f^{\star}$ of the restriction of the map $x \to f(x)$ to $V \cap \mathbb{R}^n$, i.e. a univariate polynomial vanishing at $f^{\star}$ and an isolating interval for $f^{\star}$. Furthermore, it decides whether $f^{\star}$ is reached and if so, it returns $x^{\star} \in V \cap \mathbb{R}^n$ such that $f(x^{\star}) = f^{\star}$.

This algorithm is *probabilistic*. It makes use of the notion of polar varieties. Its complexity is essentially *cubic* in $(sD)^n$ and linear in the complexity of evaluating the input. This fits within the best known *deterministic* complexity class $D^{O(n)}$.

We report on some practical experiments of a first implementation that is available as a MAPLE package. It appears that it can tackle global optimization problems that were unreachable by previous exact algorithms and can manage instances that are hard to solve with purely numeric techniques. As far as we know, even under the extra genericity assumptions on the input, it is the first probabilistic algorithm that combines practical efficiency with good control of complexity for this problem.

It is known that point searching in basic semialgebraic sets and the search for globally minimal points in polynomial optimization tasks can be carried out using $(s\,d)^{O(n)}$ arithmetic operations, where n and s are the numbers of variables and constraints and d is the maximal degree of the polynomials involved.
Subject to certain conditions, we associate in [2] to each of these problems an intrinsic system degree which becomes in worst case of order $(n\,d)^{O(n)}$ and which measures the intrinsic complexity of the task under consideration.
We design non-uniform deterministic or uniform probabilistic algorithms of intrinsic, quasi-polynomial complexity which solve these problems.

6.3.2. Algorithms for answering connectivity queries

Let $\mathbf{R}$ be a real closed field and $\mathbf{D} \subset \mathbf{R}$ an ordered domain. In [4], we give an algorithm that takes as input a polynomial $Q \in \mathbf{D}[X_1, ..., X_k]$, and computes a description of a roadmap of the set of zeros, $\mathrm{Zer}(Q, \mathbf{R}^k)$, of Q in $\mathbf{R}^k$. The complexity of the algorithm, measured by the number of arithmetic operations in the ordered domain $\mathbf{D}$, is bounded by $D^{O(k\sqrt{k})}$, where $D = \deg(Q) \geq 2$. As a consequence, there exist algorithms for computing the number of semi-algebraically connected components of a real algebraic set, $Z(Q, \mathbf{R}^n)$, whose complexity is also bounded by $D^{O(n\sqrt{n})}$, where $D = \deg(Q) \geq 2$. The best previously known algorithm for constructing a roadmap of a real algebraic subset of $\mathbf{R}^n$ defined by a polynomial of degree D has complexity $D^{O(n^2)}$.

In [36], we provide a probabilistic algorithm which computes roadmaps for smooth and bounded real algebraic sets such that the output size and the running time are polynomial in $(nD)^{n\log(n)}$. More precisely, the running time of the algorithm is essentially subquadratic in the output size. Even under these extra assumptions, it is the first roadmap algorithm with output size and running time polynomial in $(nD)^{n\log(n)}$.

6.3.3. Nearly Optimal Refinement of Real Roots of a Univariate Polynomial

In [33], we consider the following problem. We assume that a real square-free polynomial A has a degree d, a maximum coefficient bitsize τ and a real root lying in an isolating interval and having no nonreal roots nearby (we quantify this assumption). Then we combine the Double Exponential Sieve algorithm (also called the Bisection of the Exponents), the bisection, and Newton iteration to decrease the width of this inclusion interval by a factor of $t = 2^L$. The algorithm has Boolean complexity $O(d^2\tau + dL)$. This substantially decreases the known bound $O(d^3 + d^2L)$. Furthermore we readily extend our algorithm to support the same complexity bound for the refinement of r real roots, for any $r \leq d$, by incorporating the known efficient algorithms for multipoint polynomial evaluation. The main ingredient for the latter ones is an efficient algorithm for (approximate) polynomial division; we present a variation based on structured matrices computation with quasi-optimal Boolean complexity.

6.3.4. Accelerated Approximation of the Complex Roots of a Univariate Polynomial

Highly efficient and even nearly optimal algorithms have been developed for the classical problem of univariate polynomial root-finding, but this is still an area of active research. By combining some powerful techniques developed in this area we devise in [20] new nearly optimal algorithms, whose substantial merit is their simplicity, important for the implementation.

6.3.5. Nearly Optimal Computations with Structured Matrices

In [21], we estimate the Boolean complexity of multiplication of structured matrices by a vector and the solution of nonsingular linear systems of equations with these matrices. We study four basic most popular classes, that is, Toeplitz, Hankel, Cauchy and Vandermonde matrices, for which the cited computational problems are equivalent to the task of polynomial multiplication and division and polynomial and rational multipoint evaluation and interpolation. The Boolean cost estimates for the latter problems have been obtained by Kirrinnis, except for rational interpolation, which we provide now. All known Boolean cost estimates for these problems rely on using Kronecker product. This implies the d-fold precision increase for the d-th degree output, but we avoid such an increase by relying on distinct techniques based on employing FFT. Furthermore we simplify the analysis and make it more transparent by combining the representation of our tasks and algorithms in terms of both structured matrices and polynomials and rational functions. This also enables further extensions of our estimates to cover Trummer's important problem and computations with the popular classes of structured matrices that generalize the four cited basic matrix classes.

6.3.6. Bounds for the Condition Number for Polynomials with Integer Coefficients

In [31], we consider the problem of bounding the condition number of the roots of univariate polynomials and polynomial systems, when the input polynomials have integer coefficients. We also introduce an aggregate version of the condition numbers and we prove bounds of the same order of magnitude as in the case of the condition number of a single root.

6.4. Solving Systems in Finite Fields, Applications in Cryptology and Algebraic Number Theory

6.4.1. Polynomial-Time Algorithms for Quadratic Isomorphism of Polynomials: The Regular Case

Let $\mathbf{f} = (f_1, ..., f_m)$ and $\mathbf{g} = (g_1, ..., g_m)$ be two sets of $m \geq 1$ nonlinear polynomials in $\mathbb{K}[x_1, ..., x_n]$ ($\mathbb{K}$ being a field). In [25], we consider the computational problem of finding – if any – an invertible transformation on the variables mapping $\mathbf{f}$ to $\mathbf{g}$. The corresponding equivalence problem is known as *Isomorphism of Polynomials with one Secret* (IP1S) and is a fundamental problem in multivariate cryptography. Amongst its applications, we can cite Graph Isomorphism (GI) which reduces to equivalence of cubic polynomials with respect to an invertible linear change of variables, according to Agrawal and Saxena. The main result of our work is a randomized polynomial-time algorithm for solving IP1S for quadratic instances, a particular case of importance in cryptography.

To this end, we show that IP1S for quadratic polynomials can be reduced to a variant of the classical module isomorphism problem in representation theory. We show that we can essentially *linearize* the problem by reducing quadratic-IP1S to test the orthogonal simultaneous similarity of symmetric matrices; this latter problem was shown by Chistov, Ivanyos and Karpinski (ISSAC 1997) to be equivalent to finding an invertible matrix in the linear space $\mathbb{K}^{n\times n}$ of $n \times n$ matrices over $\mathbb{K}$ and to compute the square root in a certain representation in a matrix algebra. While computing square roots of matrices can be done efficiently using numerical methods, it seems difficult to control the bit complexity of such methods. However, we present exact and polynomial-time algorithms for computing a representation of the square root of a matrix in $\mathbb{K}^{n\times n}$, for various fields (including finite fields), as a product of two matrices. Each coefficient of these matrices lie in an extension field of $\mathbb{K}$ of polynomial degree. We then consider #IP1S, the counting version of IP1S for quadratic instances. In particular, we provide a (complete) characterization of the automorphism group of homogeneous quadratic polynomials. Finally, we also consider the more general *Isomorphism of Polynomials* (IP) problem where we allow an invertible linear transformation on the variables *and* on the set of polynomials. A randomized polynomial-time algorithm for solving IP when $\mathbf{f} = (x_1^d, ..., x_n^d)$ is presented. From an algorithmic point of view, the problem boils down to factoring the determinant of a linear matrix (*i.e.* a matrix whose components are linear polynomials). This extends to IP a result of Kayal obtained for PolyProj.

6.4.2. A Polynomial-Time Key-Recovery Attack on MQQ Cryptosystems

In [15], we investigate the security of the family of MQQ public key cryptosystems using multivariate quadratic quasigroups (MQQ). These cryptosystems show especially good performance properties. In particular, the MQQ-SIG signature scheme is the fastest scheme in the ECRYPT benchmarking of cryptographic systems (eBACS). We show that both the signature scheme MQQ-SIG and the encryption scheme MQQ-ENC, although using different types of MQQs, share a common algebraic structure that introduces a weakness in both schemes. We use this weakness to mount a successful polynomial time key-recovery attack. Our key-recovery attack finds an equivalent key using the idea of so-called good keys that reveals the structure gradually. In the process we need to solve a MinRank problem that, because of the structure, can be solved in polynomial-time assuming some mild algebraic assumptions. We highlight that our theoretical results work in characteristic 2 which is known to be the most difficult case to address in theory for MinRank attacks. Also, we emphasize that our attack works without any restriction on the number of polynomials removed from the public-key, that is, using the minus modifier. This was not the case for previous MinRank like-attacks against MQ schemes. From a practical point of view, we are able to break an MQQ-SIG instance of 80 bits security in less than 2 days, and one of the more conservative MQQ-ENC instances of 128 bits security in little bit over 9 days. Altogether, our attack shows that it is very hard to design a secure public key scheme based on an easily invertible MQQ structure.

6.4.3. Algebraic Cryptanalysis of a Quantum Money Scheme – The Noise-Free Case

In [13], we investigate the Hidden Subspace Problem (HSP_q) over $\mathbb{F}_q$:

Input : $p_1, ..., p_m, q_1, ..., q_m \in \mathbb{F}_q[x_1, ..., x_n]$ of degree $d \geq 3$ (and $n \leq m \leq 2n$).
Find : a subspace $A \subset \mathbb{F}_q^n$ of dimension $n/2$ (n is even) such that

$$p_i(A) = 0 \ \forall i \in \{1, ..., m\} \text{ and } q_j(A^\perp) = 0 \ \forall j \in \{1, ..., m\},$$

where $A^\perp$ denotes the orthogonal complement of A with respect to the usual scalar product in $\mathbb{F}_q$.

This problem underlies the security of the first public-key quantum money scheme that is proved to be cryptographically secure under a non quantum but classic hardness assumption. This scheme was proposed by S. Aaronson and P. Christiano at STOC'12. In particular, it depends upon the hardness of HSP_2. More generally, Aaronson and Christiano left as an open problem to study the security of the scheme for a general field $\mathbb{F}_q$. We present a randomized polynomial-time algorithm that solves the HSP_q for $q > 2$ with success probability $\approx 1 - 1/q$. So, the quantum money scheme extended to $\mathbb{F}_q$ is not secure. Finally, based on experimental results and a structural property of the polynomials that we prove, we conjecture that there is also a randomized polynomial-time algorithm solving the HSP_2 with high probability. To support our theoretical results, we also present several experimental results confirming that our algorithms are very efficient in practice. We emphasize that Aaronson and Christiano propose a non-noisy and a noisy version of the public-key quantum money scheme. The noisy version of the quantum money scheme remains secure.

6.4.4. Algebraic Algorithms for LWE Problems

In [23], we analyse the complexity of algebraic algorithms for solving systems of linear equations with *noise*. Such systems arise naturally in the theory of error-correcting codes as well as in computational learning theory. More recently, linear systems with noise have found application in cryptography. The *Learning with Errors* (LWE) problem has proven to be a rich and versatile source of innovative cryptosystems, such as fully homomorphic encryption schemes. Despite the popularity of the LWE problem, the complexity of algorithms for solving it is not very well understood, particularly when variants of the original problem are considered. Here, we focus on and generalise a particular method for solving these systems, due to Arora & Ge, which reduces the problem to non-linear but noise-free system solving. Firstly, we provide a refined complexity analysis for the original Arora-Ge algorithm for LWE. Secondly, we study the complexity of applying algorithms for computing Gröbner basis, a fundamental tool in computational commutative algebra, to solving Arora-Ge-style systems of non-linear equations. We show positive and negative results. On the one hand, we show that the use of Gröbner bases yields an exponential speed-up over the basic Arora-Ge approach. On the other hand, we give a negative answer to the natural question whether the use of such techniques can yield a subexponential algorithm for the LWE problem. Under a mild algebraic assumption, we show that it is highly unlikely that such an improvement exists. We also consider a variant of LWE known as BinaryError-LWE introduced by Micciancio and Peikert recently. By combining Gröbner basis algorithms with the Arora-Ge modelling, we show under a natural algebraic assumption that BinaryError-LWE can be solved in subexponential time as soon as the number of samples is quasi-linear. We also derive precise complexity bounds for BinaryError-LWE with $m = O(n)$, showing that this new approach yields better results than best currently-known generic (exact) CVP solver as soon as $m/n \geq 6.6$. More generally, our results provide a good picture of the hardness degradation of BinaryError-LWE for various number of samples.. This addresses an open question from Micciancio and Peikert. Whilst our results do not contradict the hardness results obtained by Micciancio and Peikert, they should rule out BinaryError-LWE for many cryptographic applications. The results in this work depend crucially on the assumption the algebraic systems considered systems are not easier and not harder to solve than a random system of equations. We have verified experimentally such hypothesis. We also have been able to prove formally the assumptions is several restricted situations. We emphasize that these issues are highly non-trivial since proving our assumptions in full generality would allow to prove a famous conjecture in commutative algebra known as Fröberg's Conjecture.

6.4.5. Practical Cryptanalysis of a Public-Key Encryption Scheme Based on New Multivariate Quadratic Assumptions

In [10], we investigate the security of a public-key encryption scheme introduced by Huang, Liu and Yang (HLY) at PKC'12. This new scheme can be provably reduced to the hardness of solving a set of quadratic equations whose coefficients of highest degree are chosen according to a discrete Gaussian distributions. The other terms being chosen uniformly at random. Such a problem is a variant of the classical problem of solving a system of non-linear equations (PoSSo), which is known to be hard for random systems. The main hypothesis of Huang, Liu and Yang is that their variant is not easier than solving PoSSo for random instances. In this paper, we disprove this hypothesis. To this end, we exploit the fact that the new problem proposed by Huang, Liu and Yang reduces to an easy instance of the Learning With Errors (LWE) problem. The main contribution of this paper is to show that security and efficiency are essentially incompatible for the HLY proposal. That is, one cannot find parameters which yield a secure and a practical scheme. For instance, we estimate that a public-key of at least 1.03 GB is required to achieve 80-bit security against the simplest of our attacks. As a proof of concept, we present 3 practical attacks against all the parameters proposed by Huang, Liu and Yang. With the most efficient attack, we have been able to recover the private-key in roughly 5 minutes for the first challenge (i.e. Case 1) proposed by HLY and less than 30 minutes for the second challenge (i.e. Case 2).

6.4.6. Lazy Modulus Switching for the BKW Algorithm on LWE

Some recent constructions based on LWE do not sample the secret uniformly at random but rather from some distribution which produces small entries. The most prominent of these is the binary-LWE problem where the secret vector is sampled from $\{0,1\}^*$ or $\{-1,0,1\}^*$. In [9], we present a variant of the BKW algorithm for binary-LWE and other small secret variants and show that this variant reduces the complexity for solving binary-LWE. We also give estimates for the cost of solving binary-LWE instances in this setting and demonstrate the advantage of this BKW variant over standard BKW and lattice reduction techniques applied to the SIS problem. Our variant can be seen as a combination of the BKW algorithm with a lazy variant of modulus switching which might be of independent interest.

In [1], 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 compare with alternative approaches based on lattice reduction. As a result, we provide new upper bounds for the concrete hardness of these LWE-based schemes. Rather surprisingly, it appears that BKW algorithm outperforms known estimates for lattice reduction algorithms starting in dimension $n \approx 250$ when LWE is reduced to SIS. However, this assumes access to an unbounded number of LWE samples.

6.4.7. Algebraic Attack against Variants of McEliece with Goppa Polynomial of a Special Form

In [17], we present a new algebraic attack against some special cases of Wild McEliece Incognito, a generalization of the original McEliece cryptosystem. This attack does not threaten the original McEliece cryptosystem. We prove that recovering the secret key for such schemes is equivalent to solving a system of polynomial equations whose solutions have the structure of a usual vector space. Consequently, to recover a basis of this vector space, we can greatly reduce the number of variables in the corresponding algebraic system. From these solutions, we can then deduce the basis of a GRS code. Finally, the last step of the cryptanalysis of those schemes corresponds to attacking a McEliece scheme instantiated with particular GRS codes (with a polynomial relation between the support and the multipliers) which can be done in polynomial-time thanks to a variant of the Sidelnikov-Shestakov attack. For Wild McEliece & Incognito, we also show that solving the corresponding algebraic system is notably easier in the case of a non-prime base field $\mathbb{F}_q$. To support our theoretical results, we have been able to practically break several parameters defined over a non-prime base field $q \in \{9, 16, 25, 27, 32\}, t < 7$, extension degrees $m \in \{2, 3\}$, security level up to 2^{129} against information set decoding in few minutes or hours.

6.4.8. Folding Alternant and Goppa Codes with Non-Trivial Automorphism Groups

The main practical limitation of the McEliece public-key encryption scheme is probably the size of its key. A famous trend to overcome this issue is to focus on subclasses of alternant/Goppa codes with a non trivial automorphism group. Such codes display then *symmetries* allowing compact parity-check or generator matrices. For instance, a key-reduction is obtained by taking *quasi-cyclic* (QC) or *quasi-dyadic* (QD) alternant/Goppa codes. We show in [6], [18], [28] that the use of such *symmetric* alternant/Goppa codes in cryptography introduces a fundamental weakness. It is indeed possible to reduce the key-recovery on the original symmetric public-code to the key-recovery on a (much) smaller code that has not anymore symmetries. This result is obtained thanks to a new operation on codes called *folding* that exploits the knowledge of the automorphism group. This operation consists in adding the coordinates of codewords which belong to the same orbit under the action of the automorphism group. The advantage is twofold: the reduction factor can be as large as the size of the orbits, and it preserves a fundamental property: folding the dual of an alternant (*resp*. Goppa) code provides the dual of an alternant (*resp*. Goppa) code. A key point is to show that all the existing constructions of alternant/Goppa codes with symmetries follow a common principal of taking codes whose support is globally invariant under the action of affine transformations (by building upon prior works of T. Berger and A. Dür). This enables not only to present a unified view but also to generalize the construction of QC, QD and even *quasi-monoidic* (QM) Goppa codes. All in all, our results can be harnessed to boost up any key-recovery attack on McEliece systems based on symmetric alternant or Goppa codes, and in particular algebraic attacks.

6.4.9. Rounding and Chaining LLL: Finding Faster Small Roots of Univariate Polynomial Congruences

In a seminal work at EUROCRYPT '96, Coppersmith showed how to find all small roots of a univariate polynomial congruence in polynomial time: this has found many applications in public-key cryptanalysis and in a few security proofs. However, the running time of the algorithm is a high-degree polynomial, which limits experiments: the bottleneck is an LLL reduction of a high-dimensional matrix with extra-large coefficients. We present in [11] the first significant speedups over Coppersmith's algorithm. The first speedup is based on a special property of the matrices used by Coppersmith's algorithm, which allows us to provably speed up the LLL reduction by rounding, and which can also be used to improve the complexity analysis of Coppersmith's original algorithm. The exact speedup depends on the LLL algorithm used: for instance, the speedup is asymptotically quadratic in the bit-size of the small-root bound if one uses the Nguyen-Stehlé L2 algorithm. The second speedup is heuristic and applies whenever one wants to enlarge the root size of Coppersmith's algorithm by exhaustive search. Instead of performing several LLL reductions independently, we exploit hidden relationships between these matrices so that the LLL reductions can be somewhat chained to decrease the global running time. When both speedups are combined, the new algorithm is in practice hundreds of times faster for typical parameters.

6.4.10. Symmetrized summation polynomials: using small order torsion points to speed up elliptic curve index calculus

Decomposition-based index calculus methods are currently efficient only for elliptic curves E defined over non-prime finite fields of very small extension degree n. This corresponds to the fact that the Semaev summation polynomials, which encode the relation search (or "sieving"), grows over-exponentially with n. Actually, even their computation is a first stumbling block and the largest Semaev polynomial ever computed is the 6-th. Following ideas from Faugère, Gaudry, Huot and Renault, our goal is to use the existence of small order torsion points on E to define new summation polynomials whose symmetrized expressions are much more compact and easier to compute. This setting allows to consider smaller factor bases, and the high sparsity of the new summation polynomials provides a very efficient decomposition step. In [16], the focus is on 2-torsion points, as it is the most important case in practice. We obtain records of two kinds: we successfully compute up to the 8-th symmetrized summation polynomial and give new timings for the computation of relations with degree 5 extension fields.

6.4.11. Sub-cubic Change of Ordering for Gröner Basis: A Probabilistic Approach

The usual algorithm to solve polynomial systems using Gröbner bases consists of two steps: first computing the DRL Gröbner basis using the F5 algorithm then computing the LEX Gröbner basis using a change of ordering algorithm. When the Bézout bound is reached, the bottleneck of the total solving process is the change of ordering step. For 20 years, thanks to the FGLM algorithm the complexity of change of ordering is known to be cubic in the number of solutions of the system to solve. We show in [14] that, in the generic case or up to a generic linear change of variables, the multiplicative structure of the quotient ring can be computed with no arithmetic operation. Moreover, given this multiplicative structure we propose a change of ordering algorithm for Shape Position ideals whose complexity is polynomial in the number of solutions with exponent ω where $2 \leq \omega < 2.3727$ is the exponent in the complexity of multiplying two dense matrices. As a consequence, we propose a new Las Vegas algorithm for solving polynomial systems with a finite number of solutions by using Gröbner basis for which the change of ordering step has a sub-cubic (i.e. with exponent ω) complexity and whose total complexity is dominated by the complexity of the F5 algorithm. In practice we obtain significant speedups for various polynomial systems by a factor up to 1500 for specific cases and we are now able to tackle some instances that were intractable.

SECRET Project-Team

6. New Results

6.1. Highlights of the Year

- Rafael Misoczki's PhD thesis on code-based cryptography (defended in November 2013) has been awarded by the Brazilian Society of Computer Science as the best thesis in computer security.
- *Security analysis of some primitives for authentication and authenticated encryption:* authentication is a major functionality in the vast majority of applications. It is usually implemented by a MAC (message authentication code). The main constructions for MAC are based on hash functions, and include the wide-spread HMAC construction. Gaëtan Leurent, together with Itai Dinur, has presented a new generic attack against HMAC when the underlying hash function follows the Haifa construction. This result points out that the hash function in HMAC has to be chosen very carefully and that some of the main families of hash functions may introduce unexpected weaknesses in the associated MAC. Also, the project-team is involved in a national cryptanalytic effort funded by the ANR which aims at evaluating the security of the recently proposed authenticated encryption schemes.
- *Parallel Repetition of Entangled Games:* In a two-player free game G, two cooperating but non communicating players receive inputs taken from two independent probability distributions. Each of them produces an output and they win the game if they satisfy some predicate on their inputs/outputs. The classical (resp. entangled) value of G is the maximum winning probability when the players are allowed to share classical random bits (resp. a quantum state) prior to receiving their inputs. The n-fold parallel repetition of G consists of n instances of G where the parties receive all the inputs at the same time, produce all the outputs at the same time and must win every instance of G. This work by André Chailloux in collaboration with Giannicola Scarpa establishes that the entangled value of the parallel repetition of G decreases exponentially with n, thereby generalizing to the quantum setting Raz's celebrated parallel repetition theorem which is concerned with the classical value of the game. The main tool for proving this result is the introduction of a new information-theoretic quantity: the superposed information cost.

6.2. Symmetric cryptosystems

Participants: Anne Canteaut, Pascale Charpin, Virginie Lallemand, Gaëtan Leurent, 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 like 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 restricted 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 in 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 authenticated encryption schemes. Most teams in the research community are actually working on the design and on the analysis (cryptanalysis and optimization of the performance) of such primitives.

6.2.1. Block ciphers

Even if the security of the current block cipher standard, AES, is not threatened 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 analyzed. Most of our work in this area is related to an ANR Project named BLOC. Our recent results then mainly concern either the analysis and design of lightweight block ciphers, or the in-depth study of the security of the block cipher standard AES.

Recent results:

- Cryptanalysis of several recently proposed lightweight block ciphers. This includes an attack against the full cipher KLEIN-64 [60], an attack against 8 rounds (out of 12) of PRINCE [48], [77], and an attack against Zorro and its variants [74].
- Formalization and generic improvements of impossible differential cryptanalysis: this type of attacks, even if extensively used, remains not fully understood, and it appears that there are numerous applications where mistakes have been discovered or where the attacks lack optimality. Our work then provides a general framework for impossible differential cryptanalysis including a generic complexity analysis of the optimal attack. Using these advances, we have also presented the best known impossible differential attacks against several ciphers including CLEFIA-128, Camellia, LBlock and Simon [46], [76], [75].
- Design of a new family of block ciphers achieving very good software performance, especially on 8-bit microcontrollers. A nice feature of these ciphers is that they offer an optimal resistance against side-channel attacks in the sense that the cost of Boolean masking is minimized [58].
- Design and study of a new construction for low-latency block ciphers, named *reflection ciphers*, which generalizes the so-called α-reflection property exploited in PRINCE. This construction aims at reducing the implementation overhead of decryption on top of encryption [24]
- Proposal of a new family of distinguishers against AES-based permutations, named *limited-birthday distinguishers*; these distinguishers exploit some some improved rebound techniques. They have been successfully applied to various AES-based primitives including AES, ECHO, Grøstl, LED, PHOTON and Whirlpool [18].
- Analysis of the differential and linear properties of the AES Superbox [65].

6.2.2. Authenticated encryption

A limitation of all classical block ciphers is that they aim at protecting confidentiality only, while most applications need both encryption and authentication. These two functionalities are provided by using a block cipher like the AES together with an appropriate mode of operation. However, it appears that the most widely-used mode of operation for authenticated encryption, AES-GCM, is not very efficient for high-speed networks. Also, the security of the GCM mode completely collapses when an IV is reused. These severe drawbacks have then motivated an international competition named CAESAR, partly supported by the NIST, which has been recently launched in order to define some new authenticated encryption schemes [0]. Our work related to this competition is then two-fold: G. Leurent has participated to the design of a CAESAR candidate named SCREAM. Also, the project-team is involved in a national cryptanalytic effort led by the BRUTUS project funded by the ANR which aims at evaluating the security of all CAESAR candidates.

Recent results:

- Submission of a proposal to the CAESAR competition [88], [67].
- Cryptanalysis of three CAESAR candidates: Wheesht [64], π-cipher [90], LAC [69].

6.2.3. Hash functions and MACS

The international research effort related to the selection of the new hash function standard SHA-3 has led to many important results and to a better understanding of the security offered by hash functions. However, hash functions are used in a huge number of applications with different security requirements, and also form the building-blocks of some other primitives, like MACs. In this context, we have investigated the security of some of these constructions, in order to determine whether some particular constructions for hash functions may affect the security of the associated MACs.

Recent results:

- Improved generic attacks against hash-based MAC, including HMAC, when the hash function follows the Haifa construction [55], [33];
- Attack against Streebog, the new Russian hash function standard: we show that the specific instantiation of the Haifa construction used in Streebog makes it weak against second-preimage attacks [59].

6.2.4. Cryptographic properties and construction of appropriate building blocks

The construction of building blocks which guarantee a high resistance against 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 at the origin of new attacks. This work involves fundamental aspects related to discrete mathematics, cryptanalysis and implementation aspects. Actually, characterizing 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.

[0] http://competitions.cr.yp.to/caesar.html

Recent results:

- Study of the algebraic properties (e.g. the algebraic degree) of the inverses of APN power permutations [19].
- Study of the cryptographic properties, including the degree, the differential uniformity and the size of the image set of permutations of the form $x \mapsto x^s + \gamma \mathsf{Tr}(x^t)$ over a finite field of characteristic two [15]. Since these functions are obtained by slightly modifying a power function, they share similar interesting implementation properties but do not present the weaknesses coming from their structure. In particular, an infinite family of permutations of this form with differential uniformity 4 has been exhibited.
- Definition of an extended criterion for estimating the resistance of a block cipher to differential attacks. Most notably, this new criterion points out the fact that affinely equivalent Sboxes may not provide the same security level regarding differential and linear cryptanalysis. This work emphasizes the role played by the affine permutation of the set of 8-bit words which follows the inverse function in the AES [65].

6.2.5. Symmetric primitives based on lattices

Lattice-based cryptography is an alternative to number-theoretic constructions for public-key cryptography. Lattice-based constructions enjoy a worst-case security reduction to hard lattice problems, and the area is very active, with many new designs offering attractive features.

Recently, this approach has also been used to build symmetric cryptosystems based on lattice problems. While those systems are less efficient than traditional symmetric systems, they are still reasonably efficient, and their security can be related to hard computational problems rather than being only heuristic. In addition, the underlying mathematical structure can offer extra properties such as parallelizability or easy protection against side-channel attacks.

Recent results:

- Design of a family of pseudo-random functions named SPRING which aims to combine the guarantees of security reductions with good performance [44]; implementation of SPRING on FPGA and protection of this hardware implementation against side-channel attacks [47].
- Implementation and side-channel evaluation of the Lapin authentication protocol, based on the LPN problem [57].

6.3. Code-based cryptography

Participants: Julia Chaulet, Adrien Hauteville, Grégory Landais, Nicolas Sendrier, 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 schemes).

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,
- addressing new functionalities, like hashing or symmetric encryption.

Recent results:

- Cryptanalysis of McEliece system based on Wild Goppa codes from a quadratic finite field extension. This polynomial-time structural attack relies on some filtration of nested subcodes which will reveal the secret algebraic description of the underlying secret code [16], [17].
- Structural cryptanalysis of some variants of McEliece scheme based on alternant codes which have a quasi-cyclic or quasi-dyadic generator matrix [86].
- Cryptanalysis of a variant of the McEliece cryptosystem based on Reed-Solomon codes [16].
- Design of a new variant of McEliece using quasi-cyclic Moderate Density Parity Check (MDPC) codes [39].

6.4. Reverse-engineering of communication systems

Participants: Marion Bellard, Nicolas Sendrier, Jean-Pierre Tillich, Audrey Tixier.

To assess the quality of a cryptographic algorithm, it is usually assumed that its specifications are public, as, in accordance with Kerckhoffs principle [0], 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, are 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 Ministry of Defense.

Recent results:

- Reconstruction of the constellation labelling (i.e. used in the modulator of a communication system) in the presence of errors and when the underlying code is convolutional [10].
- Reconstruction of a convolutional code. This reconstruction technique is based on a new method for detecting whether a given binary sequence is a noisy convolutional codeword obtained from an unknown convolutional code [45].
- Reconstruction of the interleaver of a turbo-code from the knowledge of several noisy codewords [63].

6.5. Quantum information theory

Participants: André Chailloux, Anthony Leverrier, Denise Maurice, Jean-Pierre Tillich.

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

The field of Quantum Information and Computation aims at exploiting the laws of quantum physics to manipulate information in radically novel ways. Two main applications come to mind: quantum computers, that offer the promise of solving some problems intractable with classical computers (for instance, factorization); and quantum cryptography, which provides new ways to exchange data in a provably secure fashion.

The main obstacle towards the development of quantum computing is decoherence, a consequence of the interaction of the computer with a noisy environment. We investigate approaches to quantum error-correction as a way to fight against this effect, and we study more particularly some families of quantum error-correcting codes which generalize the best classical codes available today.

Our research also covers quantum cryptography where we study the security of efficient protocols for key distribution or coin flipping, in collaboration with experimental groups. More generally, we investigate how quantum theory severely constrains the action of honest and malicious parties in cryptographic scenarios.

Finally, a promising approach to better understand the possibilities of quantum information consists in studying quantum correlations via the notion of nonlocal games, where different parties need to coordinate to answer some questions, but without communicating. The goal here is to analyze the optimal strategies and to quantify the quantum advantage, i.e. how much sharing an entangled quantum state helps compared to sharing classical randomness.

6.5.1. Quantum codes

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 with fixed non-zero rate and a minimum distance which grows proportionally to the square root of the block-length. This greatly improves the previously best known construction whose minimum distance was logarithmic in the block-length [23].
- Design of a decoding algorithm for the family of quantum codes due to Calderbank, Shor and Steane [84].
- Study of quantum error correcting codes with an iterative decoding algorithm [12].
- Error analysis for Boson Sampling, a simplified model for quantum computation [91].

6.5.2. Quantum cryptography

A recent approach to cryptography takes into account that all interactions occur in a physical world described by the laws of quantum physics. These laws put severe constraints on what an adversary can achieve, and allow for instance to design provably secure key distribution protocols. We study such protocols as well as more general cryptographic primitives such as coin flipping with security properties based on quantum theory.

Recent results:

- Composable security proof for a continuous-variable quantum key distribution protocol with coherent states [92], [71], [70].
- Proof of existence of quantum weak coin flipping with arbitrarily small bias [80].
- Experimental implementation of quantum coin flipping [20].
- Study of connections between quantum encodings, non-locality and quantum cryptography [22].

6.5.3. Quantum correlations and nonlocality

Since the seminal work from Bell in the 60's, it has been known that classical correlations obtained via shared randomness cannot reproduce all the correlations obtained by measuring entangled quantum systems. This impossibility is for instance witnessed by the violation of a Bell inequality and is known under the name of "Quantum Nonlocality". In addition to its numerous applications for quantum cryptography, the study of quantum nonlocality and quantum games has become a central topic in quantum information theory, with the hope of bringing new insights to our understanding of quantum theory.

Recent results:

- Proof of parallel repetition of entangled games with exponential decay [52],[82],[32].
- Development of a general framework for the study of quantum correlations with combinatorial tools [35].
- New bounds on the quantum value of nonlocal games with graph-theoretical arguments [51].
- Optimal bounds for parity-oblivious random access codes [50].
- Study of Local Orthogonality, a physical principle upper bounding quantum correlations [21].
- Considerations on the notion of dimension of physical systems and its implications for information processing [14].

SPECFUN Project-Team

6. New Results

6.1. Highlights of the Year

Two results are particularly important this year, our computer-checked proof [11] of irrationality of $\zeta(3)$ and our new algorithm [19] for the integration of multiple integrals. The former is our first success in the merger between computer algebra and formal methods, and stimulates further research in this direction around special functions and creative telescoping. The latter has made a large class of integrals possible in practice, thus allowing us to compute a challenging list of integrals related to famous Calabi–Yau various; it has also received attention by physicists.

6.2. A formal proof of the irrationality of $\zeta(3)$

We have obtained a formal proof, machine-checked by the Coq proof assistant, of the irrationality of the constant $\zeta(3)$, that is, the evaluation at 3 of the Riemann zeta function of number theory. The result has been known in mathematics since the French mathematician Apéry's work in 1978, and several alternative proofs have been given since then. Our formalized result is the first complete proof by the computer (under the single assumption of the asymptotic behavior of the least common multiple of the first n natural numbers). The core of this formal proof is based on (untrusted) computer-algebra calculations performed outside the proof assistant with the Mgfun Maple library developed by members of the team in the past. Then, we verify formally and a posteriori the desired properties of the objects computed by Maple and complete the proof of irrationality. This work [11] was formally presented at the conference on interactive theorem proving, ITP'14, and also as talks at MSC 2014 (Mathematical Structures of Computation) [0], at the meeting MAP 2014 of the community on mathematics, algorithms and proofs [0], and at JNCF'14, the meeting of the French computer-algebra community [0].

6.3. Criterion for the existence of telescopers for mixed hypergeometric terms

Creative telescoping is a process that determines a univariate recurrence satisfied by the sum of a summand described by a system of bivariate recurrences. For hypergeometric summands, that is, summands given by first-order linear recurrences, this has led to Zeilberger's algorithm in the early 1990s, since then followed by a large number of works, including a natural counterpart for integration. The history of creative-telescoping algorithms was surveyed this year in Chyzak's HDR [1]. Also this year, we presented in [6] a criterion for the existence of telescopers for mixed hypergeometric terms, which is based on additive and multiplicative decompositions. The criterion had enabled us to determine the termination of Zeilberger's algorithms for mixed hypergeometric inputs prior to any costly computations, and to verify that certain indefinite sums do not satisfy any polynomial differential equation.

6.4. Integration of rational functions

Periods of rational integrals are specific integrals, with respect to one or several variables, whose integrand is a rational function and whose domain of integration is closed. Periods with a parameter are classically known to satisfy linear differential equations of a type called Picard-Fuchs equations. As for other special-function manipulations, handling periods through those differential equations is a good way to actually compute them, and this was the topic of Pierre Lairez' PhD, defended this year [2].

[0] http://smc2014.univ-lyon1.fr/
[0] http://perso.crans.org/cohen/map2014/
[0] http://www.lifl.fr/jncf2014/

Computing multivariate integrals is one speciality of the team and our algorithms are known to treat much more general integrals than just periods of rational integrals. However, integration is still slow in practice when the number of variables goes increasing. By looking at periods of rational function, the hope is to obtain relevant complexity bounds and faster algorithms.

The goal of reaching relevant theoretical complexity bounds has been reached last year [35] but a practically fast algorithm was still missing. This year, we described a new algorithm which is efficient in practice [19], though its complexity is not known. This algorithm allows to compute quickly integrals that are too big to be computed with previous algorithms. As a challenging benchmark, we computed 210 integrals given by Batyrev and Kreuzer in their work on Calabi–Yau varieties. This achievement gave strong visibility to the paper and allowed a quick dissemination of the implementation, which is provided in Magma under a CeCILL B license. The algorithm is now used on a regular basis by several teams. We know of:

- Tom Coates' team (Dpt. of Mathematics, Imperial College, London, UK), which uses the software in their work about mirror symmetry and classification of Fano varieties;
- Duco van Straten (Institute of Mathematics, University of Mainz, Germany), who uses the software in his work in algebraic geometry;
- Gert Alkmvist (Dpt. of Mathematics, University of Lund, Sweden), who uses the software in his work of enumerating the Calabi–Yau differential equations.

6.5. Efficient algorithms for linear differential equations in positive characteristic

The p-curvature of a linear differential operator in characteristic p is a matrix that measures to what extent the space of polynomial solutions of the operator has dimension close to its order. This makes the p-curvature a useful tool in concrete applications, like in combinatorics and statistical physics, where it serves for instance as an a posteriori certification filter for differential operators obtained by guessing techniques. In [9], we designed a new algorithm for computing the characteristic polynomial of the p-curvature in sublinear time $\tilde{\mathrm{O}}(p^{0.5})$. Prior to this work, the fastest algorithms for this task, and even for the subtask of deciding nilpotency of the p-curvature, had had merely slightly subquadratic complexity $\tilde{\mathrm{O}}(p^{1.79})$. The new algorithm is also efficient in practice: it allows to test the nilpotency of the p-curvature for primes p of order 10^6, for which the p-curvature itself is impossible to compute using current algorithms.

6.6. Efficient algorithms for rational first integrals

We presented in [4] fast algorithms for computing rational first integrals with degree bounded by N of a planar polynomial vector field of degree $d \leq N$. The main novelty is that such rational first integrals are obtained by computing via systems of linear equations instead of systems of quadratic equations. This leads to a probabilistic algorithm with arithmetic complexity $\tilde{\mathrm{O}}(N^{2\omega})$ and to a deterministic algorithm for solving the problem in $\tilde{\mathrm{O}}(d^2 N^{2\omega+1})$ arithmetic operations, where ω is the exponent of linear algebra. By comparison, the best previous algorithm uses at least $d^{\omega+1} N^{4\omega+4}$ arithmetic operations. Our new algorithms are moreover very efficient in practice.

6.7. Computation of necessary integrability conditions for parametrized Hamiltonian systems

Let $V(\mathbf{q}_1, \mathbf{q}_2)$ be a homogeneous function whose coefficients depend rationally on parameters $\mathbf{a}_1, ..., \mathbf{a}_n$. In [10] we designed an algorithm to compute polynomial necessary conditions on the parameters $(\mathbf{a}_1, ..., \mathbf{a}_n)$ such that the dynamical system associated to the potential V is integrable. These conditions originate from those of the classical Morales-Ramis-Simó integrability criterion. The implementation of the algorithm allows to treat applications that were out of reach before, for instance concerning the non-integrability of polynomial potentials up to degree 9. Another striking application is the first complete proof of the non-integrability of the collinear three-body problem.

6.8. Non-D-finite excursions in the quarter plane

Counting lattice paths obeying various geometric constraints is a classical topic in combinatorics and probability theory. Many recent works deal with the enumeration of 2-dimensional walks with prescribed steps confined to the positive quadrant. A large part of the effort has been devoted to the classification of classes of walks according to the nature of equations that they satisfy (linear, polynomial, differential, etc). Equivalently, this provides properties of the classes of walks according to the algebraic nature of their enumerative series: whether rational, algebraic, D-finite, etc. The classification is now complete for walks with unit steps: the trivariate generating function of the numbers of walks with given length and prescribed ending point is D-finite if and only if a certain group associated with the step set is finite. We proved in [5] a refinement of this result: we showed that the sequence of numbers of excursions (finite paths starting and ending at the origin) in the quarter plane corresponding to a nonsingular step set with infinite group does not satisfy any nontrivial linear recurrence with polynomial coefficients. This solves an open problem in the field of lattice-path combinatorics.

6.9. A human proof of the Gessel conjecture

Gessel walks are planar walks confined to the positive quarter plane, that move by unit steps in any of the following directions: West, North-East, East and South-West. In 2001, Ira Gessel conjectured a closed-form expression for the number of Gessel walks of a given length starting and ending at the origin. In 2008, Kauers, Koutschan and Zeilberger gave a computer-aided proof of this conjecture. The same year, Bostan and Kauers showed, using again computer algebra tools, that the trivariate generating function of Gessel walks is algebraic. We propose in [17] the first "human proofs" of these results. They are derived from a new expression for the generating function of Gessel walks.

6.10. Enumeration of 3-dimensional lattice walks confined to the positive octant

We explored in [3] the classification problem for 3-dimensional walks with unit steps confined to the positive octant. The first difficulty is their number: there are 11 074 225 cases (instead of 79 in dimension 2). In our work, we focused on the 35 548 that have at most six steps. We applied to them a combined approach, first experimental and then rigorous. Among the 35 548 cases, we first found 170 cases with a finite group; in the remaining cases, our experiments suggest that the group is infinite. We then rigorously proved D-finiteness of the generating series in all the 170 cases, with the exception of 19 intriguing step sets for which the nature of the generating function still remains unclear. In two challenging cases, no human proof is currently known, and we derived computer-algebra proofs, thus constituting the first proofs for those two step sets.

6.11. Asymptotic expansions for linear homogeneous divide-and-conquer recurrences: Algebraic and analytic approaches collated

Linear divide-and-conquer recurrences are a classical topic in computer science, but they are often dealt with in an offhand way. Particularly the subtle oscillations they show are usually not emphasized. After having elaborated last year a new approach to the asymptotic study of such recurrences, we provide in [7] a comparison with an older approach based on number theoretic tools as Dirichlet series and residue computation. The most striking aspect of the linear approach is the simplicity and the ease of use. Reduction to normal Jordan form, computation of a joint spectral radius, dealing with a dilatation equation are all workable with a computer-algebra system. Moreover these concepts are better known by computer scientists than those of complex analysis and analytic number theory. So there is hope that this approach will more easily gain acceptance among computer scientists.

6.12. Asynchronous interaction with Coq

We have integrated the Coq proof assistant with the PIDE architecture [13], [12] ("prover integrated development environment"). The architecture is aimed at asynchronous, parallel interaction with proof assistants, originally aimed at the Isabelle proof assistant, and is tied in heavily with a plugin that allows the jEdit editor to work with proof assistants. We have made several generalizations to the PIDE architecture to accommodate for more provers than just Isabelle, and adapted Coq to understand the core protocol: this delivered a working system in about two man-months; further work improved the connection and added novel functionalities to the interface. The tool has also been presented informally at seminars at the University of Dundee and the Université Paris 13.

VEGAS Project-Team

5. New Results

5.1. Non-linear computational geometry

Participants: Guillaume Moroz, Sylvain Lazard, Marc Pouget, Mohamed Yacine Bouzidi, Laurent Dupont, Olive Chakraborty, Rémi Imbach.

5.1.1. Solving bivariate systems and topology of plane algebraic curves

In the context of our algorithm Isotop for computing the topology of plane algebraic curves [3], we work on the problem of solving a system of two bivariate polynomials. 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. The PhD thesis of Yacine Bouzidi [10] presented several results on this theme obtained during the past three years.

Separating linear forms. We addressed the problem of computing a linear separating form of a system of two bivariate polynomials with integer coefficients, that is a linear combination of the variables that takes different values when evaluated at the distinct solutions of the system. The computation of such linear forms is at the core of most algorithms that solve algebraic systems by computing rational parameterizations of the solutions and this is the bottleneck of these algorithms in terms of worst-case bit complexity. We presented for this problem a new algorithm of worst-case bit complexity $\widetilde{O}_B(d^7 + d^6\tau)$ where d and τ denote respectively the maximum degree and bitsize of the input (and where $\widetilde{O}$ refers to the complexity where polylogarithmic factors are omitted and O_B refers to the bit complexity). This algorithm simplifies and decreases by a factor d the worst-case bit complexity of a previous algorithm we presented in 2013 [24]. Our new algorithm also yields, for this problem, a probabilistic Las-Vegas algorithm of expected bit complexity $\widetilde{O}_B(d^5 + d^4\tau)$. These results were presented at the *International Symposium on Symbolic and Algebraic Computation* in 2014 [15].

Solving bivariate systems & RURs. Given such a separating linear form, we presented an algorithm for computing a RUR with worst-case bit complexity in $\widetilde{O}_B(d^7 + d^6\tau)$ and a bound on the bitsize of its coefficients in $\widetilde{O}(d^2 + d\tau)$. We showed in addition that isolating boxes of the solutions of the system can be computed from the RUR with $\widetilde{O}_B(d^8 + d^7\tau)$ bit operations. Finally, we showed how a RUR can be used to evaluate the sign of a bivariate polynomial (of degree at most d and bitsize at most τ) at one real solution of the system in $\widetilde{O}_B(d^8 + d^7\tau)$ bit operations and at all the $\Theta(d^2)$ real solutions in only $O(d)$ times that for one solution. These results were submitted in 2013, revised in 2014 and will appear in 2015 in the *Journal of Symbolic Computation* [12].

This work is done in collaboration with Fabrice Rouillier (project-team Ouragan at Inria Paris-Rocquencourt).

5.1.2. Topology of the projection of a space curve

Let $\mathcal{C}$ be a real plane algebraic curve defined by the resultant of two polynomials (resp. by the discriminant of a polynomial). Geometrically, such a curve is the projection of the intersection of the surfaces $P(x, y, z) = Q(x, y, z) = 0$ (resp. $P(x, y, z) = \frac{\partial P}{\partial z}(x, y, z) = 0$), and generically its singularities are nodes (resp. nodes and ordinary cusps). State-of-the-art numerical algorithms cannot handle, in practice, the computation of the curve topology in non-trivial instances. The main challenge is to find numerical criteria that guarantee the existence and the uniqueness of a singularity inside a given box B, while ensuring that B does not contain any closed loop of $\mathcal{C}$. We solve this problem by providing a square deflation system that can be used to certify numerically whether B contains a singularity p. Then we introduce a numeric adaptive separation criterion based on interval arithmetic to ensure that the topology of $\mathcal{C}$ in B is homeomorphic to the local topology at p. The theoretical parts of these results are summarized in [18] and are to be combined with experimental data before submission to a journal.

5.1.3. Reflection through quadric mirror surfaces

We addressed the problem of finding the reflection point on quadric mirror surfaces, especially ellipsoid, paraboloid or hyperboloid of two sheets, of a light ray emanating from a 3D point source P_1 and going through another 3D point P_2, the camera center of projection. This is a classical problem known as Alhazen's problem dating from around 1000 A.D. and based on the work of Ptolemy around 150 A.D. [22], [27]. We proposed a new algorithm for this problem, using a characterization the reflection point as the tangential intersection point between the mirror and an ellipsoid with foci P_1 and P_2. The computation of this tangential intersection point is based on our algorithm for the computation of the intersection of quadrics [5], [21]. The implementation is in progress. This work is done in collaboration with Nuno Gonçalves, University of Coimbra (Portugal).

5.1.4. Describing the workspace of a manipulator

We studied the geometry of the solutions of the 3-RPS parallel manipulator. In particular, a parallel manipulator usually has several solutions to the Direct Kinematic Problem. These solutions correspond to different *assembly modes*. A challenge is to find non-singular trajectories connecting different assembly modes. In the literature, this is done by encircling locally a cusp point of the discriminant variety in the joint space. In this work, we used tools from computer algebra to compute a partition of the work space in uniqueness domains. This allowed us to find global singularity-free trajectories reaching up to three assembly modes [16], [17].

5.2. Classical and probabilistic computational geometry

Participants: Sylvain Lazard, Marc Pouget.

5.2.1. Worst-case silhouette size of random polytopes

We studied from a probabilistic point of view the size of the silhouette of a polyhedron. While the silhouette size of a polyhedron with n vertices may be linear for some view points, several experimental and theoretical studies show a sublinear behavior for a wide range of constraints. The latest result on the subject proves a bound in $\Theta(\sqrt{n})$ on the size of the silhouette from a random view point of polyhedra of size n approximating non-convex surfaces in a reasonable way [7]. In this result, the polyhedron is considered given and the sizes of its silhouettes are averaged over all view points. We addressed the problem of bounding the worst-case size of the silhouette where the average is taken over a set of polyhedra. Namely, we consider random polytopes defined as the convex hull of a Poisson point process on a sphere in $\mathbb{R}^3$ such that its average number of points is n. We show that the expectation over all such random polytopes of the maximum size of their silhouettes viewed from infinity is $\Theta(\sqrt{n})$. This work, done in collaboration with Marc Glisse (Inria Geometrica) and Julien Michel (Université de Poitiers), was submitted this year to the *Journal of Computational Geometry* [28].

5.2.2. Recognizing shrinkable complexes is NP-complete

We say that a simplicial complex is shrinkable if there exists a sequence of admissible edge contractions that reduces the complex to a single vertex. We prove [14] that it is NP-complete to decide whether a (three-dimensional) simplicial complex is shrinkable. Along the way, we describe examples of contractible complexes which are not shrinkable. This work was done in collaboration with Dominique Attali (CNRS, Grenoble), Olivier Devillers and Marc Glisse (Inria Geometrica).

5.2.3. On point-sets that support planar graphs

A set of points is said universal if it supports a crossing-free drawing of any planar graph. For a planar graph with n vertices, if edges can be drawn as polylines with at most one bend, we exhibited universal point-sets of size n if the bend-points can be placed arbitrarily [26]. Furthermore, if the bend points are also required to be chosen in the universal set, we proved the existence of universal sets of subquadratic size, $O(n^2/\log n)$ [25]. More recently, we considered the setting in which graphs are drawn with curved edges. We proved that, surprisingly, there also exists a universal set of n points in the plane for which every n-vertex planar graph admits a planar drawing in which the edges are drawn as a circular arc [11].

ALF Project-Team

6. New Results

6.1. Highlights of the Year

André Seznec and Pierre Michaud won the 4th Championship Branch Prediction in all the 3 categories, 4KB, 32 KB and unlimited storage predictors [23], [33], thus confirming the past championships in 2011, 2006 and 2004.

6.2. Processor Architecture

Participants: Pierre Michaud, Bharath Narasimha Swamy, Sylvain Collange, Erven Rohou, André Seznec, Arthur Perais, Surya Khizakanchery Natarajan, Sajith Kalathingal, Tao Sun, Andrea Mondelli, Aswinkumar Sridharan, Alain Ketterlin.

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 (maybe 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 process performance.

6.2.1. Microarchitecture

6.2.1.1. Branch prediction

Participants: André Seznec, Pierre Michaud.

We submitted 3 predictors to the 4th Championship Branch Prediction that took place along with the ISCA 2014 conference [33], [22], [23]. Our predictors combine some branch prediction techniques that we introduced in our previous works, in particular TAGE [10] and GEHL [12]. The predictor we submitted to the 4KB and 32KB tracks was ranked first [33] in both tracks. The 3 predictors we submitted to the unlimited-size track took the first three ranks. We have established a new reference point for branch predictability limits [23].

The 12 competing predictors were mostly using already published branch prediction techniques. The main learning from this year's contest is that choosing the right combination of techniques for the given constraints is at least as important as trying to specialize branch predictors for certain branch behaviors.

6.2.1.2. Revisiting Value Prediction

Participants: Arthur Perais, 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 some silicon area 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.

First, we introduce a new value predictor VTAGE harnessing the global branch history [29]. VTAGE directly inherits the structure of the indirect jump predictor ITTAGE [10]. 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, VTAGE does not suffer from short critical prediction loops and can seamlessly handle back-to-back predictions, contrarily to previously proposed, hard to implement FCM predictors.

Second, we show that all predictors are amenable to very high accuracy at the cost of some loss on prediction coverage [29]. This greatly diminishes the number of value mispredictions and allows to delay validation until commit-time. As such, no complexity is added in the out-of-order engine because of VP (save for ports on the register file) and pipeline squashing at commit-time can be used to recover. This is crucial as adding *selective replay* in the OoO core would tremendously increase complexity.

Third, we leverage the possibility of validating predictions at commit to introduce a new microarchitecture, EOLE [28]. EOLE features *Early Execution* to execute simple instructions whose operands are ready in parallel with Rename and *Late Execution* to execute simple predicted instructions and high confidence branches just before Commit. EOLE depends on Value Prediction to provide operands for *Early Execution* and predicted instructions for *Late Execution*. However, Value Prediction requires EOLE to become truly practical. That is, EOLE allows to reduce the out-of-order issue-width by 33% without impeding performance. As such, the number of ports on the register file diminishes. Furthermore, optimizations of the register file such as *banking* further reduce the number of required ports. Overall EOLE possesses a register file whose complexity is on-par with that of a regular wider-issue superscalar while the out-of-order components (scheduler, bypass) are greatly simplified. Moreover, thanks to Value Prediction, speedup is obtained on many benchmarks of the SPEC'00/'06 suite.

6.2.1.3. Skewed Compressed Caches

Participant: André Seznec.

Cache compression seeks the benefits of a larger cache with the area and power of a smaller cache. Ideally, a compressed cache increases effective capacity by tightly compacting compressed blocks, has low tag and metadata overheads, and allows fast lookups. Previous compressed cache designs, however, fail to achieve all these goals. In this study, we propose the Skewed Compressed Cache (SCC), a new hardware compressed cache that lowers overheads and increases performance. SCC tracks super- blocks to reduce tag overhead, compacts blocks into a variable number of sub-blocks to reduce internal fragmentation, but retains a direct tag-data mapping to find blocks quickly and eliminate extra metadata (i.e., no backward pointers). SCC does this using novel sparse super-block tags and a skewed associative mapping that takes compressed size into account. In our experiments, SCC provides on average 8% (up to 22%) higher performance, and on average 6% (up to 20%) lower total energy, achieving the benefits of the recent Decoupled Compressed Cache [47] with a factor of 4 lower area overhead and lower design complexity.

This study was done in collaboration with Somayeh Sardashti and David Wood from University of Wisconsin.

6.2.1.4. Efficient Execution on Guarded Instruction Sets

Participant: André Seznec.

ARM ISA based processors are no longer low complexity processors. Nowadays, ARM ISA based processor manufacturers are struggling to implement medium-end to high-end processor cores which implies implementing a state-of-the-art out-of-order execution engine. Unfortunately providing efficient out-of-order execution on legacy ARM codes may be quite challenging due to guarded instructions.

Predicting the guarded instructions addresses the main serialization impact associated with guarded instructions execution and the multiple definition problem. Moreover, guard prediction allows to use a global branch-and-guard history predictor to predict both branches and guards, often improving branch prediction accuracy. Unfortunately such a global branch-and-guard history predictor requires the systematic use of guard predictions. In that case, poor guard prediction accuracy would lead to poor overall performance on some applications.

Building on top of recent advances in branch prediction and confidence estimation, we propose a hybrid branch and guard predictor, combining a global branch history component and global branch-and-guard history component. The potential gain or loss due to the systematic use of guard prediction is dynamically evaluated at run-time. Two computing modes are enabled: systematic guard prediction use and high confidence only guard prediction use. Our experiments show that on most applications, an overwhelming majority of guarded instructions are predicted. Therefore a relatively inefficient but simple hardware solution can be used to execute the few unpredicted guarded instructions. Significant performance benefits are observed on most applications while applications with poorly predictable guards do not suffer from performance loss [7].

This study is accepted to ACM Transactions on Architecture and Compiler Optimizations (to appear January 2015) and will be presented at the HIPEAC conference in January 2015.

6.2.1.5. Clustered microarchitecture

Participants: Andrea Mondelli, Pierre Michaud, André Seznec.

In the last 10 years, the clock frequency of high-end superscalar processors did not increase significantly. Performance keeps being increased mainly by integrating more cores on the same chip and by introducing new instruction set extensions. However, this benefits only to some applications and requires rewriting and/or recompiling these applications. A more general way to increase performance is to increase the IPC, the number of instructions executed per cycle.

We argue that some of the benefits of technology scaling should be used to increase the IPC of future superscalar cores. Starting from microarchitecture parameters similar to recent commercial high-end cores, we show that an effective way to increase the IPC is to increase the issue width. But this must be done without impacting the clock cycle. We propose to combine two known techniques: clustering and register write specialization. The objective of past work on clustered microarchitecture was to allow a higher clock frequency while minimizing the IPC loss. This led researchers to consider narrow-issue clusters. Our objective, instead, is to increase the IPC without impacting the clock cycle, which means wide-issue clusters. We show that, on a wide-issue dual cluster, a very simple steering policy that sends 64 consecutive instructions to the same cluster, the next 64 instructions to the other cluster, and so on, permits tolerating an inter-cluster delay of several cycles. We also propose a method for decreasing the energy cost of sending results of one cluster to the other cluster.

This work is currently under submission.

6.2.1.6. Adaptive Intelligent Memory Systems

Participants: André Seznec, Aswinkumar Sridharan.

On multicores, the processors are sharing the memory hierarchy, buses, caches, and memory. The performance of any single application is impacted by its environment and the behavior of the other applications co-running on the multicore. Different strategies have been proposed to isolate the behavior of the different co-running applications, for example performance isolation cache partitioning, while several studies have addressed the global issue of optimizing throughput through the cache management.

However these studies are limited to a few cores (2-4-8) and generally feature mechanisms that cannot scale to 50-100 cores. Moreover so far the academic propositions have generally taken into account a single parameter, the cache replacement policy or the cache partitioning. Other parameters such as cache prefetching and its aggressiveness already impact the behavior of a single thread application on a uniprocessor. Cache prefetching policy of each thread will also impact the behavior of all the co-running threads.

Our objective is to define an Adaptive and Intelligent Memory System management hardware, AIMS. The goal of AIMS will be to dynamically adapt the different parameters of the memory hierarchy access for each individual co-running process in order to achieve a global objective such as optimized throughput, thread fairness or respecting quality of services for some privileged threads.

6.2.2. Microarchitecture Performance Modeling

6.2.2.1. Multiprogram throughput of multicore/SMT processors

Participant: Pierre Michaud.

This research was done in collaboration with Stijn Eyerman and Wouter Rogiest from Ghent University.

There are several aspects to the performance of a multicore processor. One of them is multiprogram throughput, that is, how fast a multicore can execute several independent jobs. However, defining throughput metrics that are both meaningful and practical for computer architecture studies is not straightforward. We present a method to construct throughput metrics in a systematic way: we start by expressing assumptions on job sizes, job types distribution, scheduling, etc., that together define a theoretical throughput experiment. The throughput metric is then the average throughput of this experiment. Different assumptions lead to different metrics, so one should be aware of these assumptions when making conclusions based on results using a specific metric. Throughput metrics should always be defined from explicit assumptions, because this leads to a better understanding of the implications and limits of the results obtained with that metric. We elaborate multiple metrics based on different assumptions. In particular, we show that commonly used throughput metrics such as instructions per cycle and weighted speedup implicitly assume a variable workload, that is, a workload which depends on the machine being evaluated. However, in many situations, it is more realistic to assume a fixed workload. Hence we propose some new fixed-workload throughput metrics. Evaluating these new metrics requires to solve a continuous-time Markov chain. We released a software, TPCalc, that takes as input the performance results of individual coschedules simulations and computes fixed-workload throughput, taking advantage of multicore symmetries [15].

In a subsequent work, we applied our framework to symbiotic scheduling on a symmetric multicore or SMT processor. Symbiotic scheduling tries to exploit the fact, because of resource sharing (execution units, caches, memory bandwidth, etc.) and because different jobs have different characteristics, the performance may be increased by carefully choosing the coschedules. We show that, when assuming a fixed workload, an optimal schedule maximizing throughput can be found by solving a linear programming problem. However, the throughput gains we observed in our experiments, 3% on average, are significantly smaller than what we expected based on published studies on symbiotic scheduling. We analyzed the reasons for this and we found the two main reasons for this discrepancy: previous studies either did not consider a fixed workload but a variable one, or did not report throughput gains but response time reductions. Response time reductions can be artificially magnified by setting the job arrival rate close to the maximum throughput.

This work will be presented at the ISPASS 2015 conference.

6.2.2.2. Modeling multi-threaded programs execution time in the many-core era

Participants: Surya Khizakanchery Natarajan, Bharath Narasimha Swamy, André Seznec.

Estimating the potential performance of parallel applications on the yet-to-be-designed future many cores is very speculative. The traditional laws used to predict performance of an application do not reflect on the various scaling behaviour of a multi-threaded (MT) application leading to optimistic estimation of performance in manycore era. In this paper, we study the scaling behavior of MT applications as a function of input workload size and the number of cores. For some MT applications in the benchmark suites we analysed, our study shows that the serial fraction in the program increases with input workload size and can be a

scalability-limiting factor. Similar to previous studies [41] , we find that using a powerful core (heterogeneous architecture) to execute this serial part of the program can mitigate the impact of serial scaling and improve the overall performance of an application in many-core era [25].

6.2.3. Hardware/Software Approaches

6.2.3.1. Helper threads

Participants: Bharath Narasimha Swamy, Alain Ketterlin, André Seznec.

Heterogeneous Many Cores (HMC) architectures that mix many simple/small cores with a few complex/large cores are emerging as a design alternative that can provide both fast sequential performance for single threaded workloads and power-efficient execution for throughput oriented parallel workloads. The availability of many small cores in a HMC presents an opportunity to utilize them as low-power helper cores to accelerate memory-intensive sequential programs mapped to a large core. However, the latency overhead of accessing small cores in a loosely coupled system limits their utility as helper cores. Also, it is not clear if small cores can execute helper threads sufficiently in advance to benefit applications running on a larger, much powerful, core. In [24], we present a hardware/software framework called core-tethering to support efficient helper threading on heterogeneous many-cores. Core-tethering provides a co-processor like interface to the small cores that (a) enables a large core to directly initiate and control helper execution on the helper core and (b) allows efficient transfer of execution context between the cores, thereby reducing the performance overhead of accessing small cores for helper execution. Our evaluation on a set of memory intensive programs chosen from the standard benchmark suites show that, helper threads using moderately sized small cores can significantly accelerate a larger core compared to using a hardware prefetcher alone. We find that a small core provides a good trade-off against using an equivalent large core to run helper threads in a HMC. Additionally, helper prefetching on small cores when used along with hardware prefetching, can provide an alternate design point to growing instruction window size for achieving higher sequential performance on memory intensive applications.

6.2.3.2. Branch Prediction and Performance of Interpreter

Participants: Erven Rohou, André Seznec, Bharath Narasimha Swamy.

Interpreters have been used in many contexts. They provide portability and ease of development at the expense of performance. The literature of the past decade covers analysis of why interpreters are slow, and many software techniques to improve them. A large proportion of these works focuses on the dispatch loop, and in particular on the implementation of the switch statement: typically an indirect branch instruction. Folklore attributes a significant penalty to this branch, due to its high misprediction rate. We revisit this assumption, considering state-of-the-art branch predictors and the three most recent Intel processor generations on current interpreters. Using both hardware counters on Haswell, the latest Intel processor generation, and simulation of the ITTAGE, we show that the accuracy of indirect branch prediction is no longer critical for interpreters. We further compare the characteristics of these interpreters and analyze why the indirect branch is less important than before.

This study [8] has been accepted for publication at CGO 2015 (International Symposium on Code Generation and Optimization).

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

Participants: Sylvain Collange, André Seznec, Sajith Kalathingal.

We aim at exploring the design of a unique core that efficiently runs both sequential and massively parallel sections. We explore how the architecture of a complex superscalar core has to be modified or enhanced to efficiently run several threads from the same application.

Rather than vectorize at compile-time, our approach is to dynamically vectorize SPMD programs at the micro-architectural level. The SMT-SIMD hybrid core we propose extracts data parallelism from thread parallelism by scheduling groups of threads in lockstep, in a way inspired by the execution model of GPUs. As in GPUs, conditional branches whose outcomes differ between threads are handled with conditionally masked execution. However, while GPUs rely on explicit re-convergence instructions to restore lockstep execution, we target existing general-purpose instruction sets, in order to run legacy binary programs. Thus, the main challenge consists in detecting re-convergence points dynamically.

To handle this difficulty, we can build on [17]. In this work done in collaboration with Fernando Pereira and his team at UFMG, Brasil, we proposed instruction fetch policies that apply heuristics to maximize the cycles spent in lockstep execution, and evaluated them under a micro-architecture independent model [17]. Results highlight the necessity of a tradeoff between maximizing throughput and extracting data-level parallelism with lockstep execution.

6.3. Compiler, vectorization, interpretation

Participants: Erven Rohou, Emmanuel Riou, Bharath Narasimha Swamy, Arjun Suresh, André Seznec, Nabil Hallou, Alain Ketterlin, Sylvain Collange.

6.3.1. Compilers for emerging throughput architectures

Participant: Sylvain Collange.

This work is done in collaboration with Fernando Pereira and his team at UFMG, Brasil.

GPU architectures present new challenges for compilers. Their performance characteristics demand SPMD programs with a high control-flow and memory regularity. Such architecture takes advantage of the regularity in programs to exploit data-level parallelism. In addition to the traditional challenges of code parallelization, new compilers for GPU and future throughput architectures face the task of improving the regularity of parallel programs. In particular, compiler analyses that identify control-flow divergence and memory divergence are a stepping stone for many optimizations. These optimizations include traditional code transformation such as loop interchange and tiling, which use divergence as an additional decision criterion, but also new optimizations specific to GPU architectures such as iteration delaying or branch fusion. In addition, the regularity parameter is an important aspect for workload characterization, as it provides a criterion for task scheduling in heterogeneous environments, such as multi-core processors with GPU. Our objectives include both accurate static and dynamic analyses for thread divergence, and the applications that it enables. We propose to combine static analyses with runtime checks, in order to get the best from both complementary approaches.

6.3.2. Improving sequential performance through memoization

Participants: Erven Rohou, André Seznec, Arjun Suresh.

Many applications perform repetitive computations, even when properly programmed and optimized. Performance can be improved by caching results of pure functions, and retrieving them instead of recomputing a result (a technique called memoization).

We proposed a simple technique for enabling software memoization of any dynamically linked pure function and we illustrate our framework using a set of computationally expensive pure functions – the transcendental functions.

Our technique does not need the availability of source code and thus can be applied even to commercial applications as well as applications with legacy codes. As far as users are concerned, enabling memoization is as simple as setting an environment variable.

Our framework does not make any specific assumptions about the underlying architecture or compiler tool-chains, and can work with a variety of current architectures.

We present experimental results for x86-64 platform using both gcc and icc compiler tool-chains, and for ARM cortex-A9 platform using gcc. Our experiments include a mix of real world programs and standard benchmark suites: SPEC and Splash2x. On standard benchmark applications that extensively call the transcendental functions we report memoization benefits of upto 16 %, while much higher gains were realized for programs that call the expensive Bessel functions. Memoization was also able to regain a performance loss of 76 % in *bwaves* due to a known performance bug in the gcc libm implementation of *pow* function.

6.3.3. Code Obfuscation

Participant: Erven Rohou.

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

A new obfuscation technique [27] based of decomposition of CFGs into threads has been proposed. We exploit the mainstream multi-core processing in these systems to substantially increase the complexity of programs, making reverse engineering more complicated. The novel method automatically partitions any serial thread into an arbitrary number of parallel threads, at the basic-block level. The method generates new control-flow graphs, preserving the blocks' serial successor relations and guaranteeing that one basic-block is active at a time through using guards. The method generates m^n different combinations for m threads and n basic-blocks, significantly complicating the execution state. We also provide proof of correctness for the method.

We propose 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. More precisely a JIT engine is used to generate new versions of a function each time it is invoked, applying different optimizations, heuristics and parameters to generate diverse binary code. A strong random number generator will guarantee that generated code is not reproducible, though the functionality is the same.

This work has been accepted for publication in January 2015 at the International Workshop on Dynamic Compilation Everywhere (DCE-2015).

6.3.4. Padrone

Participants: Erven Rohou, Alain Ketterlin, Emmanuel Riou.

The objective of the ADT PADRONE is to design and develop a platform for re-optimization of binary executables at run-time. Development is ongoing, and an early prototype is functional. In [30], we described the infrastructure of Padrone, and showed that its profiling overhead is minimum. We illustrated its use through two examples. The first example shows how a user can easily write a tool to identify hotspots in their application, and how well they perform (for example, by computing the number of executed instructions per cycle). In the second example, we illustrate the replacement of a given function (typically a hotspot) by an optimized version, while the program runs.

We believe PADRONE fills an empty design point in the ecosystem of dynamic binary tools.

6.3.5. Dynamic Binary Re-vectorization

Participants: Erven Rohou, Nabil Hallou, Alain Ketterlin, Emmanuel Riou.

This work is done in collaboration with Philippe Clauss (Inria CAMUS).

Applications are often under-optimized for the hardware on which they run. Several reasons contribute to this unsatisfying situation, including the use of legacy code, commercial code distributed in binary form, or deployment on compute farms. In fact, backward compatibility of instruction sets guarantees only the functionality, not the best exploitation of the hardware. In particular SIMD instruction sets are always evolving.

We proposed a runtime re-vectorization platform that dynamically adapts applications to execution hardware. Programs distributed in binary forms are re-vectorized at runtime for the underlying execution hardware. Focusing on the x86 SIMD extensions, we are able to automatically convert loops vectorized for SSE into the more recent and powerful AVX. A lightweight mechanism leverages the sophisticated technology put in a static vectorizer and adjusts, at minimal cost, the width of vectorized loops. We achieve speedups in line with a native compiler targeting AVX. Our re-vectorizer is implemented inside a dynamic optimization platform; its usage is completely transparent to the user and requires neither access to source code nor rewriting binaries.

6.4. WCET estimation

Participants: Damien Hardy, Hanbing Li, Isabelle Puaut, Erven Rohou.

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 new results this year are on (i) multi-core architectures (ii) WCET estimation for faulty architectures (iii) traceability of flow information in compilers for WCET estimation.

6.4.1. WCET estimation and its interactions with compilation

6.4.1.1. On the comparison of deterministic and probabilistic WCET estimation techniques

Participants: Damien Hardy, Isabelle Puaut.

This is joint work with Jaume Abella, Eduardo Quinones and Francisco J. Cazorla from Barcelona Supercomputing Center

Several timing analysis techniques have been proposed to obtain Worst-Case Execution Time (WCET) estimates of applications running on a particular hardware. They can be classified into two classes of approaches: deterministic timing analysis techniques (DTA), that produce a unique WCET estimate, and probabilistic timing analysis techniques (PTA) that produce multiple WCET estimates with associated probabilities. Both approaches have their static (SDTA, SPTA) and measurement-based (MBDTA, MBPTA) variants. The lack of comparison figures among those techniques makes complex the selection of the most appropriate one.

This work [19] makes a first attempt towards comparing comprehensively SDTA, SPTA and MBPTA qualitatively and quantitatively, under different cache configurations implementing LRU and random replacement. We identify strengths and limitations of each technique depending on the characteristics of the program under analysis and the hardware platform, thus providing users with guidance on which approach to choose depending on their target application and hardware platform.

6.4.2. WCET estimation for architectures with faulty caches

Participants: Damien Hardy, Isabelle Puaut.

Technology scaling, used to increase performance, has the negative consequence of providing less reliable silicon primitives, resulting in an increase of the probability of failure of circuits, in particular for SRAM cells. While space redundancy techniques exist to recover from failures and provide fault-free chips, they will not be affordable anymore in the future due to their growing cost. Consequently, other approaches like fine grain disabling and reconfiguration of hardware elements (e.g. individual functional units or cache blocks) will become economically necessary. This fine-grain disabling will lead to degraded performance compared to a fault-free execution.

A common implicit assumption in all static worst-case execution time (WCET) estimation methods is that the target processor is not subject to faults. Their result is not safe anymore when using fine grain disabling of hardware components, which degrades performance.

In [16] a method that statically calculates a probabilistic WCET bound in the presence of permanent faults in instruction caches is provided. The method, from a given program, cache configuration and probability of cell failure, derives a probabilistic WCET bound. An essential benefit of our approach is that its probabilistic nature stems only from the probability associated with the presence of faults. By construction, the worst-case execution path cannot be missed, since it is determined using static analysis, extended to cope with permanent faults. This allows our method to be used in safety-critical real-time systems. The method is computationally tractable, since it avoids the exhaustive enumeration of all possible fault locations. Experimental results show that the proposed method accurately estimates WCETs in the presence of permanent faults compared to a method that explores all possible locations for faults. On the one hand, the proposed method allows to quantify the impact of permanent faults on WCET estimates for chips with a known probability of cell failure for the whole chip lifetime. On the other hand, and most importantly, our work can also be used in architectural exploration frameworks to select the most appropriate fault management mechanisms, for current and future chip designs.

6.4.3. Traceability of flow information for WCET estimation

Participants: Hanbing Li, Isabelle Puaut, Erven Rohou.

This research is part of the ANR W-SEPT project.

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 our recent work [21], we have proposed a framework to systematically transform flow information from source code to machine code.

The framework defines a set of formulas to transform flow information for standard compiler optimizations. Transforming the flow information is done within the compiler, in parallel with transforming the code. There thus is no guessing what flow information have become, it is transformed along with the code. The framework is general enough to cover all linear flow constraints and all typical optimizations implemented in modern compilers. Our implementation in the LLVM compiler shows that we can improve the WCET of Malardalen benchmarks by 60% in average (up to 86%) by turning on optimizations. We also provide new insight on the impact of existing optimizations on the WCET.

6.4.4. Verified WCET estimation

Participant: Isabelle Puaut.

This is joint work with Andre Oliveira Maroneze, David Pichardie and Sandrine Blazy from the Celtique group at Inria/IRISA Rennes.

Current WCET estimation tools, even when based on sound static analysis techniques, are not verified. This may lead to bugs being accidentally introduced in the implementation. The main contribution of this work [13], [26] is a formally verified WCET estimation tool operating over C code.

Our tool is integrated to the formally verified CompCert C compiler. It is composed of two main parts: a loop bound estimation and an Implicit Path Enumeration Technique (IPET)-based WCET calculation method. We evaluated the precision of the WCET estimates on a reference benchmark and obtained results which are competitive with state-of-the-art WCET estimation techniques. The code of our tool is automatically generated from its formal specification. Furthermore, machine-checked proofs ensure the estimated WCET is at least as large as the actual WCET.

6.5. Computer arithmetic

Participant: Sylvain Collange.

6.5.1. Application-specific number systems

Collaboration with Mark G. Arnold, XLNS Research, USA.

Reconfigurable FPGA platforms let designers build efficient application-specific circuits, when the performance or energy efficiency of general-purpose CPUs is insufficient, and the production volume is not enough to offset the very high cost of building a dedicated integrated circuit (ASIC). One way to take advantage of the flexibility offered by FPGAs is to tailor arithmetic operators for the application. In particular, the Logarithmic Number System (LNS) is suitable for embedded applications dealing with low-precision, high-dynamic range numbers.

Like floating-point, LNS can represent numbers from a wide dynamic range with constant relative accuracy. However, while standard floating-point offer so-called subnormal numbers to represent numbers close to zero with constant absolute accuracy, LNS numbers abruptly overflow to zero, resulting in a gap in representable numbers close to zero that can impact the accuracy of numerical algorithms.

In collaboration with Mark G. Arnold, Sylvain Collange proposed a generalization of LNS that incorporates features analogous to subnormal floating-point [14]. The Denormal LNS (DLNS) system we introduce defines a class of hybrid number systems that offer quasi-constant absolute accuracy close to zero and quasi-constant relative accuracy on larger numbers. These systems can be configured to range from pure LNS (constant relative accuracy) to fixed-point (constant absolute accuracy across the whole range).

6.5.2. Deterministic floating-point primitives for high-performance computing

Parallel algorithms such as reduction are ubiquitous in parallel programming, and especially high-performance computing. Although these algorithms rely on associativity, they are use on floating-point data, on which operations are not associative. As a result, computations become non-deterministic, and the result may change according to static and dynamic parameters such as machine configuration or task scheduling.

In collaboration with David Defour (UPVD), Stef Graillat and Roman Iakymchuk (LIP6), we introduce a solution to compute deterministic sums of floating-point numbers efficiently and with the best possible accuracy. A multi-level algorithm incorporating a filtering stage that uses fast vectorized floating-point expansions and an accumulation stage based on super-accumulators in a high-radix carry-save representation guarantees accuracy to the last bit even on degenerate cases while maintaining high performance in the common cases [35].

ATEAMS Project-Team

5. New Results

5.1. Highlights of the Year

- Davy Landman, Jurgen Vinju received a Best paper award nomination, for their paper "Empirical analysis of the relationship between CC and SLOC in a large corpus of Java methods"(ICSM'14).

5.2. Cyclomatic complexity $\neq$ Lines of Code

It has long been believed that cyclomatic complexity of source code correlates linearly with lines of code (SLOC). After extensive study of a large corpus of Java source code, Davy Landman and Jurgen Vinju refuted this belief [34]. This provides a new landmark in how to assess and measure the quality of software. In short: cyclomatic complexity measures something different than lines of code.

5.3. Language-Parametric, Capture-Avoiding Program Transformation

Hygienic transformations are well-studied in the area of programming languages that feature (syntax) macros. For instance, in Scheme, macro expansion is guaranteed to not involuntarily capture existing bindings, or allow new bindings to be captured. Together with Sebastian Erdweg and Yi Dai, Tijs van der Storm designed a technique, "name-fix", that can be used to ensure hygiene in arbitrary program transformations, even when source and target language are completely different [24].

5.4. Memory Efficient Hash Tries

The hash trie data structure is a common part in standard collection libraries of JVM programming languages such as Clojure and Scala. It enables fast immutable implementations of maps, sets, and vectors, but it requires considerably more memory than an equivalent array-based data structure. Michael Steindorfer designed a product family of hash tries to generate specialized Java source code [29]. A preliminary experiment on the implementation of sets and maps shows that this technique leads to a median decrease of 55% in memory footprint for maps and 78% for sets.

5.5. Reflection without Remorse

A series of list appends or monadic binds for many monads performs algorithmically worse when it is left-associated. Continuation-passing style (CPS) is well-known to cure this severe dependence of performance on the association pattern. The advantage of CPS dwindles or disappears if we have to examine or modify the intermediate result of a series of appends or binds, before continuing the series. Such examination is frequently needed, for example, to control search in non-determinism monads. Atze van der Ploeg (together with Oleg Kiselyov) developed an alternative approach that is just as general as CPS but more robust: it makes series of binds and other such operations efficient regardless of the association pattern [30]. This solution solves previously undocumented, severe performance problems in iteratees, LogicT transformers, free monads and extensible effects.

5.6. General Parser Combinators

Parser combinators are a well-known approach to parsing where grammars are represented using (higher-order) functions. Unfortunately, parser combinators are commonly implemented using recursive descent parsing as the underlying algorithm. As a result, most parser combinators frameworks do not support left-recursive rules, and may exhibit exponential runtime performance due to backtracking. Anastasia Izmaylova and Ali Afroozeh developed "general parser combinators" (GPC) which do not suffer from these problems: all context-free grammars are supported (even ambiguous ones) and performance is worst-case cubic. As result, GPC combines the expressiveness and performance guarantees of general parsing algorithms like GLL and GLR with the flexibility and extensibility of parser combinators.

CAIRN Project-Team

6. New Results

6.1. Highlights of the Year

Our work on accuracy evaluation and optimisation for fixed point arithmetic was presented during a tutorial "Automatic Fixed-Point Conversion: a Gateway to High-Level Power Optimization" at IEEE/ACM Design Automation and Test in Europe [77].
As a proof of concept of our research on improving efficiency of dynamic reconfiguration in FPGAs [47] [48], the *eFPGA* (Figure 5) chip was designed and fabricated in 65nm CMOS technology. In the proposed and patented architecture [73] (EU patent), the configuration of the FPGA becomes independent from its placement and is moreover significantly compressed (up to $\times 10$). This notion of *Virtual Bit Stream* allows for seamless partial and dynamic reconfiguration and for task migration.

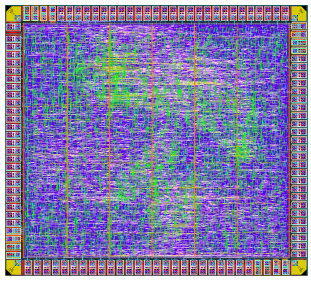

Figure 5. CAIRN*'s eFPGA chip*

6.2. Reconfigurable Architecture Design

6.2.1. Dynamic reconfiguration support in FPGA

Participants: Olivier Sentieys, Antoine Courtay, Christophe Huriaux.

Almost since the creation of the first SRAM-based FPGAs there has been a desire to explore the benefits of partially reconfiguring a portion of an FPGA at run-time while the remainder of design functionality continues to operate uninterrupted. Currently, the use of partial reconfiguration imposes significant limitations on the FPGA design: reconfiguration regions must be constrained to certain shapes and sizes and, in many cases, bitstreams must be precompiled before application execution depending on the precise region of the placement in the fabric. We plan to develop an FPGA architecture that allows for seamless translation of partially-reconfigurable regions, even if the relative placement of fixed-function blocks within the region is changed.

FPGA Architecture Support for Heterogeneous, Relocatable Partial Bitstreams.

The use of partial dynamic reconfiguration in FPGA-based systems has grown in recent years as the spectrum of applications which use this feature has increased. For these systems, it is desirable to create a series of partial bitstreams which represent tasks which can be located in multiple regions in the FPGA fabric. While the transferal of homogeneous collections of lookup-table based logic blocks from region to region has been shown to be relatively straightforward, it is more difficult to transfer partial bitstreams which contain fixed-function resources, such as block RAMs and DSP blocks. In this work we consider FPGA architecture enhancements which allow for the migration of partial bitstreams including fixed-function resources from region to region even if these resources are not located in the same position in each region. Our approach does not require significant, time-consuming place-and-route during the migration process. We quantify the cost of inserting additional routing resources into the FPGA architecture to allow for easy migration of heterogeneous, fixed-function resources. Our experiments show that this flexibility can be added for a relatively low overhead and performance penalty. This work was performed during Christophe Huriaux's visit at UMASS in summer 2014 in the context of Inria Associate Team Hardiesse and has been published in [48] and in [74] as a poster.

Virtual Bit Streams: Design Flow and Run-Time Management of Compressed and Relocatable FPGA Configurations.

The aim of partially and dynamically reconfigurable hardware is to provide an increased flexibility through the load of multiple applications on the same reconfigurable fabric at the same time. However, a configuration bit-stream loaded at runtime should be created offline for each task of the application. Moreover, modern applications use a lot of specialized hardware blocks to perform complex operations, which tends to cancel the "single bit-stream for a single application" paradigm, as the logic content for different locations of the reconfigurable fabric may be different. We proposed a design flow for generating compressed configuration bit-streams abstracted from their final position on the logic fabric. Those configurations can then be decoded and finalized in real-time and at run-time by a dedicated reconfiguration controller to be placed at a given physical location. The VTR framework has been expanded to include bit-stream generation features. A bit-stream format is proposed to take part of our approach and the associated decoding architecture was designed. We analyzed the compression induced by our coding method and proved that compression ratios of at least $2.5\times$ can be achieved on the 20 largest MCNC benchmarks. The introduction of clustering which aggregates multiple routing resources together showed compression ratio up to a factor of $10\times$, at the cost of a more complex decoding step at runtime. Future perspectives on the VBS include extension of the architecture to support commercially available FPGAs as well as the improvement of the associated CAD tool flow to include smarter coding of the VBS to gain in runtime efficiency and in size. The VBS approach can provide increased online relocation capabilities using a decoding algorithm capable of decoding the VBS on-the-fly during the task migration. We applied for a European Patent on this work [73] and the results will be published in 2015 at IEEE/ACM DATE [47].

6.2.2. Power Models of Reconfigurable Architectures

Participants: Robin Bonamy, Daniel Chillet, Olivier Sentieys.

Including a reconfigurable area in complex systems-on-chip is considered as an interesting solution to reduce the area of the global system and to support high performance. But the key challenge in the context of embedded systems is currently the power budget and the designer needs some early estimations of the power consumption of its system. Power estimation for reconfigurable systems is a difficult issue since several parameters need to be taken into account to define an accurate model. In this research, we consider the opportunity of the dynamic reconfiguration for the reduction of power consumption by the management of tasks scheduling and placement. We analyzed the power consumption during the dynamic reconfiguration on a Virtex 5 board. Three models of the partial and dynamic reconfiguration power consumption with different complexity/accuracy tradeoffs are defined. These models are used in design space exploration to evaluate the impact of reconfiguration on energy consumption of a complete system. We propose a methodology for power/energy consumption modeling and estimation in the context of heterogeneous (multi)processor(s) and dynamically reconfigurable hardware systems. We developed an algorithm to explore all task mapping possibilities for a complete application (e.g., for H264 video coding) with the aim to extract one of the best solutions with respect to the designer's requirements. This algorithm is a step ahead for defining on-line power

management strategies to decide which task instances must be executed to efficiently manage the available power using dynamic partial reconfiguration [24].

6.2.3. Real-time Spatio-Temporal Task Scheduling on 3D Architecture

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

One of the main advantages offered by a three-dimensional system-on-chip (3D SoC) is the reduction of wire length between different blocks of a system, thus improving circuit performance and alleviating power overheads of on-chip wiring. To fully exploit this advantage, an efficient management referring to allocate temporarily the tasks at different levels of the architecture is greatly important. In the context of 3D SoC, we have developed several spatio-temporal scheduling algorithms for 3D MultiProcessor Reconfigurable System-on-Chip (3DMPRSoC) architectures composed of a multiprocessor layer and an embedded Field Programmable Gate Array (eFPGA) layer with dynamic reconfiguration. These two layers are interconnected vertically by through-silicon vias (TSVs) ensuring tight coupling between software tasks on processors and associated hardware accelerators on the eFPGA. Our algorithms cope with task dependencies and try to allocate communicating tasks close to each other in order to reduce direct communication cost, thus reducing global communication cost. In the 3DMPRSoC context, our algorithms favor direct communications including: i) point-to-point communication between hardware accelerators on the eFPGA, ii) communication between software tasks through the Network-on-Chip of the multiprocessor layer, and iii) communication between software task and accelerator through TSV. When a direct communication between two tasks occurs, the data are stored in a shared memory placed onto the multiprocessor layer.

The algorithm proposed in [50] considers heterogenenous reconfigurable architecture and proposes a mathematical formulation for spatio-temporal scheduling of a task graph. The placement consists in finding the best mapping of the application task model onto the reconfigurable region. To improve the performance of our algorithm, we propose to configure the tasks by taking account of their priority. The global objective consists in the reduction of the global execution time. The second algorithm presented in [51] improves the previous one and proposes to exploit the presence of processor in the multiprocessor layer in order to anticipate a software execution of a task when no sufficient area is available. In this case, classical algorithms reject the task, and continue their execution. Our algorithm starts a software execution of the task, but the software execution is a speculative execution. Indeed, if a sufficient area is freed by a hardware task later, in this case our algorithm evaluates if the software execution must continue or if it is better to stop this execution to restart the task in the reconfigurable area. We demonstrated that the execution time of an application can be significantly reduced by applying this software speculation.

In [53], we proposed a heuristic which focus on the online task placement problem on a multi-context, dynamically and partially reconfigurable heterogeneous architecture. Configuration prefetching and anti-fragmentation well known techniques are combined with the place reservation technique that takes into account tasks to be placed in the future (pre-allocated tasks) while fulfilling task execution deadline constraint. Compared to a placement without reservation, our approach improves the number of placed tasks and the resource utilization rate.

6.2.4. Run-time Task Management to Increase Resource Utilisation for Concurrent Critical Tasks in Mixed-Critical Systems

Participant: Angeliki Kritikakou.

When integrating mixed critical systems on a multi/many-core system, one challenge is to ensure predictability for the high criticality tasks and an increased utilization for low criticality tasks. In [52], we proposed a distributed run-time WCET controller to address this problem, when several high criticality tasks with different deadlines, periods and offsets are concurrently executed on a multi core system.

During the system execution, the proposed controller regularly checks locally at each critical task if the interferences due to the low criticality tasks can be tolerated. This is achieved by monitoring the ongoing execution time, dynamically computing the remaining worst case execution time of the critical task when only critical tasks are executed on the system and checking our safety condition. In case that the condition is

violated for one critical task, the concurrent execution of the low criticality tasks with the critical one will lead to its deadline miss. Therefore, the local controller decides the suspension of the less critical tasks. However, the local controller is not responsible for the actual suspension of the low criticality tasks. The controller sends a request to a master which has a global view of the system. The master is in charge of collecting the requests of the critical tasks, suspending and restarting the low criticality tasks. When at least one critical task sends the request for suspension of the low criticality tasks, the master suspends them. During execution, the master updates the number of active requests and it restarts the low criticality tasks when all requesters have finished their execution. We have implemented our approach as a software controller on a real multi-core COTS system, the TMS320C6678 chip of Texas Instruments, where we have observed significant gains up to 556% for our case study.

6.2.5. Arithmetic Operators for Cryptography and Fault-Tolerance

Participants: Arnaud Tisserand, Emmanuel Casseau, Nicolas Veyrat-Charvillon, Karim Bigou, Franck Bucheron, Jérémie Métairie, Gabriel Gallin, Huu Van Long Nguyen, Nicolas Estibals.

Arithmetic Operators for Fast and Secure Cryptography.

In the paper [39] presented at ASAP, we describe a new RNS (residue number system) modular multiplication algorithm, for finite field arithmetic over GF(p), based on a reduced number of moduli in base extensions with only $3n/2$ moduli instead of $2n$ for standard ones. Our algorithm reduces both the number of elementary modular multiplications (EMMs) and the number of stored precomputations for large asymmetric cryptographic applications such as elliptic curve cryptography or Diffie-Hellman (DH) cryptosystem. It leads to faster operations and smaller circuits.

The PhD thesis defended by Karim Bigou [16] deals with the RNS representation and the associated arithmetic algorithms for asymmetric cryptography (ECC and RSA). The title of the PhD is "Theoretical Study and Hardware Implementation of Arithmetical Units in Residue Number System (RNS) for Elliptic Curve Cryptography".

Scalar recoding is popular to speed up ECC (elliptic curve cryptography) scalar multiplication: non-adjacent form, double-base number system, multi-base number system (MBNS). Ensuring uniform computation profiles is an efficient protection against some side channel attacks (SCA) in embedded systems. Typical ECC scalar multiplication methods use two point operations (addition and doubling) scheduled according to secret scalar digits. Euclidean addition chains (EAC) offer a natural SCA protection since only one point operation is used. Computing short EACs is considered as a very costly operation and no hardware implementation has been reported yet. We designed an hardware recoding unit for short EACs which works concurrently to scalar multiplication. It has been integrated in an in-house ECC processor on various FPGAs. The implementation results show similar computation times compared to non-protected solutions, and faster ones compared to typical protected solutions (e. g. 18 % speed-up over 192 b Montgomery ladder).

In the paper [40], we introduce a robust asynchronous logic family which does not rely on timing assumptions and/or delay elements and can operate with sub-powered devices. The key element behind our proposal is a simplified completion detection mechanism which makes it substantially more energy effective when compared with other dual-rail approaches. A 32-bit Ripple Carry Adder (RCA) is implemented in 65nm and 45nm CMOS process to evaluate the practicability of our approach. Firstly, the Optimal Energy Point (OEP) of the proposed RCA is investigated by scaling VDD from 0.4V to 0.2V (50mV interval), where the OEP occurs at 0.25V for both technologies. Secondly, while comparing the energy consumption with the corresponding single-rail benchmark at its OEP in 65nm process, 30% (34 fJ for 65nm) and 40% (54fJ for 45nm after scaling) energy savings are achieved respectively. More impressive (10x better) energy efficiency and reasonable performance are obtained over dual-rail counterparts. This work is done in the SPiNaCH project.

ECC Crypto-Processor with Protections Against SCA.

A dedicated processor for elliptic curve cryptography (ECC) is under development. Functional units for arithmetic operations in GF(2^m) and GF(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. This work is done in the PAVOIS project.

Arithmetic Operators and Crypto-Processor for HECC.

In the HAH project, we study and prototype efficient arithmetic algorithms for hyperelliptic curve cryptography for hardware implementations (on FPGA circuits). We study new advanced arithmetic algorithms and representations of numbers for efficient and secure implementations of HECC in hardware.

Arithmetic Operators for Fault Tolerance.

In the ARDyT and Reliasic projects, we work on computation algorithms, representations of numbers and hardware implementations of arithmetic operators with integrated fault detection (and/or fault tolerance) capabilities. The target arithmetic operators are: adders, subtracters, multipliers (and variants of multiplications by constants, square, FMA, MAC), division, square-root, approximations of the elementary functions. We study two approaches: residue codes and specific bit-level coding in some redundant number systems for fault detection/tolerance integration at the arithmetic operator/unit level. FPGA prototypes are under development.

Secure Virtualization in Hardware

In the paper [70] presented at SDTA, we deal with secure solutions that can help virtualization and communication which can be implemented on new hybrids (Core + FPGA) development platforms. On one side, these boards are featured with processors that do not have virtualization extensions but are powerfull enough to really support hypervisors and their guests. On the other side some virtualization solutions presently exist for ARM processors but they only refer to TrustZone for their (hardware) security. These hybrid boards can offer us more: we have read some recents and up-to-date specifications made by a consortium to help the implementation of hardware security. In this area, FPGA can help in securing virtualization. But we must notice that, for now, all has been made for Intel/AMD architectures and for a lone operating system. Even so, the whole propositions are too complex to be implemented on embedded systems. So, we will have to use some capabilities in hardware development and make software rearrangements to help us to design a functional solution.

6.3. Compilation and Synthesis for Reconfigurable Platform

6.3.1. Numerical Accuracy Analysis and Optimization

Participants: Olivier Sentieys, Steven Derrien, Romuald Rocher, Pascal Scalart, Tomofumi Yuki, Aymen Chakhari, Gaël Deest.

The problem of accuracy evaluation is one of the most time consuming tasks during the fixed-point refinement process. Analytical techniques based on perturbation theory have been proposed in order to overcome the need for long fixed-point simulation. However, these techniques are not applicable in the presence of certain operations classified as un-smooth operations. In such circumstances, fixed-point simulation should be used. In [33], an algorithm detailing the hybrid technique which makes use of an analytical accuracy evaluation technique used to accelerate fixed-point simulation was proposed. This technique is applicable to signal processing systems with both feed-forward and feedback interconnect topology between its operations. The proposed algorithm makes use of the classification of operators as smooth or un-smooth and uses the analytical SNS model obtained by using our previously published analytical techniques to evaluate the impact of finite precision on smooth operators, while performing simulation of the un-smooth operators during fixed-point simulation. In other words, parts of the system are selectively simulated only when un-smooth errors occur and not otherwise. Thus, the effort for fixed-point simulation is greatly reduced. The acceleration obtained as a result of applications of the proposed technique is consistent with fixed-point simulation, while reducing the time taken for fixed-point simulation by several orders of magnitude. The preprocessing overhead consists

of deriving the single-noise-source model, and it is often small in comparison to the time required for fixed-point simulation. The advantage of using the proposed technique is that the user need not spend time on characterizing the nonlinearities associated with un-smooth operations. Several examples from general signal processing, communication, and image processing domains are considered for evaluation of the proposed hybrid technique. The acceleration obtained is quantified as an improvement factor. Very high improvement factors indicate that the hybrid simulation is several orders of magnitude faster than classical fixed-point simulation.

One of the limitation of analytical accuracy technique is that they are based on a Signal Flow Graph Representation of the system to be analyzed. This SFG model is currently built-out of a source program by flattening its whole control-flow (including full loop unrolling) which raises significant accuracy analysis issues. To overcome these limitations, we have proposed [41] to adapt state of the art accuracy analysis techniques to take advantage of compact polyhedral program representations. Combining the two approaches provide a more general and scalable framework which significantly extends the applicability of accuracy models, enabling the analysis of complex image processing kernels operating on multidimensional data-sets.

An analytical approach was 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 longer valid 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 was proposed. We focused on recursive structures where an error influences future decision (such as Decision Feedback Equalizer). In that case, numerical analysis method (e.g., Newton Raphson algorithm) can be used. Moreover, an upper bound of the error probability can be analytically determined. We also studied the case of Turbo Coder and Decoder to determine data word-length ensuring sufficient system quality [17].

6.3.2. Reconfigurable Processor Extension Generation

Participants: Christophe Wolinski, François Charot.

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 scheduling into account 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 techniques. This approach allows the features of the extensible processor to be taken into account with a high degree of flexibility. Different architectures models can be envisioned. This can be an extensible processor tightly coupled to a hardware extension having a minimal number of internal registers used to store intermediate results, or a VLIW-oriented extension made up of several processing units working in parallel and controlled by a specialized instruction. These techniques have been implemented in the Gecos source-to-source framework.

Novel techniques addressing the interactions between code transformation (especially loops) and instruction set extension are under study. The idea is to automatically transform the original loop nests of a program (using the polyhedral model) to select specialized and vector instructions. These new instructions may use local memories located in the hardware extension and used to store intermediates data produced at a given loop iteration. Such transformations lead to patterns whose effect is to significantly reduce the pressure on the memory of the processor.

We also studied a way to identify custom instructions at the application domain level instead of addressing it on a per-application basis. Domain-specific instruction set extension aims at maximizing the usage of a custom instruction across a set of applications belonging to an application domain. The idea is to guarantee that each custom instruction has a high degree of utilization across many applications of a given domain,

while still delivering the required performance improvement. The instruction identification problem is here formulated as the maximum common subgraph problem and it is solved by transforming it into a maximum clique problem.

6.3.3. Optimization of Loop Kernels Using Software and Memory Information

Participant: Angeliki Kritikakou.

The compilers optimize the compilation sub-problems one after the other following an order which leads to less efficient solutions because the different sub-problems are independently optimized taking into account only a part of the information available in the algorithms and the architecture. In a paper accepted for publication in Computer Languages, Systems & Structures (COMLAN), Elsevier, we have presented an approach which applies loop transformations in order to increase the performance of loop kernels. The proposed approach focuses on reducing the L1, L2 data cache and main memory accesses and the addressing instructions. Our approach exploits the software information, such as the array subscript equations, and the memory architecture, such as the memory sizes. Then, it applies source-to-source transformations taking as input the C code of the loop kernels and producing a new C code which is compiled by the target compiler. We have applied our approach to five well-known loop kernels for both embedded processors and general purpose processors. From the obtained experimental results we observed speedup gains from 2 up to 18.

6.3.4. Design Tools for Reconfigurable Video Coding

Participants: Emmanuel Casseau, Yaset Oliva Venegas.

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. We have investigated the mapping of RVC applications onto a dedicated multiprocessor platform. Actually, our goal is to propose an automated co-design flow based on the RVC framework. The designer provides the application description in the RVC-CAL language, after which the co-design flow automatically generates a network of processors that can be synthesized on FPGA platforms. Two kinds of platforms can be targeted. The first platform is made of processors based on a low complexity and configurable TTA processor (Very Long Instruction Word -style processor). The architecture model of the platform is composed of processors with their local memories, an interconnection network and shared memories. Both shared and local memories are used to limit the traditional memory bottleneck. Processors are connected together through the shared memories [72] [69] [36]. The second platform more specifically targets the Zynq platform from Xilinx. The processors are MicroBlaze processors. Their local memory is dedicated to instruction code only. A common shared memory is used for the data exchanges between the processors (to store the data that communicate between actors). At present time, the actor mapping is chosen at compile time but we expect dynamic mapping soon. The mapping will be computed at runtime on the ARM processor. The actor's code will be stored in the DDR memory so that it can be easily transferred to the MicroBlaze instruction cache depending on the actor mapping [55] [76] . This work is done in collaboration with IETR and has been implemented in the Orcc open-source compiler (Open RVC-CAL Compiler: http://orcc.sourceforge.net).

6.3.5. A Domain Specific Language for Rapid Prototyping of Software Radio Waveforms

Participants: Matthieu Gautier, Olivier Sentieys, Ganda-Stéphane Ouedraogo.

Software Defined Radio (SDR) is now becoming a ubiquitous concept to describe and implement Physical Layers (PHYs) of wireless systems. Moreover, even though the FPGA (Field Programmable Gate Array) technology is expected to play a key role in SDR, describing a PHY at the Register-Transfer-Level (RTL) requires tremendous efforts. We introduced a novel methodology to rapidly implement PHYs for FPGA-SDR

platforms. The work relies upon High-Level Synthesis tools and dataflow modeling to infer an efficient system-level control unit for the application. The proposed software-based over-layer partly handles the complexity of programming an FPGA and integrates reconfigurable features. It consists essentially of a Domain-Specific Language (DSL) [60] that handles the complexity of programming an FPGA and a DSL-Compiler [32] for automation purpose. IEEE 802.11a a and IEEE 802.15.4 transceivers have been designed and explored [45] via this new methodology in order to show the rapid prototyping feature.

6.4. Interaction between Algorithms and Architectures

6.4.1. Cooperative-cum-Constrained Maximum Likelihood Algorithm for UWB-based Localization in Wireless BANs

Participants: Antoine Courtay, Matthieu Gautier, Gia Minh Hoang [Master's Student].

Wireless Body Area Network (BAN) is a mainstream technology for numerous application fields (medicine, security, sport science, etc.) and precise determination of wireless sensors' positions responses to the great needs in many applications. This study leverages Ultra Wide Band (UWB) radio which is an attractive technology to achieve the centimeter-level distance measurements. However, the aggregation of the distance information remains a challenge to achieve an accurate localization in wireless BAN. To this aim, we have proposed a novel Cooperative-cum-Constrained Maximum Likelihood (CCML) localization algorithm. This algorithmic study shows the improvement that could be achieved by combining UWB radio and dedicated algorithms. Future works is to integrate UWB technology in the second version of the Zyggie platform developed in CAIRN.

6.4.2. MIMO Systems and Cooperative Strategies for Low-Energy Wireless Networks

Participants: Olivier Berder, Olivier Sentieys, Baptiste Vrigneau, Viet-Hoa Nguyen.

Since a couple of years, the CAIRNteam has reached a significant expertise in multi-antenna systems, especially in linear precoding. If this technique is traditionally used in a collocated way, it could also be used for wireless sensor networks (WSN) in a distributed manner. We presented a new approach, named distributed max-dmin precoding (DMP). This protocol is based on the deployment of a virtual 2x2 max-dmin precoding over one source, one forwarding relay, both equipped with one antenna and a destination involving two antennas. In this context, two kinds of relaying, amplify and forward or decode and forward protocols, were investigated. The performance evaluation in terms of Bit-Error-Rate (BER) and energy efficiency was compared with non cooperative techniques (SISO, SIMO) and the distributed space time block code (STBC) scheme. Our investigations showed that the DMP takes the advantage in terms of energy efficiency from medium transmission distances.

A receiver initiated cooperative medium access control (RIC-MAC) protocol was also proposed for cooperative communications to reduce the energy consumption of WSN. Considering a real WSN platform, the simulation results show that using the proposed RIC-MAC protocol in cooperative communications provides latency and energy gains as compared to multi-hop communications. Even if the energy gain is shown to be reduced when the network traffic load increases, our protocol still brings an energy gain about 22% at 1 packet/second. Finally, considering the impact of traffic load on energy consumption and latency, RIC-MAC is illustrated to be robust to traffic load variations in terms of latency [66].

6.4.3. Adaptive protocols for Wireless Sensor Networks

Participants: Olivier Berder, Matthieu Gautier, Nhat-Quang Nhan [Master's Student], Van-Thiep Nguyen.

As tiny sensor nodes are equipped with limited battery, the optimization of the power consumption of these devices is extremely vital. In typical WSN platforms, the radio transceiver consumes major proportion of the energy. Major concerns are therefore to decrease the radio activity by designing efficient MAC protocols.

Energy consumption plays an important role in the design of Wireless Body Area Sensor Network (WBASN). Unfortunately, the performance of WBASNs decreases in high interference environments such as the Industrial, Scientific and Medical (ISM) band where wireless spectrums are getting crowded. In this study [59], an energy-efficient Medium Access Control (MAC) protocol named C-RICER (Cognitive-Receiver Initiated CyclEd Receiver) is specifically designed for WBASN to cognitively work in high interference environment. C-RICER protocol adapts both transmission power and channel frequency to reduce the interferences and thus, the energy consumption. The protocol is simulated with the OMNET++ simulator. Simulation results show that, depending on the interference level, C-RICER is able to outperform the traditional RICER protocol in terms of energy consumption, packet delay, and network throughput.

In recent years, many MAC protocols for Wireless Sensor Networks (WSNs) have been proposed and evaluated using Matlab simulator and/or network simulators (OMNeT++, NS2, etc.). However, most of them have a static behavior and few network simulations are available for adaptive protocols. Specially, in OMNeT++/MiXiM, there is few energy-efficient MAC protocol for WSNs (B-MAC and L-MAC) and no adaptive protocol. To this end, the TAD-MAC (Traffic Aware Dynamic MAC) protocol has been simulated in OMNeT++ with the MiXiM framework [57]. The simulation results have been used to compare with B-MAC and L-MAC protocol, showing the gain brought by TAD-MAC.

6.4.4. Energy Harvesting and Power Management

Participants: Olivier Berder, Olivier Sentieys, Arnaud Carer, Trong-Nhan Le.

To design autonomous Wireless Sensor Networks (WSNs) with a theoretical infinite lifetime, energy harvesting (EH) techniques have been recently considered as promising approaches. Ambient sources can provide everlasting additional energy for WSN nodes and exclude their dependence on battery. An efficient energy harvesting system which is compatible with various environmental sources such as light, heat or wind energy was proposed. Our platform takes advantage of double-level capacitors not only to prolong the system lifetime but also to enable robust booting from the exhausting energy of the system. Simulations and experiments showed that it can achieve booting time in order of seconds. Although capacitors have virtual recharge cycles, they suffer from higher leakage compared to rechargeable batteries. Increasing their size can decrease the system performance due to leakage energy. Therefore, an energy neutral design framework providing a methodology to determine the minimum size of the storage devices satisfying Energy Neutral Operation (ENO) and maximizing system Quality of Service (QoS) in EH nodes when using a given energy source was proposed. Experiments validating this framework were performed on a real WSN platform with both photovoltaic cells and thermal generators in an indoor environment [30].

A new PM for EH-WSNs scavenging energy from periodic sources, i.e., ambient energy is not available during the full harvesting cycle, was proposed. Not only respecting the ENO condition, our PM is able to balance the Quality of Service (QoS) during the whole cycle to provide regular data tracking, which is essential for WSN applications like monitoring. Simulations on OMNET++ show that our PM can improve the QoS during the absence of energy by a factor up to 84% compared to state-of-the-art PMs, while guaranteeing the same global QoS [54].

6.4.5. Multimedia Processing

Participant: Pascal Scalart.

Most noise reduction methods for multimedia signals are usually based on the application of a short-time Wiener filter (MMSE) that is generally expressed as a spectral gain depending on the local signal-to-noise ratio (SNR) on each frequency bin. To estimate such filter, several algorithms can be found in the literature but these conventional approaches lead to a biased estimator for the a priori signal-to-noise estimate. To reduce this bias, we have proposed in [26] a new strategy that relies on the introduction of a correction term in the computation of the Wiener filter depending on the current state of both the available a priori and a posteriori SNR estimates. The proposed solution leads to a bias-compensated a priori SNR estimate, and allows to finely estimating the target signal that is very close to the original noise-free reference. Such refinement procedure has been tested under various noisy environments and show the superiority of the proposed strategy compared to competitive algorithms.

Audio classification systems have recently gained interest for the design of various real-world multimedia services such as audio database indexing with musical genre classification, video indexing using the soundtrack or context awareness. A large majority of audio classification systems can be viewed as offline applications in the sense that there is no strong restriction about how the signal to be classified is accessed. In [44], we investigate the case where the classification task is performed in real-time in a low-latency classification framework. We proposed different methodologies for the use of feature integration that are based on three key aspects: the selection of the features which have to be temporally integrated, the choice of the integration techniques, i.e. how the temporal information is extracted, and the size of the integration window. The experiments carried out for the classification task show that these different methodologies have a significant impact on the global performance even with the low-latency constraints. In addition, we investigate the detection of howlings that arise in audio signals in [43]. To do so, the processing algorithm is based on a Support Vector Machine (SVM) model in the decision stage and on the combination of energy-based features and also a new feature related to the frequency stability of a howling component. The proposed method can be used in different situation since its provides good results with a very low false alarm rate for a wide range of experimental conditions.

6.4.6. Non-Intrusive Load Monitoring

Participants: Olivier Sentieys, Baptiste Vrigneau, Xuan Chien Le.

Natural resource preservation has recently become a significant concern and has therefore motivated many research and development efforts for energy consumption management in buildings and homes. Efficiently reducing energy consumption at home, work or in a factory, could be afforded by mixing different technologies to not only reduce the energy consumed by consumers, but also to adapt (manage) the energy consumed to the energy that is produced. SMART 2020 outlined the opportunity to capture savings of both energy and Greenhouse Gas (GHG) emissions in 2020, through a range of actions developed by the Information and Communications Technologies (ICT) sector. Smart Grid, Smart Buildings, and Green ICT have the main impact on energy savings. At the energy production side, the electrical grid infrastructure is comprised of three elements: power generation, transmission, and distribution. Electrical power generation consists mainly of the power plants but also includes more and more renewable sources such as wind power or solar panels on energy farms or locally on top of buildings. The cost of energy storage is very high, and hence the current practice is to match energy consumption closely with energy generation, which is more and more fluctuating: challenges could be seen as being able to use energy when the wind blows or the sun shines, and also to avoid the strong power consumption peaks due to people's life. A typical example at home could be to automatically use the dryer when energy is available and therefore cheap, and is now well defined as Smart Grid technologies. At the energy consumption side, the main objective is of course to reduce energy consumption of the different subsystems. Interior lighting, office equipment, heating, cooling, and ventilation make up of more than 85% of the total electricity use and the reduction effort should therefore be concentrated on these systems. For energy management and reduction in homes or building a key enabler is the use of wireless sensor networks to monitor the environment (temperature, activity of people, power consumption of equipment, light, etc.) and to act on subsystems (decrease room temperature, stop or start an equipment, adjust cooling or ventilation, etc.). This is the emerging field of Smart Building Automation.

The objective of this work is strongly linked to the usage of these WSN nodes in the context of smart monitoring of energy consumption and environment (temperature, activity, light). We will propose new Indirect Power Monitoring techniques which enable to estimate energy consumed in a building or in a home without effectively measuring the power consumed. A typical AC smart meter is costly equipment and we therefore want to propose cheap and non-invasive sensor nodes. As an example, to estimate the power consumed by the TV, it is not necessary to measure precisely the current it consumed, but a simple sensor able to recognize that TV is on or off can do the same job with a far less complexity. Another example is the development and deployment of room occupancy and people activity sensors that can lead to significant reduction of the energy by regulating HVAC (Heating, Ventilation and Air-Conditioning) or by switching lights and office equipment. The wireless transmission is the main reason of consuming energy and the new algorithms will propose to make the sensors to cooperate inside a low-distance cluster (an office for example). The algorithms will decide the best strategy and the best information to send back in order to offer the best

trade-off between Performance/Complexity/Consumption. This work is closely links to power management techniques and energy harvesting (in-door light, heat, vibration). A power manager embedded in energy harvesting WSN nodes adapts the power consumption and computation loads according to the harvested energy to obtain a theoretically infinite lifetime. The main advantage of using energy harvesting (EH) in the context of building and home monitoring is to avoid battery replacement and therefore to reduce installation and maintenance costs of the system.

CAMUS Team

6. New Results

6.1. Highlights of the Year

One of Philippe Clauss' early papers on Ehrhart polynomials has been celebrated, 18 years later, in a selection of papers for the International Conference on Supercomputing (ICS) 25th anniversary retrospective [13]. 35 papers have been selected among roughly 1800 papers published between 1987 and 2011. The paper is:

"Counting Solutions to Linear and Nonlinear Constraints Through Ehrhart Polynomials: Applications to Analyze and Transform Scientific Programs", by Philippe Clauss, ICS'96, which introduced Ehrhart polynomials in the field of program analysis and optimization.

Philippe Clauss wrote an additional retrospective [12] related to this research which complements the paper in the ICS special issue.

6.2. APOLLO (Automatic speculative POLyhedral Loop Optimizer)

The goal of the APOLLO 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. It is based on the prototype named VMAD which was developed previously by the team between 2009 and 2012. Alexandra Jimborean defended her PhD thesis on this topic in 2012 [30].

APOLLO includes 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 skeletons, and callbacks to the runtime system which is implemented as a dynamic library. External modules associated to specific analyses and transformations are dynamically loaded when required at runtime.

APOLLO uses sampling, multi-versioning and code skeletons 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. These latter versions are generated on-the-fly through fast instantiation of the code skeletons. After each chunk execution, decisions can be taken relatively to the current optimization strategy. APOLLO is handling advanced memory access profiling through linear interpolation of the addresses, dynamic dependence analysis, version selection and speculative polyhedral parallelization [9].

Several extensions and improvements have been implemented inside Apollo in 2014:

- the scheduler of the polyhedral compiler Pluto has been integrated inside the framework. Thus, the runtime decision regarding what optimizing and parallelizing transformation is now entirely depending on Pluto, whose input is generated by the instrumentation and interpolation phase of Apollo [20].
- the static compilation phase of Apollo has been significantly enforced. Linear dependencies between values of scalars and memory addresses are identified in order to alleviate the cost of the instrumented code version. Additionally, memory reference functions that can be disambiguated at compile-time are now fully handled. These improvements are using analysis passes of the LLVM compiler, as well as passes that were specifically developed.
- Apollo is now using the LLVM JIT compiler to further optimize the instantiated code skeletons. Previously, code skeletons were generated as binary executable at compile-time with global variables instantiated at runtime. This approach yielded sub-optimal code including unnecessary or invariant computations. Code skeletons are now kept in LLVM intermediate form until being instantiated and compiled at runtime using the LLVM JIT compiler, thus resulting in faster optimized codes.

- Other memory behavior modeling approaches are now being studied and implemented, in order to allow Apollo handling codes that do not have a completely linear behavior. Three main cases are addressed:
  - quasi-linear behavior in which memory accesses which do not fit the linear prediction are checked on-the-fly, i.e., if these delinquent accesses do not invalidate the current parallel schedule.
  - linear regression behavior in which memory accesses are staying inside a “tube” bordered by linear functions.
  - behavior in which memory accesses are staying inside disjointed address ranges.

6.3. The XFOR programming structure

We have proposed a new programming control structure called “xfor” or “multifor”, providing users a way to schedule explicitly the statements of a loop nest, and take advantage of optimization and parallelization opportunities that are not easily attainable using the standard programming structures. This work is the PhD work of Imen Fassi, who started her work in 2013 and who is co-advised by Yosr Slama, Assistant Professor at the University El Manar in Tunis, Tunisia, and Philippe Clauss.

Data locality optimization is a well-known goal when handling programs that must run as fast as possible or use a minimum amount of energy. However, usual techniques never address the significant impact of numerous stalled processor cycles that may occur when consecutive load and store instructions are accessing the same memory location. In [15], we show that two versions of the same program may exhibit similar memory performance, while performing very differently regarding their execution times because of the stalled processor cycles generated by many pipeline hazards. The xfor structure enables the explicit control of the way data locality is optimized in a program and thus, to control the amount of stalled processor cycles. In [15], we also show the benefits of xfor regarding execution time and energy saving.

While many advanced and fully automatic program analysis and optimization techniques have been developed thanks to the accuracy and expressiveness of the polyhedral model, these techniques may fail in producing efficient codes in some circumstances. The xfor structure eases the manual application of optimizing transformations on loop nests for expert programmers and allows to generate executable codes that may be significantly faster than those generated automatically using well-established polyhedral strategies. we highlight five main gaps regarding these strategies and discuss some ideas on how to bridge them in [14].

6.4. CPU+GPU adaptive computation

We aim to automatically use CPU and GPU to jointly execute a parallel code. To ensure load balance between different PUs, thus to preserve performance, it is necessary to consider the underlying hardware and the program parameters. Compiler optimizations, execution context, hardware availability and specification make it difficult to determine execution times statically. To overcome this hurdle we rely on a portable and automatic method for predicting execution times of statically generated codes on multicore CPUs and on CUDA GPUs. This approach relies on three stages: automatic code generation, offline profiling of the target code and online prediction.

This is mainly the work of PhD student Jean-François Dollinger, advised by Vincent Loechner since 2011. Preliminary results, a "fastest-wins" algorithm between a multicore CPU and the best predicted GPU code version, was published in 2013 in ICPP. Our latest advances, load balancing code between multiple cores CPUs and multiple GPUs will be presented at the IMPACT 2015 workshop [25] in conjunction with the HiPEAC conference. We are currently preparing an extended journal paper to present this work, and Jean-François Dollinger will defend his PhD in 2015.

6.5. Minimizing the synchronization overhead of X10 programs

The CAMUS team has for long focused on compiling, optimizing, and parallelizing *sequential* programs. The project described in this section is somewhat unusual in this context, in that it targets programs written in an explicitly parallel language, and applies polyhedral modeling techniques to reschedule computations, effectively introducing parallel-to-parallel program transformations. This work has been done in collaboration with the Inria COMPSYS team at ENS Lyon, and first results were presented at the *Compiler Construction* conference (CC'14) in April 2014.

The need to leverage the computing power of multi-core processors (and distributed computers) has lead to the design of explicitly parallel programming languages. Such languages often employ a fork/join model, and include syntax to launch and synchronize tasks (also called activities) with well-defined semantics. This brings parallel constructions under the control of the compiler, and introduces new optimization opportunities. Our work has focused on the various synchronization primitives available to the programmer, and more specifically on how one type of synchronization can be replaced with another for specific classes of programs, the goal being to minimize the synchronization overhead. We have demonstrated significant speedups on programs written using the X10 programming language, and have obtained similar results on equivalent Habanero-Java programs.

More specifically, our work focused on synchronization primitives of X10. The X10 language basically has two activity synchronization primitives: one is the explicit use of "clocks" (synchronization barriers) during activity execution, the other is the implicit use of activity containers that synchronize only on the end of activities. Under reasonable conditions on the patterns of activity creation and control, we showed that long-running activities using clocks can be replaced by short-lived activities synchronized only on the end of their containers, and that this transformation provides a significant gain at run time.

We have studied the converse transformation, i.e. starting with an unclocked X10 program, obtaining a system of sequential threads executing in parallel and synchronizing with clocks. This transformation is interesting since it yields to further optimization opportunities. We have elaborated a system of rules to execute the transformation. Applying these rules to "regular" programs gives good results, but fails on some paradigmatic X10 codes. For irregular programs, some parallelism may be lost. We now are investigating a new set of rules to give a correct result for arbitrary X10 programs. A main difficulty is bringing the proof that the set of upgraded rules will give a correct result.

This work has been done in collaboration with Paul Feautrier, member of the COMPSYS Inria team, in ENS Lyon. The CAMUS team has invited Paul Feautrier one more time for one week in June 2014 in Strasbourg.

6.6. Hardware/Software helper thread prefetching

Heterogeneous Many Cores (HMC) architectures that mix many simple/small cores with a few complex/large cores are emerging as a design alternative that can provide both fast sequential performance for single threaded workloads and power-efficient execution for through-put oriented parallel workloads. The availability of many small cores in a HMC presents an opportunity to utilize them as low-power helper cores to accelerate memory-intensive sequential programs mapped to a large core. However, the latency overhead of accessing small cores in a loosely coupled system limits their utility as helper cores. Also, it is not clear if small cores can execute helper threads sufficiently in advance to benefit applications running on a larger, much powerful, core.

In this project, we designed a hardware/software framework called core-tethering to support efficient helper threading on heterogeneous manycores. Core-tethering provides a co-processor like interface to the small cores that (a) enables a large core to directly initiate and control helper execution on the helper core and (b) allows efficient transfer of execution context between the cores, thereby reducing the performance overhead of accessing small cores for helper execution. Our evaluation on a set of memory intensive programs chosen from the standard benchmark suites shows that helper threads using moderately sized small cores can significantly accelerate a larger core compared to using a hardware prefetcher alone. We find that a small core provides a good trade-off against using an equivalent large core to run helper threads in a HMC. Additionally, helper prefetching on small cores when used along with hardware prefetching, can provide an alternate design

point to growing instruction window size for achieving higher sequential performance on memory intensive applications.

This work is a collaboration between the ALF team in Rennes and CAMUS in Strasbourg. Our contribution is mainly on the generation of helper thread code (as a followup to our work on program skeletonization). The result of the work has been published in October 2014 in the Proceedings of the SBAC-PAD conference [17].

6.7. Loop-based Modeling of Parallel Communication Traces

Parallel communication traces are traces of the various actions performed by parallel programs (typically written using MPI or some such library). The traces usually contain actions like message sending and receiving, and entering and exiting collective operations. The goal of this project is to build a model of the parallel program from the traces of the various processes that form the program. Consolidating on our previous work on sequential traces, we have developed an algorithm that takes the traces of the individual processes and merges them into a global model.

The main characteristics of our algorithm is that the result takes the form of loops enclosing various parallel constructs and communication actions. The driving goal of this work is to use the model for various analyzes, mainly to draw qualitative conclusions on the program (like the affinity of the various processes involved), but also to extract quantitative information (like communication matrices). A long term goal is to use the parallel loops to suggest program optimizations.

As of today, our algorithm has been evaluated on several applications. The most obvious is trace compression, with spectacular results because of the underlying loop-nest model (as was already the case for our sequential trace analysis algorithm). Another application is replay, where the program's (actual, i.e., traced) behavior can be simulated on a different parallel architecture. The last application is to build a lightweight model from a subset of trace data, and use the model to index into potentially massive quantitative data associated to the various events.

It turns out that it is difficult to publish such algorithms without evaluating them in "realistic" settings, on applications running on massively parallel hardware, something we don't have easy access to. Also, there are currently a few algorithms that provide similar solutions to practitioners, in a way that we think are fundamentally inferior to our proposition but that seem to be good enough for their current use. Waiting for better opportunities to illustrate the power of our method, we have published a research report summarizing our work [26].

6.8. Switcheable scheduling

Parallel applications used to be executed alone until their termination on partitions of supercomputers. The recent shift to multicore architectures for desktop and embedded systems is raising the problem of the coexistence of several parallel programs. Operating systems already take into account the *affinity* mechanism to ensure a thread will run only onto a subset of available processors (e.g., to reuse data remaining in the cache since its previous execution). But this is not enough, as demonstrated by the large performance gaps between executions of a given parallel program on desktop computers running several processes. To support many parallel applications, advances must be made on the system side (scheduling policies, runtimes, memory management...). However, automatic optimization and parallelization can play a significant role by generating programs with dynamic-auto-tuning capabilities to adapt themselves to the complete execution context, including the system load.

Our approach is to design at compile-time programs that can adapt at run-time to the execution context. The originality of our solution is to rely on *switcheable scheduling*, a selected set of program restructuring which allows to swap between program versions at some meeting points without backtracking. A first step selects pertinent versions according to their performance behavior on some execution contexts. The second step builds the auto-adaptive program with the various versions. Then at runtime the program selects the best version by a low overhead sampling and profiling of the versions, ensuring every computation is useful.

This is an ongoing work with the PhD student Lénaïc Bagnères (POSTALE Team at Inria Saclay-Île-de-France, co-advised by Christine Eisenbeis and Cédric Bastoul). The first results have been presented in 2014 at the Euro-Par International Conference [11].

6.9. Interactive Code Restructuring

This work falls within the exploration and development of semi-automatic programs optimization techniques. It consists in designing and evaluating new visualization and interaction techniques for code restructuring, by defining and taking advantage of visual representations of the underlying mathematical model. The main goal is to assist programmers during program optimization tasks in a safe and efficient way, even if they neither have expertise into code restructuring nor knowledge of the underlying theories. This project is an important step for the efficient use and wider acceptance of semi-automatic optimization techniques, which are still tedious to use and incomprehensible for most programmers. More generally, this research is also investigating new presentation and manipulation techniques for code, algorithms and programs, which could lead to many practical applications: collaboration, tracking and verification of changes, visual search in large amount of code, teaching, etc.

This is a rather new research direction which strengthen CAMUS's static parallelization and optimization issue. It has been initiated at Paris-Sud University as a collaboration between Compilation, represented by Cédric Bastoul before he joined CAMUS, and Human-Machine Interaction, represented by Stéphane Huot from the IN-SITU Team at Inria Saclay-Île-de-France. This work is essentially the PhD topic of Alexander Zinenko (IN-SITU Team at Inria Saclay-Île-de-France, co-advised by Stéphane Huot and Cédric Bastoul, CORDI Grant) which started in 2013. The first results have been presented in 2014 to the IEEE VL/HCC Conference [22]. Moreover, another paper on the topic has been accepted to the International IMPACT 2015 Workshop to be held in conjunction with the HiPEAC International Conference.

COMPSYS Project-Team

6. New Results

6.1. Highlights of the Year

For 2014, from the point of view of organization, funding, collaborations, the main points to highlight are:

- Christophe Alias and Alexandru Plesco have co-founded the XTREMLOGIC start-up in January 2014 (see Section 7.2), following the incubation of Zettice. XTREMLOGIC recently won the "concours région rhône-alpes" grant in November 2014 (40k).
- Tomofumi Yuki was hired as an Inria researcher and became a permanent member of Compsys.
- The 1988 "Array Expansion" seminal paper of Paul Feautrier has been selected for the 25th Anniversary Volume of the ACM International Conference on Supercomputing (ICS) together with 34 other papers selected from the 1800 papers published from 1987 to 2011. A short "reminescence" paper [13] was written for the occasion.
- The team was evaluated in Nov. 2014 by the HCERES (new name of AERES), as part of the LIP lab evaluation. The report has not been received yet.

From a scientific point of view, the shift, in Compsys III, towards the analysis of parallel programs and the extensions of the polyhedral model, both in terms of techniques and applications, is continuing, see the section "New Results", in particular:

- The design (by Christophe Alias and Alexandru Plesco) of a HLS compiler technology (see Section 6.2), patented by Inria [12] and transferred to XTREMLOGIC under an Inria licence (see Section 5.5).
- Two new static analyses: a more precise array bound check analysis [9] (see Section 6.3) and a more scalable termination algorithm for C programs (see Section 6.4).
- A novel equivalence-checking algorithm [7] modulo associativity/commutativity, which is a first step towards semantic program transformations (see Section 6.5).
- A groundbreaking introduction of polyhedral techniques for the analysis of parallel programs, in particular X10 (see [29] and [6]) and OpenStream (see Section 6.6).
- A seminal paper [5] introducing polynomial techniques in program analysis and compilation (see Section 6.7).
- Innovative contributions on parametric tiling [8], [3], [4] as extensions of the polyhedral model (see Sections 6.8 and 6.9).

6.2. Data-Aware Process Networks

Participants: Christophe Alias, Alexandru Plesco [XTREMLOGIC start-up].

Process networks are execution models expressing naturally the parallelism of a computation. They are a natural intermediate representation for high-level synthesis tools, where the front-end extracts the parallelism and produces a process network and the back-end compiles the process network to the target architecture.

In that context, we have defined a new model of process network that fits HLS-specific constraints, the data-aware process network (DPN). Our model makes explicit the communications with the central memory and the parallel access to channels, and is close enough to the hardware constraints to be translated directly to a circuit. We show how to compile an imperative program to a DPN, so as to optimize both the I/O and the parallelism, while using the polyhedral model.

DPNs are used as the intermediate representation for the HLS compiler suite of the XTREMLOGIC start-up. They are generated from C programs by the Dcc compiler (see Section 5.5). The apparatus underlying the DPN synchronizations has been patented by Inria [12].

6.3. Preventing from Out-of-Bound Memory Accesses

Participants: Laure Gonnord, Fernando Pereira [Univ. Mineas Gerais, Brasil].

The C programming language does not prevent out-of-bounds memory accesses. There exist several techniques to secure C programs; however, these methods tend to slow down these programs substantially, because they populate the binary code with runtime checks. To deal with this problem, we designed and tested two static analyses (symbolic region and range analysis), which we combine to remove the majority of these guards.

In addition to the analyses themselves, we brought two other contributions:

- First, we described live-range splitting strategies that improve the efficiency and the precision of our analyses.
- Secondly, we showed how to deal with integer overflows, a phenomenon that can compromise the correctness of static algorithms validating memory accesses.

We validated our claims by incorporating our findings into AddressSanitizer (see https://code.google.com/p/address-sanitizer/). We generated SPEC CINT 2006 code that is 17% faster and 9% more energy efficient than the code originally produced by this tool. Furthermore, our approach is 50% more effective than Pentagons, a state-of-the-art analysis to sanitize memory accesses. This work was published at the OOPSLA 2014 conference [9].

6.4. Scaling Termination Proofs

Participants: Laure Gonnord, Gabriel Radanne [ENS Rennes], David Monniaux [CNRS/VERIMAG], Fernando Pereira [Univ. Mineas Gerais, Brasil], Raphael Rodrigues [Univ. Mineas Gerais, Brasil].

In [15], we presented a new algorithm adapted from scheduling techniques to synthesize (multi-dimensional) affine functions from general flowcharts programs. But, as for other methods, our algorithm tried to solve linear constraints on each control point and each transition, which can lead to quasi-intractable linear programming instances. In contrast to these approaches, we proposed a new algorithm based on the following observations:

- Searching for ranking functions for loop headers is sufficient to prove termination.
- Furthermore, there exist loops such that there is a linear lexicographic ranking function that decreases along each path inside the loop, from one loop iteration to the next, but such that there is no lexicographic linear ranking function that decreases at each step along these paths. For these reasons, it is tempting to treat each path inside a loop as a single transition.

Unfortunately the number of paths may be exponential in the size of the program, thus the constraint system may become very large, even though it features fewer variables. To face this theoretical complexity, even though the number of paths may be large, we argue that, in practice, few of them actually matter in the constraint system (we formalize this concept by giving a characterization as geometric extremal points). Our algorithm therefore builds the constraint system lazily, taking paths into account *on demand*.

In 2014, we consolidated this approach with a work on complexity issues (inspired by [19]) and a new implementation: Termite (see Section 5.13). A corresponding paper is currently under submission for PLDI.

With Fernando Pereira's group in Brazil, we also studied the relevance of fast and simple solutions to compute approximations of the number of iterations of loops (*loop trip count*) of imperative real-world programs. The context of this work is the use of these approximations in compiler optimizations: most of the time, the optimizations yield greater benefits for large trip counts, and are either innocuous or detrimental for small ones. In our paper published at WST'14 [10], we have shown that, most of the time, there is no need to use computationally-expensive state-of-the-art methods to compute (an approximation of) it. We support our position with an actual case study. We show that a fast predictor can be used to speedup the JavaScript JIT compiler of Firefox - one of the most well-engineered runtime environments in use today.

6.5. Equivalence-Checking of Programs with Reductions

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

Program equivalence is a well-known problem with a wide range of applications, such as algorithm recognition, program verification, and program optimization. This problem is also known to be undecidable if the class of programs is rich enough, in which case semi-algorithms are commonly used.

We focused on programs represented as systems of affine recurrence equations (SARE), defined over parametric polyhedral domains, a well-known formalism for the *polyhedral model*. SAREs include, as a proper subset, the class of affine control loop programs. Several semi-algorithms for program equivalence were already proposed for this class. Some take into account algebraic properties such as associativity and commutativity. To the best of our knowledge, none of them manage reductions, i.e., accumulations of a *parametric* number of sub-expressions using an associative and commutative operator. Our main contribution has been a new semi-algorithm to manage reductions. In particular, we outlined the ties between this problem and the perfect matching problem in a parametric bipartite graph.

This work was published at the SAS 2014 conference [7].

6.6. Analysis and Transformation of Parallel Programs

Participants: Albert Cohen [Inria/PARKAS], Alain Darte, Paul Feautrier, Abdoulaye Gamatie [CNRS/LIRMM], Laure Gonnord, Alain Ketterlin [Inria/CAMUS], Sanjay Rajopadhye [Colorado State University], Vijay Saraswat [IBM Research], Eric Violard [Inria/CAMUS], Tomofumi Yuki.

While, historically, Compsys has applied polyhedral analysis to sequential programs, it was recently realized that it also applies to parallel programs, with the aim of checking their correctness or improving their performance. The prospect of having to program exascale architectures, with their millions of cores, has led to the development of new programming languages, whose objective is to increase the programmer productivity. Compsys has applied polyhedral techniques to synchronous languages (see [25], [26] and previous activity reports), to IBM's high-productivity language X10, and, in the context of the ManycoreLabs project, to a streaming language, OpenStream, developed by Albert Cohen's group.

X10 is based on the creation of independent *activities* (light-weight threads), which can synchronize either by a generalization of the fork/join scheme, or with *clocks*, an improved version of the familiar barriers. X10 is deadlock-free by construction but it is the programmer responsibility to insure determinism by a proper use of synchronizations. Non-determinism bugs may have a very low occurrence probability thus be very difficult to detect by testing, hence the interest for detecting races at compile time. In collaboration with CSU (S. Rajopadhye, T. Yuki) and IBM (V. Saraswat), we extended array dataflow analysis to polyhedral clock-free X10 programs [29]. We have been working on clocked programs too: race detection becomes undecidable [30], but realistic problems may still be solved by heuristics.

As a side-effect of this work, we have shown in cooperation with Eric Violard and Alain Ketterlin (Inria Team Camus, Strasbourg) that clocks can be removed and replaced by `async/finish` constructs without modifying the program semantics [6]. While this transformation incurs a large overhead for general programs, in the polyhedral case the overhead is negligible, thus improving the program performance.

In contrast to X10, OpenStream is deterministic by construction, but may have deadlocks. A usual way of disproving deadlocks is by exhibiting a schedule for the program operations, a well-known problem for polyhedral programs, where dependences can be described by affine constraints. In the case of OpenStream, communications use one-dimensional channels and, in a form of linearization, give rise to polynomial dependences for polyhedral OpenStream codes. In a ManycoreLabs project deliverable (see Section 7.1), we have formalized the problem and proved that deadlock detection is undecidable in general.

6.7. Handling Polynomials for Program Analysis and Transformation

Participant: Paul Feautrier.

As shown in the previous section, many problems in parallel programs analysis and verification can be reduced to proving or disproving properties of polynomials in the variables of the program. For instance, so-called "linearizations" (replacing a multi-dimensional object by a one-dimensional one) generate polynomial access functions. These polynomials then reappear in dependence testing, scheduling, and invariant construction. This is also the case in OpenStream where nested loops act on one-dimensional streams. What is needed here is a replacement for the familiar emptiness tests and for Farkas lemma (deciding whether an affine form is positive inside a polyhedron).

Recent mathematical results by Handelman and Schweighofer on the *Positivstellensatz* allow one to devise algorithms that are able to solve these problems. The difference is that one gets only sufficient conditions, and that complexity is much higher than in the affine cases. A paper presenting applications of these ideas to three use cases – dependence testing, scheduling, and transitive closure approximation – will be presented at the 5th International Workshop on Polyhedral Compilation Techniques (IMPACT'15) [5] in Amsterdam in January 2015.

6.8. Parametric Loop Tiling with Constant Aspect Ratio

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

Parametric tiling is a well-known transformation which is widely used to improve locality, parallelism, and granularity (see also the next section for more details). However, parametric tiling is also a non-linear transformation and this prevents polyhedral analysis or further polyhedral transformation after parametric tiling. It is therefore generally applied during the code generation phase.

To address this issue, we presented a method to remain in a polyhedral representation, in a special case of parametric tiling where all the dimensions are tiled and all tile sizes are constant multiples of a single tile size parameter. We call this *Constant Aspect Ratio Tiling*. We showed how to mathematically transform a polyhedron and an affine function into their tiled counterpart, which are the two main operations needed in such a transformation.

The approach is now implemented, and has been tested successfully on several kernels commonly used in the community (matrix multiply, jacobi 1D, jacobi 2D). A corresponding paper was published at the IMPACT 2014 workshop [8].

6.9. Exact and Approximated Data-Reuse Optimizations for Tiling with Parametric Sizes

Participants: Alain Darte, Alexandre Isoard.

Loop tiling is a loop transformation widely used to improve spatial and temporal data locality, to increase computation granularity, and to enable blocking algorithms, which are particularly useful when offloading kernels on computing units with smaller memories. When caches are not available or used, data transfers and local storage must be software-managed, and some useless remote communications can be avoided by exploiting data reuse between tiles. An important parameter of tiling is the sizes of the tiles, which impact the size of the required local memory. However, for most analyses involving several tiles, which is the case for inter-tile data reuse, the tile sizes induce non-linear constraints, unless they are numerical constants. This complicates or prevents a parametric analysis with polyhedral optimization techniques.

We showed that, when tiles are executed in sequence along tile axes, the parametric (with respect to tile sizes) analysis for inter-tile data reuse is nevertheless possible, i.e., one can determine, at compile-time and in a parametric fashion, the copy-in and copy-out data sets for all tiles, with inter-tile reuse, as well as sizes for the induced local memories. When approximations of transfers are performed, the situation is much more complex, and involves a careful analysis to guarantee correctness when data are both read and written. We provide the mathematical foundations to make such approximations possible, thanks to the introduction of the concept of *pointwise functions*. Combined with hierarchical tiling, this result opens perspectives for the automatic generation of blocking algorithms, guided by parametric cost models, where blocks can be pipelined

and/or can contain parallelism. Previous work on FPGAs and GPUs already showed the interest and feasibility of such automation with tiling, but in a non-parametric fashion. Our method is currently implemented with the `iscc` calculator of `ISL`, a library for the manipulation of integer sets defined with Presburger arithmetic, a complete implementation within the PPCG compiler is in progress.

We believe that our approximation technique can be used for other applications linked to the extension of the polyhedral model as it turns out to be fairly powerful. Our future work will be to derive efficient approximation techniques, either because the program cannot be fully analyzable, or because approximations can speed-up or simplify the results of the analysis without losing much in terms of memory transfers and/or memory sizes.

A preliminary version of this work has been presented at the IMPACT'14 workshop [3]. A revised version has been accepted for publication at the International Conference on Compiler Construction (CC'15) [4].

6.10. Studying Optimal Spilling in the Light of SSA

Participants: Florian Brandner [ENSTA ParisTech], Quentin Colombet [Apple, Cupertino], Alain Darte.

Recent developments in register allocation, mostly linked to static single assignment (SSA) form, have shown the benefits of decoupling the problem in two 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 variables to physical registers and reduces the number of move instructions among registers. At the end of Quentin Colombet's PhD thesis, we focused on the first phase, for which many open questions remained: in particular, we studied the notion of optimal spilling (what can be expressed?) and the impact of SSA form (does it help?). To identify the important features for optimal spilling on load-store architectures, we developed a new integer linear programming formulation, more accurate and expressive 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. Based on this formulation, we presented a thorough analysis of the results obtained for the SPECINT 2000 and EEMBC 1.1 benchmarks, from which we drawed the following conclusions: a) rematerialization is extremely important, b) SSA complicates the formulation of optimal spilling, especially because of memory coalescing when the code is not in conventional SSA (CSSA), c) micro-architectural features are significant and thus have to be accounted for, which is not the case with standard cost functions, d) significant savings can be obtained in terms of static spill costs, cache miss rates, and dynamic instruction counts, e) however, cost models based only on static spill costs are not always relevant, in particular when spilling is "at the limit": in this situation, bad interactions with register coalescing and post-pass scheduling can be exacerbated and it may be better to spill a bit more. This important observation indicates that more research is needed to explore alternative cost models that reliably guide spilling.

Parts of this work were already published at CASES 2011. The publication at ACM Transactions on Architecture and Code Optimization [1] contains more detailed discussions, more examples illustrating new concepts and existing approaches, and additional experiments covering the observed worst-case behavior, a new post-latency heuristic, and empiric evidence showing why static spill costs are a poor metric. Three configurations were added: Appel and George under SSA, Koes and Goldstein, and the heuristic of Braun and Hack. This work was partly supported by the Mediacom contract with STMicroelectronics (ended in 2013).

DREAMPAL Team

5. New Results

5.1. Highlights of the Year

The papers [4] and [6] are published in journals (Software Testing Verification and Analisys, resp. Formal Aspects of Computing) that are among the best in their respective fields.

5.2. HoMade

HOMADE V5 is available from 03/2014. New features cover :

- new pipeline architecture with delayed conditional branch
- new unified FSM: Pipeline 2 stages
- renumbering of some IPs
- new activity management on the Slaves in 1D / 2D : by the master OnX , OnY, OnXY, and by the slaves the IPsleep removes the Slave from the next SPMDcall
- new bit per bit loading of program memories, for master and slaves
- new names for some components.
- new versions of a lot of IPs (inside)
- new communication network between Slaves: 2D torus ring with broadcast and communication on x or y axis
- new input binary file format (to respect !!)
- new test_bench for fast reading of instruction files
- new UART wrapper
- new assembler Hasm for those that do not speak binary
- nexys3 version for cheap platform experimentation (does not support more than 2x1 Slaves)
- V6 V7 xilinx supports up to 12 x 12 slaves
- Isim supports many more slaves !!!

More details can be found on www.lifl.fr/~dekeyser/Homade.

5.3. HiHope : A higher level language for the HoMade processor

HiHope is a programming language inspired by Forth used to program the HoMade processor. It includes language constructs for switching at runtime between hardware functions (implemented by IPs) and software functions in a transparent way. We also propose the notion of parallel function language construct. As a result, HiHope programs can use either hardware IPs or software functions, and can perform both sequential and parallel function calls, as well as sequential and parallel function redefinitions.

5.4. Integrating Profiling into MDE Compilers

This work [3] aims at improving performance by returning to the high-level models, specific execution data from a profiling tool enhanced by smart advices computed by an analysis engine. In order to keep the link between execution and model, the process is based on a traceability mechanism. Once the model is automatically annotated, it can be re-factored aiming better performances on the re-generated code. Hence, this work allows keeping coherence between model and code without forgetting to harness the power of parallel architectures. The example uses a transformation chain from UML-MARTE models to OpenCL code.

5.5. Language-Independent Symbolic Execution, Program Equivalence, and Program Verification

A significant part of our research project consists in applying formal techniques for symbolically executing and formally verifying HiHope programs, as well as for formally proving the equivalence of HiHope programs with the corresponding HoMade assembly and machine-code programs obtained by compilation of HiHope.

- Symbolic execution will detect bugs (e.g., stack undeflow) in HiHope programs. Additionaly, symbolic execution is the natural execution manner of HiHope programs as soon as they contain (typically, underspecified) hardware IPs;
- program verification will guarantee the absence of bugs (with respect to specified properties, e.g., no stack underflow, no invocation of unavailable IPs, ...);
- program equivalence will guarantee that such above-mentioned bugs are also absent from the HoMade assembly and machine-code programs obtained by compilation of HiHope source code.

Since these languages are still evolving we decided to work (together with our colleagues from Univ. Iasi, Romania) on language-independent symbolic execution, program-equivalence, and program-verification techniques. In this way, when all the languages in our project become stable, we will be readily able to instantiate the above generic techniques on (the K formal definitions of) the languages in question. We note that all the techniques described below are also independent of K: they are applicable to other language-definition frameworks that use similar rewriting-based formal operational semantics.

5.5.1. Symbolic Execution

In [15] we propose a language-independent symbolic execution framework. The approach is parameterised by a language definition, which consists of a signature for the language's syntax and execution infrastructure, a model interpreting the signature, and rewrite rules for the language's operational semantics. Then, symbolic execution amounts to performing a so-called symbolic rewriting, which consists in changing both the model and the manner in which the operational semantics rules are applied. We prove that the symbolic execution thus defined has the properties naturally expected from it. A prototype implementation of our approach was developed in the K Framework. We demonstrate the genericity of our tool by instantiating it on several languages, and show how it can be used for the symbolic execution, bounded model checking, and deductive verification of several programs. With respect to earlier versions of this work, we have redefined symbolic execution in a more generic way and have included applications to model checking and deductive verification. The current version of the report [15] is submitted to a journal and is based on Andrai Arusoaie's PhD thesis [1], defended in September 2014 at Univ. Iasi (Romania). Andrei was co-supervised by Vlad Rusu and has since joined Dreampal as a postdoc.

5.5.2. Program Equivalences

In [6] we propose a logic and a deductive system for stating and automatically proving the equivalence of programs written in languages having a rewriting-based operational semantics. The chosen equivalence is parametric in a so-called observation relation, and it says that two programs satisfying the observation relation will inevitably be, in the future, in the observation relation again. This notion of equivalence generalises several well-known equivalences and is appropriate for deterministic (or, at least, for confluent) programs. The deductive system is circular in nature and is proved sound and weakly complete; together, these results say that, when it terminates, our system correctly solves the given program-equivalence problem. We show that our approach is suitable for proving equivalence for terminating and non-terminating programs as well as for concrete and symbolic programs. The latter are programs in which some statements or expressions are symbolic variables. By proving the equivalence between symbolic programs, one proves the equivalence of (infinitely) many concrete programs obtained by replacing the variables by concrete statements or expressions. The approach is illustrated by proving program equivalence in two languages from different programming paradigms. The examples in the paper, as well as other examples, can be checked using an online tool. This work was started in 2012. With respect to earlier versions, the new journal publication [6] includes a new and more general presentation of program equivalence as a temporal-logic formula, the generalisation of the

approach to nondeterministic-confluent language semantics, substantially more compact proofs, and a new application to corecursive programs.

In another work [10] we deal with a different kind of equivalence: *mutual equivalence*, which says that two programs are mutually equivalent if they both diverge or they end up in similar states. Mutual equivalence is an adequate notion of equivalence for programs written in deterministic languages. It is useful in many contexts, such as capturing the correctness of, program transformations within the same language, or capturing the correctness of compilers between two different languages. In the case of different languages one needs an operation called *language aggregation*, which we present in [11] in more detail, that combine two languages into a single one. We introduce a language-independent proof system for mutual equivalence, which is parametric in the operational semantics of two languages and in a state-similarity relation. The proof system is sound: if it terminates then it establishes the mutual equivalence of the programs given to it as input. We illustrate it on two programs in two different languages (an imperative one and a functional one), that both compute the Collatz sequence.

5.5.3. Program Verification

In [16] we present an automatic, language-independent program verification approach and prototype tool based on symbolic execution. The program-specification formalism we consider is Reachability Logic, a language-independent alternative to Hoare logics. Reachability Logic has a sound and relatively complete deduction system that offers a lot of freedom to the user regarding the manner and order of rule application, but it lacks a strategy for automatic proof construction. Hence, we propose a procedure for proof construction, in which symbolic execution plays a major role. We prove that, under reasonable conditions on its inputs (the operational semantics of a programming language, and a specification of a program, both given as sets of Reachability Logic formulas) our procedure is partially correct: if it terminates it correctly answers (positively or negatively) to the question of whether the given program specification holds when executing the program according to the given semantics. Termination, of course, cannot be guaranteed, since program-verification is an undecidable problem; but it does happen if the provided set of goals includes enough information in order to be circularly provable (using each other as hypotheses). We introduce a prototype program-verification tool implementing our procedure in the K language-definition framework, and illustrate it by verifying nontrivial programs written in languages defined in K. With respect to earlier versions of this work from 2013, program verification is now presented as a procedure (instead of a proof system), which leads to a direct implementation in the new version of our prototype tool. We also have a new theoretical result: *weak completeness*, which says that a negative answers returned by the verification procedure imply the fact that that the program does not meet its specification. Finally, since Andrei Arusoaie's arrival in the Dreampal team as a postdoc (Nov 2014) we have started working on certifying our verification procedure in the Coq proof assistant.

5.5.4. Language Definitions as Rewrite Theories

In [8] we study the relationships between language definition frameworks (e.g., the K framework) and rewrite theories (e.g., as those embodied in the Maude tool). K is a formal framework for defining the operational semantics of programming languages. It includes software tools for compiling K language definitions to Maude rewrite theories, for executing programs in the defined languages based on the Maude rewriting engine, and for analyzing programs by adapting various Maude analysis tools. A recent extension to the K tool suite is an automatic transformation of language definitions that enables the symbolic execution of programs, i.e., the execution of programs with symbolic inputs. In this paper we investigate more particularly the theoretical relationships between K language definitions and their translations to Maude, between symbolic extensions of K definitions and their Maude encodings, and how the relations between K definitions and their symbolic extensions are reflected on their respective representations in Maude. These results show, in particular, how analyses performed with Maude tools can be formally lifted up to the original language definitions. The results presented in this paper provide the theoretical underpinnings for the current version of the K-Maude tool.

5.6. Hardware chain for partial reconfiguration

The cost overhead due to the use of a softcore processor (MicroBlaze) to drive dynamic reconfiguration led us to explore alternative solutions. The one we have adopted is the use of a dedicated hardware IP (that can be invoked by HoMade) to control and manage dynamic and partial reconfiguration. This approach has led us to develop a complete hardware chain for partial bitstreams reads and writes. The proposed architecture is based on an external memory controller (DDR3) whose role is to manage bitstreams transfers from and to the DDR. Bitstreams loading are managed by a HoMade instruction implemented in a dedicated IP that drives the ICAP interface to transfer data into the reconfigurable area through the physical ICAP. One of the most important performance criteria of dynamic and partial reconfiguration is the reconfiguration time, that we always try to reduce while taking into account the compromise cost / area, speed and power consumption. Preliminary results give a transfer rate exceeding 500 MB/s. Such a result is clearly promising, especially since our hardware reconfiguration chain is constructed to be easily adaptable to SPMD (multi HoMade) needing parallel partial reconfiguration. This work has been the subject of a first communication in the GDR / SOCSIP conference in Paris: 11, 12, 13 June 2014.

5.7. Generic pixel distribution for parallel video processing application

In the frame of the PhD thesis of Karim Ali, we exploited this year the usage of parallel architectures for real-time image/video processing applications. Our main concern was the data distribution according to the parallelism level and respecting real-time processing constraint. As a first step, we proposed a generic pixel distribution model to be used with different image/video applications. Several parameters in the model can be configured according to the required size of the distributed macro-block with the possibility to control the sliding step in both horizontal and vertical directions. We have implemented our architecture on the Xilinx Zynq ZC706 FPGA evaluation board for two applications: the video downscaler (1:16) and the convolution filter. The experimental results showed the low hardware cost of the solution and how flexible is the model to be configured for different distribution scenarios. The architecture and experimental results were published in a paper entitled "A Generic Pixel Distribution Architecture for Parallel Video Processing" at Reconfigurable computing and FPGA international conference (ReConFig) in December 2014, Cancun, Mexico [7].

As a next step, we will reduce the operating clock frequency to decrease the power consumption while increasing the number of processing elements in the parallel architecture to maintain the same performance results. In this way, we will obtain a set of different design points differ in (area, power, other factors) and the system will have the ability to adapt its structure by moving between different design points according to the available resources to keep the same performance measurements. Furthermore, we will target intelligent transportation system, specially dynamic obstacle detection and tracking for autonomous vehicle navigation in collaboration with NAVYA (http://navya-technology.com).

5.8. Massively Parallel Dynamically Reconfigurable Multi-FPGA

In the frame of the PhD thesis of Venkatasubramanian Viswanathan, we conceived and validated a massively parallel and dynamically reconfigurable execution model for next generation high performance embedded systems. We have designed a multi-FPGA platform in order to conceive the massively parallel dynamically reconfigurable execution model. We have used several IP cores developed during the first two years of my PhD in order to test and validate the proposed model. We have proposed a new parallel dynamic reconfiguration mechanism for our architecture. We use our parallel reconfiguration model to reconfigure a subset or several IPs in parallel. We have proposed a partial reconfiguration model for next generation 3D FPGAs well-traced on the execution model (SPMD) in order to reconfigure in parallel a subset of the computing nodes. Finally, we have used the PicoComputing platform as an example to validate our proposed execution and reconfiguration models.

In order to demonstrate various features of such an architecture, we have implemented a scalable distributed secure H.264 encoding application with a FMC based high-speed sFPDP (serial Front Panel Data Port) data acquisition protocol to capture RAW video data. The system has been implemented on 3 different FPGAs, respecting the SPMD execution model managing several input video sources in parallel. We have measured various performance metrics of the proposed massively parallel dynamically reconfigurable system

and demonstrated several benefits. This work is going to be published in the FPGA 2015 conference as a poster titled "A Parallel And Scalable Multi-FPGA based Architecture for High Performance Applications" [13].

Later an ICAP controller was setup for dynamic partial reconfiguration in order to swap IPs during runtime on a single FPGA. We have used this IP along with the parallel communication feature of the multi-FPGA architecture, in order to broadcast a partial bitstream to all FPGAs at the same time and to do a parallel DPR in several FPGAs, thus emulating the reconfiguration model for next generation 3D FPGAs. These results represent a conceptual proof for a massively parallel dynamically reconfigurable next generation embedded computers that will use 3D PFGAs and reconfigure several logic layers in parallel.

5.9. HoMade-based MPSoC

The goal of this work is to build an MPSoC based on HoMade. The aimed system is a completely dynamically reconfigurable system. This mean that both the processing elements (HoMade) and the interconnection network are dynamically reconfigurable. The basic block in this system developped here is the interconnection network. It is a MIN (Multistage Interconnection Network) that would utilize oversizing techniques in order to reconfigure the network depending on the traffic.

5.10. Communication-Computation Overlap in Massively Parallel System-on-Chip

The Synchronous Communication Asynchronous Computation (SCAC) model is an execution model dedicated to the Massively Parallel System-on-Chip. This model proposes a novel processing paradigm, teh communication-computation overlap [17]. This concept does not only consider the programming level but also the implementation level. Using a decoupled control structure, the synchronous communication control is performed independently of the asynchronous computation control. Separating these two control phases allows the programmer to define programming strategies that overlap communication by computation to decrease the execution time.

To achieve this communication-computation overlap in SCAC architecture while avoiding the centralized control, in addition to the master controller, we define a second hierarchical control level, namely the slave controllers. The concept of this dual ontrol structure departs from the centralized configuration and instead of a uni-processor master controlling a set of parallel Processing Elements (PEs), the master cooperates with a grid of parallel slave controllers which supervises the activities of cluster of PEs. Based on this decoupled control structure, the programmer can manage the master-slave program to overlap communication by computation phase. Therefore, the basic idea to implement this paradigm is to divide the principal program into small blocks of parallel instructions, called Slave Program (SP), and send these blocks to the activated PEs of the system. Then, according to a predefined mask, the slave controllers send the begin execution orders. In parallel to computation, the slave controllers manage the synchronous inter-node communication. Distinguish communication from computation needs the separation of these two phases in different blocks. This repartition should be provided at programming level. Then, the overlapped execution of these blocks will be done in parallel according to the program description.

The aim of these last works is to define a new paradigm of a communication-computation overlap in massively parallel System-on-Chip. This paradigm allows to decrease the execution time of parallel programs using specific strategies in the programming level and a partially decoupled control system in the hardware level. The difficulty of implementing this paradigm lies in the coordination between the programming level and the architecture designing level in order to hide the communication cost.

GCG Team

6. New Results

6.1. Highlights of the Year

Graduate Research Award of the OSU department in 2015 for Venmugil Elango (co-advised by Fabrice Rastello)

6.2. An interval constrained memory allocator for a GAS runtime

Participants: François Gindraud, Fabrice Rastello, Albert Cohen [ENS Ulm].

This work presents a memory allocator for global address space (GAS) runtime targeting distributed memory embedded architectures (MPSoC). MPSoC we are interested in are relatively new architectures, composed of several nodes with multiple general purpose cores and a local memory, linked by a network, all on one chip (NoC). They have promising energy and computing performances, but are hard to program due to the multilevel parallelism and the hardware constraints (limited memory, network structure). Existing programming framework are either thin but let the programmer do the hard choices (OpenMP + MPI) or heavy and automatic but target specific kind of applications on big systems (Global Arrays).

Givy 5.1 is a runtime currently developed to execute dynamic task graphs with data-flow dependencies on MPSoC. It has a focus on supporting irregular applications, using the dependencies to perform data-aware dynamic task scheduling and data transfer. Data blocks live in a GAS, and thus requires a GAS-aware memory allocator to avoid address collisions when they are dynamically allocated. The allocator implementation proposed in this paper does this with zero synchronization between nodes, while being memory efficient in the small distributed memories, and fast on each multithreaded node.

This work will be submitted at ACM ISMM Symposium.

6.3. A Framework for Enhancing Data Reuse via Associative Reordering

Participants: Kevin Stock [OSU], Martin Kong [OSU], Tobias Grosser [ENS Ulm], Louis-Noël Pouchet [UCLA], Fabrice Rastello, J. Ramanujam [LSU], P. Sadayappan [OSU].

The freedom to reorder computations involving associative operators has been widely recognized and exploited in designing parallel algorithms and to a more limited extent in optimizing compilers.

In this work, we develop a novel framework utilizing the associativity and commutativity of operations in regular loop computations to enhance register reuse. Stencils represent a particular class of important computations where the optimization framework can be applied to enhance performance. We show how stencil operations can be implemented to better exploit register reuse and reduce load/stores. We develop a multi-dimensional retiming formalism to characterize the space of valid implementations in conjunction with other program transformations. Experimental results demonstrate the effectiveness of the framework on a collection of high-order stencils.

This work is the fruit of the collaboration 8.1 with OSU and has been presented at the conference ACM PLDI'14.

6.4. Beyond Reuse Distance Analysis: Dynamic Analysis for Characterization of Data Locality Potential

Participants: Naznin Fauzia [OSU], Venmugil Elango [OSU], Mahesh Ravishankar [OSU], J. Ramanujam [LSU], Fabrice Rastello, Atanas Routnev [OSU], Louis-Noël Pouchet [UCLA], P. Sadayappan [OSU].

Emerging computer architectures will feature drastically decreased flops/byte (ratio of peak processing rate to memory bandwidth) as highlighted by recent studies on Exascale architectural trends. Further, flops are getting cheaper while the energy cost of data movement is increasingly dominant. The understanding and characterization of data locality properties of computations is critical in order to guide efforts to enhance data locality.

Reuse distance analysis of memory address traces is a valuable tool to perform data locality characterization of programs. A single reuse distance analysis can be used to estimate the number of cache misses in a fully associative LRU cache of any size, thereby providing estimates on the minimum bandwidth requirements at different levels of the memory hierarchy to avoid being bandwidth bound. However, such an analysis only holds for the particular execution order that produced the trace. It cannot estimate potential improvement in data locality through dependence preserving transformations that change the execution schedule of the operations in the computation.

In this work, we develop a novel dynamic analysis approach to characterize the inherent locality properties of a computation and thereby assess the potential for data locality enhancement via dependence preserving transformations.

This work is the fruit of the collaboration 8.1 with OSU and has been published at ACM TACO'14.

6.5. On Using the Roofline Model with Lower Bounds on Data Movement

Participants: Venmugil Elango [OSU], Naser Sedaghati [OSU], Fabrice Rastello, Louis-Noël Pouchet [UCLA], J. Ramanujam [LSU], Radu Teodorescu [OSU], P. Sadayappan [OSU].

The roofline model is a popular approach to "bounds and bottleneck" performance analysis. It focuses on the limits to performance of processors because of limited bandwidth to off-chip memory. It models upper bounds on performance as a function of operational intensity, the ratio of computational operations per byte of data moved from/to memory. While operational intensity can be directly measured for a specific implementation of an algorithm on a particular target platform, it is of interest to obtain broader insights on bottlenecks, where various semantically equivalent implementations of an algorithm are considered, along with analysis for variations in architectural parameters. This is currently very cumbersome and requires performance modeling and analysis of many variants.

In this work, we alleviate this problem by using the roofline model in conjunction with upper bounds on the operational intensity of computations as a function of cache capacity, derived using lower bounds on data movement. This enables bottleneck analysis that holds across all dependence-preserving semantically equivalent implementations of an algorithm. We demonstrate the utility of the approach in in assessing fundamental limits to performance and energy efficiency for several benchmark algorithms across a design space of architectural variations.

This work is the fruit of the collaboration 8.1 with OSU and is to be published at ACM TACO'15.

6.6. On Characterizing the Data Access Complexity of Programs

Participants: Venmugil Elango [OSU], Fabrice Rastello, Louis-Noël Pouchet [UCLA], J. Ramanujam [LSU], P. Sadayappan [OSU].

Technology trends will cause data movement to account for the majority of energy expenditure and execution time on emerging computers. Therefore, computational complexity will no longer be a sufficient metric for comparing algorithms, and a fundamental characterization of data access complexity will be increasingly important. The problem of developing lower bounds for data access complexity has been modeled using the formalism of Hong & Kung's red/blue pebble game for computational directed acyclic graphs (CDAGs). However, previously developed approaches to lower bounds analysis for the red/blue pebble game are very limited in effectiveness when applied to CDAGs of real programs, with computations comprised of multiple sub-computations with differing DAG structure. We address this problem by developing an approach for effectively composing lower bounds based on graph decomposition. We also develop a static analysis algorithm to derive the asymptotic data-access lower bounds of programs, as a function of the problem size and cache size.

This work is the fruit of the collaboration 8.1 with OSU and is to be presented at ACM POPL'15.

6.7. PolyCheck: Dynamic Verification of Iteration Space Transformations on Affine Programs

Participants: Sriram Krishnamoorthy [PNNL], Bao Wenlei [OSU], Louis-Noël Pouchet [UCLA], P. Sadayappan [OSU], Fabrice Rastello.

High-level compiler transformations, especially loop transformations, are widely recognized as critical optimizations to restructure programs to improve data locality and expose parallelism.

Guaranteeing the correctness of program transformations is essential, and to date three main approaches have been developed: proof of equivalence of affine programs, matching the execution traces of programs, and checking bit-by-bit equivalence of the outputs of the programs. Each technique suffers from limitations in either the kind of transformations supported, space complexity, or the sensitivity to the testing dataset. In this paper, we take a novel approach addressing all three limitations to provide an automatic bug checker to verify any iteration reordering transformations on affine programs, including non-affine transformations, with space consumption proportional to the original program data, and robust to arbitrary datasets of a given size. We achieve this by exploiting the structure of affine program control- and data-flow to generate at compile-time a lightweight checker code to be executed within the transformed program. Experimental results assess the correctness and effectiveness of our method, and its increased coverage over previous approaches.

This work is the result of the collaboration 8.1 with OSU.

6.8. On Using Lower Bounds for Discrimination of Utility/Futility of Loop Fusion

Participants: Samyam Rajbhandari [OSU], Martin Konk [OSU], P. Sadayappan [OSU], Robert J. Harrison [Stonybrook], Fabrice Rastello.

Fusion is an important loop transformation for data locality enhancement. However, it is very challenging to determine which of a set of possible fusion choices is best. In this paper, we pursue a novel approach to addressing this problem. Instead of the conventional approach of explicitly modeling different possible fused loop configurations and modeling the expected performance with each, we instead use lower bounds modeling to characterize conditions where fusion might have utility and where it will be futile because the maximal possible improvement from fusion is much lower than the minimal data movement overheads for each of the unfused components. We successfully demonstrate the use of such a methodology with two practically important codes from the quantum chemistry domain, i) with the affine 4-index transform code, and ii) unstructured tree operations with the MADNESS framework.

This work is the result of the collaboration 8.1 with OSU.

6.9. A Tiling Perspective for Register Optimization

Participants: Duco Van Amstel, Lukasz Domagala, P. Sadayappan [OSU], Fabrice Rastello.

Register allocation is a much studied problem. A particularly important context for optimizing register allocation is within loops, since a significant fraction of the execution time of programs is often inside loop code. A variety of algorithms have been proposed in the past for register allocation, but the complexity of the problem has resulted in a decoupling of several important aspects, including loop unrolling, register promotion, and instruction reordering.

In this work, we develop an approach to register allocation and promotion in a unified optimization framework that simultaneously considers the impact of loop unrolling and instruction scheduling. This is done via a novel instruction tiling approach where instructions within a loop are represented along one dimension and innermost loop iterations along the other dimension. By exploiting the regularity along the loop dimension, and imposing essential dependence based constraints on intra-tile execution order, the problem of optimizing register pressure is cast in a constraint programming formalism. Experimental results are provided from thousands of innermost loops extracted from the SPEC benchmarks, demonstrating improvements over the current state-of-the-art.

This work is the fruit of both the collaboration 8.1 with OSU and with Kalray 7.1 7.2 .

6.10. Hybrid Pointer Disambiguation

Participants: Fernando Pereira, Alexandros Labrineas, Péricles Alves, Fabian Gruber, Fabrice Rastello.

In order to provide effective optimizations, compilers must deal with memory dependences. However, the state-of-the-art heuristics available in the literature to track memory dependencies are inherently imprecise and computationally expensive. Consequently, the most advanced code transformations that compilers have today are ineffective when applied on real-world programs. The goal of this paper is to solve this conundrum - a goal that we accomplish through the hybrid disambiguation of pointers. We provide a static analysis that generates dynamic tests to determine when two memory locations can overlap. We then produce two versions of a loop: one that is aliasing-free - hence, easy to optimize - and another that is not. Our checks lets us safely branch to the optimizable region. We have applied these ideas on Polly-LLVM, a loop optimizer built on top of the LLVM compilation infrastructure. Our experiments indicate that our method is precise, effective and useful: we can disambiguate the vast majority of checks in benchmarks that go from the loop intensive Polybench suite to the more general SPEC CPU 2006 benchmark collection. The result of this precision is code quality: the binaries that we generate are 9.5% faster than those that Polly-LLVM produces without our optimization. Given the current technology to statically solve alias analysis, we believe that our ideas are a necessary step to make modern compiler optimizations useful in practice.

This work is the fruit of the collaboration with UFMG 8.1 and Kalray 7.1 7.2 .

6.11. Parameterized Construction of Program Representations for Sparse Dataflow Analyses

Participants: André Tavares [ENS Lyon], Benoit Boissinot [ENS Lyon], Fernando Pereira, Fabrice Rastello.

Data-flow analyses usually associate information with control flow regions. Informally, if these regions are too small, like a point between two consecutive statements, we call the analysis dense. On the other hand, if these regions include many such points, then we call it sparse. This paper presents a systematic method to build program representations that support sparse analyses. To pave the way to this framework we clarify the bibliography about well-known intermediate program representations. We show that our approach, up to parameter choice, subsumes many of these representations, such as the SSA, SSI and e-SSA forms. In particular, our algorithms are faster, simpler and more frugal than the previous techniques used to construct SSI - Static Single Information - form programs. We produce intermediate representations isomorphic to Choi *et al.*'s Sparse Evaluation Graphs (SEG) for the family of data-flow problems that can be partitioned per variables. However, contrary to SEGs, we can handle - sparsely - problems that are not in this family. We have tested our ideas in the LLVM compiler, comparing different program representations in terms of size and construction time.

This work is the fruit of the collaboration with UFMG 8.1 and has been presented at Springer CC'14.

6.12. Time-critical Computing on a Single-chip Massively Parallel Processor

Participants: Benoit Dupont-de-Dinechin [Kalray], Duco Van Amstel, Marc Poulhiès [Kalray], Guillaume Lager [Kalray].

In this work we demonstrate the capabilities of the MPPA(TM)-256 chip in the field of time-critical computations. This manycore chip features amongst others a Network-on-Chip (NoC) linking the seperate computational clusters each disposing of its own local memory and processing elements (PEs). The PEs architectural features induce a locally deterministic behaviour and the memory access arbitration that is used allows for a Worst-Case Execution Time (WCET) that is achieved for the combination of all local worst-cases. As such, in order to achieve a WCET analysis for a full MPPA(TM)-256 chip, we provide a Worst-Case Traversal Time (WCTT) analysis for the NoC to link the WCETs provided by each computational cluster. This part of the work is based on the (sigma, rho) model used for general network flow analysis and Quality-of-Service (QoS) parametrization.

This work has been presented at DATE'14.

6.13. Guaranteed Services of the NoC of a Manycore Processor

Participants: Benoit Dupont-de-Dinechin [Kalray], Yves Durand [CEA], Duco Van Amstel [Kalray], Alexandre Ghiti [Kalray].

In the case of the MPPA(TM)-256 chip the study of the integrated Network-on-Chip (NoC) is a fundamental subject for anyone using this architecture for time-critical purposes or real-time use-cases that need guarantees on the Worst-Case Traversal Time (WCTT) of the NoC. Previous work has already shown that the MPPA(TM)-256 NoC can be modelled using the (sigma, rho)-model. In the current work we will elaborate on this point by providing an indepth analysis of the NoC as well as the method to guarantee Quality-of-Service properties.

This work has been presented at the International Workshop on Network on Chip Architectures 2014.

PAREO Project-Team

6. New Results

6.1. Static Analysis

Participant: Sergueï Lenglet.

6.1.1. Static Analysis for Control Operators

Control operators, such as *call/cc* in Scheme or SML, allow programs to have access and manipulate their execution context. We study the behavioral theory of the $\lambda\mu$-calculus, an extension of the λ-calculus with a control feature similar to *call/cc*. In [6], we define an applicative bisimilarity for the $\lambda\mu$-calculus, demonstrating the differences in the definitions between call-by-name and call-by-value. We give equivalence examples to illustrate how our relations can be used; in particular, we prove David and Py's counter-example, which cannot be proved with the preexisting bisimilarities for the $\lambda\mu$-calculus. The proofs are in the accompanying research report [8], where we also define environmental bisimulations for the calculus.

6.1.2. Polymorphism and Higher-order Functions for XML

In [7], we define a calculus with higher-order polymorphic functions, recursive types with arrow and product type constructors and set-theoretic type connectives (union, intersection, and negation). We study the explicitly-typed version of the calculus in which type instantiation is driven by explicit instantiation annotations. In particular, we define an explicitly-typed λ-calculus with intersection types and an efficient evaluation model for it. The work presented in this article provides the theoretical foundations needed to design and implement higher-order polymorphic functional languages for semi-structured data.

6.2. Termination under Strategies

Participants: Horatiu Cirstea, Sergueï Lenglet, Pierre-Etienne Moreau.

Several approaches for proving the confluence and the termination of term rewriting systems have been proposed [10] and the corresponding techniques have been implemented in tools like Aprove [17] and TTT2 [26]. 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 [15] 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 tools 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. The corresponding tool is available at http://gforge.inria.fr/projects/tom.

6.3. Property-based Testing

Participants: Nauval Atmaja, Horatiu Cirstea, Pierre-Etienne Moreau.

Quality is crucial for software systems and several aspects should be taken into account. Formal verification techniques like model checking and automated theorem proving can be used to guarantee the correctness of finite or infinite systems. While these approaches provide a high level of confidence they are sometimes difficult and expensive to apply. Software testing is another approach and although it cannot guarantee correctness it can be very efficient in finding errors.

We have proposed a property based testing framework for the *Tom* language inspired from the ones proposed in the context of functional programming. The previously developed tool has been improved by integrating it in the *Junit* framework. The tests are thus highly automatized and the library can be smoothly integrated in classical IDEs. The relatively simple shrinking method which searches a smaller counter-example starting from an initial relatively complex one has been also improved. The library is available at http://gforge.inria.fr/projects/tom.

6.4. Inductive Reasoning

Participant: Sorin Stratulat.

6.4.1. Decision Procedures to Prove Inductive Theorems Without Induction

Automated inductive reasoning for term rewriting has been extensively studied in the literature. Classes of equations and term rewriting systems (TRSs) with decidable inductive validity have been identified and used to automatize the inductive reasoning. In [9], we give procedures for deciding the inductive validity of equations in some standard TRSs on natural numbers and lists. Contrary to previous decidability results, our procedures can automatically decide without involving induction reasoning the inductive validity of arbitrary equations for these TRSs, that is, without imposing any syntactical restrictions on the form of equations. We also report on the complexity of our decision procedures. These decision procedures are implemented in our automated provers for inductive theorems of TRSs and experiments are reported.

6.4.2. Implementing Reasoning Modules in Implicit Induction Theorem Provers

In [30], we detail the integration in SPIKE, an implicit induction theorem prover, of two reasoning modules operating over naturals combined with interpreted symbols. The first integration schema is à la Boyer-Moore, based on the combination of a congruence closure procedure with a decision procedure for linear arithmetic over rationals/reals. The second follows a 'black-box' approach and is based on external SMT solvers. It is shown that the two extensions significantly increase the power of SPIKE; their performances are compared when proving a non-trivial application.

6.4.3. Building Explicit Induction Schemas for Cyclic Induction Reasoning

In the setting of classical first-order logic with inductive predicates, two kinds of sequent-based induction reasoning are distinguished: cyclic and structural. Proving their equivalence is of great theoretical and practical interest for the automated reasoning community. Previously, it has been shown how to transform any structural proof developed with the LKID system into a cyclic proof using the CLKID$^\omega$ system. However, the inverse transformation was only conjectured. In [29], we provide a simple procedure that performs the inverse transformation for an extension of LKID with explicit induction rules issued from the structural analysis of CLKID$^\omega$ proofs, then establish the equivalence of the two systems. This result is further refined for an extension of LKID with Noetherian induction rules. We show that Noetherian induction subsumes the two kinds of reasoning. This opens the perspective for building new effective induction proof methods and validation techniques supported by (higher-order) certification systems integrating the Noetherian induction principle, like Coq.

POSTALE Team

5. New Results

5.1. Highlights of the Year

CovTrack: Agile multi-target multi-threaded realtime tracker We have developed and highly optimized a multi-target tracking system based on covariance tracking algorithm. The complexity of the algorithm – connected to the number of features – can be tuned to fit the processor computation power (with/without SIMD). Moreover the features can be also selected from a large set of features to adapt the algorithm to the scene and the nature of tracking (indoor/outdoor, pedestrian/car,). Some software and algorithmic transforms have been also applied to accelerate the code for scalar/SIMD processors. [20]

The Light Speed Labeling (LSL) algorithm is still the world fastest connected component labeling (CCL) algorithm. We have proposed a new benchmark that performs fair comparisons for such a data-dependent algorithm (that involves Union-Find algorithm optimization combined with memory and control flow optimization). We show that thanks to its run-based approach and its line-relative labeling, LSL is intrinsically more efficient that all State-of-the-Art pixel-based algorithms, whatever the memory management.[23]

5.2. Excalibur: An Autonomic Cloud Architecture for Executing Parallel Applications

Participants: Alessandro Ferreira Leite, Claude Tadonki, Christine Eisenbeis, Tainá Raiol, Maria Emilia Walter, Alba Cristina de Melo.

IaaS providers often allow the users to specify many requirements for their applications. However, users without advanced technical knowledge usually do not provide a good specification of the cloud environment, leading to low performance and/or high monetary cost. In this context, the users face the challenges of how to scale cloud-unaware applications without re-engineering them. Therefore, in this paper, we propose and evaluate a cloud architecture, namely Excalibur, to execute applications in the cloud. In our architecture, the users provide the applications and the architecture sets up the whole environment and adjusts it at runtime accordingly. We executed a genomics workflow in our architecture, which was deployed in Amazon EC2. The experiments show that the proposed architecture dynamically scales this cloud-unaware application up to 10 instances, reducing the execution time by 73compared to the execution in the configuration specified by the user.[25]

5.3. A Fine-grained Approach for Power Consumption Analysis and Prediction

Participants: Alessandro Ferreira Leite, Claude Tadonki, Christine Eisenbeis, Alba Cristina de Melo.

Power consumption has became a critical concern in modern computing systems for various reasons including financial savings and environmental protection. With battery powered de-vices, we need to care about the available amount of energy since it is limited. For the case of supercomputers, as they imply a large aggregation of heavy CPU activities, we are exposed to a risk of overheating. As the design of current and future hardware is becoming more and more complex, energy prediction or estimation is as elusive as that of time performance. However, having a good prediction of power consumption is still an important request to the computer science community. Indeed, power consumption might become a common performance and cost metric in the near future. A good methodology for energy prediction could have a great impact on power-aware programming, compilation, or runtime monitoring. In this paper, we try to understand from measurements where and how power is consumed at the level of a computing node. We focus on a set of basic programming instructions, more precisely those related to CPU and memory. We propose an analytical prediction model based on the

hypothesis that each basic instruction has an average energy cost that can be estimated on a given architecture through a series of micro-benchmarks. The considered energy cost per operation includes both the overhead of the embedding loop and associated (hardware/software) optimizations. Using these precalculated values, we derive a linear extrapolation model to predict the energy of a given algorithm expressed by means of atomic instructions. We then use three selected appli-cations to check the accuracy of our prediction method by comparing our estimations with the corresponding measurements obtained using a multimeter. We show a 9.48% energy prediction on sorting.[27]

5.4. Automated Code Generation for Lattice Quantum Chromodynamics and beyond

Participants: Denis Barthou, Konstantin Petrov, Olivier Brand-Foissac, Olivier Pène, Gilbert Grosdidier, Michael Kruse, Romain Dolbeau, Christine Eisenbeis, Claude Tadonki.

This is ongoing work on a Domain Specific Language which aims to simplify Monte-Carlo simulations and measurements in the domain of Lattice Quantum Chromodynamics. The tool-chain, called Qiral, is used to produce high-performance OpenMP C code from LaTeX sources. We discuss conceptual issues and details of implementation and optimization. The comparison of the performance of the generated code to the well-established simulation software is also made.[17]

5.5. Switchable Scheduling for Runtime Adaptation of Optimization

Participants: Lénaïc Bagnères, Cédric Bastoul.

Parallel applications used to be executed alone until their termination on partitions of supercomputers: a very static environment for very static applications. The recent shift to multicore architectures for desktop and embedded systems as well as the emergence of cloud computing is raising the problem of the impact of the execution context on performance. The number of criteria to take into account for that pur-pose is significant: architecture, system, workload, dynamic parameters, etc. Finding the best optimization for every context at compile time is clearly out of reach. Dynamic optimization is the natural solution, but it is often costly in execution time and may offset the optimization it is en-abling. In this paper, we present a static-dynamic compiler optimization technique that generates loop-based programs with dynamic auto-tuning capabilities with very low overhead. Our strategy introduces switchable scheduling, a family of program transformations that allows to switch between optimized versions while always processing useful computation. We present both the technique to generate self-adaptive programs based on switchable scheduling and experimental evidence of their ability to sustain high-performance in a dynamic environment.[22]

5.6. Efficient distributed randomized algorithms for solving large dense symmetric indefinite linear systems

Participants: Marc Baboulin, Dulceneia Becker, George Bosilca, Anthony Danalis, Jack Dongarra.

Randomized algorithms are gaining ground in high-performance computing applications as they have the potential to outperform deterministic methods, while still providing accurate results. We propose a randomized solver for distributed multicore architectures to efficiently solve large dense symmetric indefinite linear systems that are encountered, for instance, in parameter estimation problems or electromagnetism simulations. Our contribution is to propose efficient kernels for applying random butterfly transformations (RBT) and a new distributed implementation combined with a runtime (PaRSEC) that automatically adjusts data structures, data mappings, and the scheduling as systems scale up. Both the parallel distributed solver and the supporting runtime environment are innovative. To our knowledge, the randomization approach associated with this solver has never been used in public domain software for symmetric indefinite systems. The underlying runtime framework allows seamless data mapping and task scheduling, mapping its capabilities to the underlying hardware features of heterogeneous distributed architectures. The performance of our software is similar to that obtained for symmetric positive definite systems, but requires only half the execution time and half the amount of data storage of a general dense solver. [15]

5.7. Solvers for 3D incompressible Navier-Stokes equations on hybrid CPU/GPU systems

Participants: Yushan Wang, Marc Baboulin, Karl Rupp, Olivier Le Maître, Yann Fraigneau.

We developed a hybrid multicore/GPU solver for the incompressible Navier-Stokes equations with constant coefficients, discretized by the finite difference method. By applying the prediction-projection method, the Navier-Stokes equations are transformed into a combination of Helmholtz-like and Poisson equations for which we describe efficient solvers. We propose a new implementation that takes advantage of GPU accelerators. We present numerical experiments on a current hybrid machine.

5.8. The Numerical Template toolbox: A Modern C++ Design for Scientific Computing

Participants: Pierre Esterie, Joël Falcou, Mathias Gaunard, Jean-Thierry Lapresté, Lionel Lacassagne.

The design and implementation of high level tools for parallel programming is a major challenge as the complexity of modern architectures increases. Domain Specific Languages (or DSL) have been proposed as a solution to facilitate this design but few of those DSL s actually take full advantage of said parallel architectures. In this paper, we propose a library-based solution by designing a C++ DSL s using generative programming: View the MathML source. By adapting generative programming idioms so that architecture specificities become mere parameters of the code generation process, we demonstrate that our library can deliver high performance while featuring a high level API and being easy to extend over new architectures. [18]

5.9. Boost.SIMD: generic programming for portable simdization

Participants: Pierre Esterie, Joël Falcou, Mathias Gaunard, Jean-Thierry Lapresté, Lionel Lacassagne.

Abstract SIMD extensions have been a feature of choice for processor manufacturers for a couple of decades. Designed to exploit data parallelism in applications at the instruction level, these extensions still require a high level of expertise or the use of potentially fragile compiler support or vendor-specific libraries. While a large fraction of their theoretical accelerations can be obtained using such tools, exploiting such hardware becomes tedious as soon as application portability across hardware is required. In this paper, we describe Boost.SIMD, a C++ template library that simplifies the exploitation of SIMD hardware within a standard C++programming model. Boost.SIMD provides a portable way to vectorize computation on Altivec, SSE or AVX while providing a generic way to extend the set of supported functions and hardwares. We introduce a C++standard compliant interface for the users which increases expressiveness by providing a high-level abstraction to handle SIMD operations, an extension-specific optimization pass and a set of SIMD aware standard compliant algorithms which allow to reuse classical C++ abstractions for SIMD computation. We assess Boost.SIMD performance and applicability by providing an implementation of BLAS and image processing algorithms.

5.10. Automatic Task-based Code Generation for High Performance Domain Specific Embedded Language

Participants: Antoine Tran Tan, Joël Falcou, Daniel Etiemble, Harmut Kaiser.

Providing high level tools for parallel programming while sustaining a high level of performance has been a challenge that techniques like Domain Specific Embedded Languages try to solve. In previous works, we investigated the design of such a DSEL-NT2- providing a Matlab-like syntax for parallel numerical computations inside a C++ library. In this paper, we show how NT2 has been redesigned for shared memory systems in an extensible and portable way.[28]

5.11. High Level Transforms for SIMD and low-level computer vision algorithms

Participants: Lionel Lacassagne, Daniel Etiemble, Alain Dominguez, Pascal Vezolle.

This paper presents a review of algorithmic transforms called High Level Transforms for IBM, Intel and ARM SIMD multi-core processors to accelerate the implementation of low level image processing algorithms. We show that these optimizations provide a significant acceleration. A first evaluation of 512-bit SIMD XeonPhi is also presented. We focus on the point that the combination of optimizations leading to the best execution time cannot be predicted, and thus, systematic benchmarking is mandatory. Once the best configuration is found for each architecture, a comparison of these performances is presented. The Harris points detection operator is selected as being *representative* of low level image processing and computer vision algorithms. Being composed of five convolutions, it is more complex than a simple filter and enables more opportunities to combine optimizations. The presented work can scale across a wide range of codes using 2D stencils and convolutions. Such High Level Transforms provide a speedup of x89 on a 2×4 core Intel Xeon processor versus a code that is already SIMDized and OPenMPized.[26]

5.12. What Is the World's Fastest Connected Component Labeling Algorithm?

Participants: Laurent Cabaret, Lionel Lacassagne.

Optimizing connected component labeling is cur- rently a very active research field. Some teams claim to have design the fastest algorithm ever designed. This paper presents a review of these algorithms and a enhanced benchmark that improve classical random images benchmark with a varying granularity set of random images in order to become closer to natural image behavior. Our algorithm, the Light Speed Labeling is from $\times 3.5$ up to $\times 5.3$ faster than the best State-of-the-Art competitor.[23]

5.13. Covariance tracking: architecture optimizations for embedded systems

Participants: Andrés Romero, Lionel Lacassagne, Michèle Gouiffès, Ali Hassan Zahraee.

Covariance matching techniques have recently grown in interest due to their good performances for object retrieval, detection, and tracking. By mixing color and texture information in a compact representation, it can be applied to various kinds of objects (textured or not, rigid or not). Unfortunately, the original version requires heavy computations and is difficult to execute in real time on embedded systems. This article presents a review on different versions of the algorithm and its various applications; our aim is to describe the most crucial challenges and particularities that appeared when implementing and optimizing the covariance matching algorithm on a variety of desktop processors and on low-power processors suitable for embedded systems. An application of texture classification is used to compare different versions of the region descriptor. Then a comprehensive study is made to reach a higher level of performance on multi-core CPU architectures by comparing different ways to structure the information, using single instruction, multiple data (SIMD) instructions and advanced loop transformations. The execution time is reduced significantly on two dual-core CPU architectures for embedded computing: ARM Cortex-A9 and Cortex-A15 and Intel Penryn-M U9300 and Haswell-M 4650U. According to our experiments on covariance tracking, it is possible to reach a speedup greater than 2 on both ARM and Intel architectures, when compared to the original algorithm, leading to real-time execution. [20]

TASC Project-Team

6. New Results

6.1. Highlights of the Year

In the context of the MiniZinc Challenge and in concurrency with 16 other solvers, CHOCO has won three bronze medals in three out of four categories: free search, parallel search and Open class.

6.2. CHOCO

Participants: Jean-Guillaume Fages, Narendra Jussien, Xavier Lorca, Charles Prud'Homme.

- For second consecutive year, CHOCO has participated at the MiniZinc Challenge, an annual competition of constraint programming solvers. In concurrency with 16 other solvers, CHOCO has won three bronze medals in three out of four categories (Free search, Parallel search and Open class). Five versions have been released all year long, the last one (v3.3.0, Dec. 17th) has the particularity to be promoted on Maven Central Repository. The major modifications were related to a simplification of the API but also improvement of the overall solver.
- Within the context of the PhD thesis of Charles Prud'homme [15], a domain specific language that allows prototyping propagation engines was integrated within CHOCO, A paper appears at Constraints.
- Within the context of the PhD thesis of Charles Prud'homme [15], a generic strategy based on explanations for large neighborhood search was designed and integrated within CHOCO. A corresponding paper appears at Constraints [23].
- Within the context of the PhD thesis of Jean-Guillaume Fages, a documented package for graph variables was designed and integrated within CHOCO.

6.3. IBEX Solver

Participants: Gilles Chabert, Alexandre Goldsztejn, Bertrand Neveu, Gilles Trombettoni.

In 2014 the development on IBEX has focused on the following points:

- Rejection test based on first-order conditions (see First Order Rejection Tests For Multiple-Objective Optimization, A. Goldsztejn et al. [42]).
- Q-intersection (see Q-intersection Algorithms for Constraint-Based Robust Parameter Estimation, C. Carbonnel et al., AAAI 2014)

6.4. Packing curved objects

Participants: Nicolas Beldiceanu, Gilles Chabert, Ignacio Salas Donoso.

The development of algorithms to pack curved objects has continued. The filtering algorithm developed in 2013 for generic objects shapes has been published in the CP 2014 conference. Based on this result, we have started the design of a generic (nonlinear) packing solver in 2014. The strategy for packing is directly inspired from a successful approach recently proposed by our project partners (see On solving mixed shapes packing problems by continuous, T. Martinez et al., first BRICS countries congress on Computational Intelligence). It makes use of a stochastic optimization algorithm (CMA-ES) with a fitness function that gives a violation cost and equals zero when objects are all packed. We have generalized their approach by replacing the ad-hoc formulas (for measuring the overlapping between two objects) with an automatic calculation based on our filtering algorithm. The solver is done and the experiments have started.

6.5. Robustness and scheduling

Participants: Nicolas Beldiceanu, Mats Carlsson, Alban Derrien, Arnaud Letort, Thierry Petit, Stéphane Zampelli.

- *Robustness in the Context of the Cumulative Constraint*: This research [33] investigates cumulative scheduling in uncertain environments, using constraint programming. We get a new declarative characterization of robustness, which preserves solution quality which allow adding constraints to the main problem. In this context we adapt the 2013 sweep based algorithm in order to scale and handle several thousand of activities. We highlight the significance of our framework on a crane assignment problem with business constraints.
- *Characterization of Relevant Intervals in the Context of Energetic Reasoning*: Energetic Reasoning (ER) is a powerful filtering algorithm for the Cumulative constraint. Unfortunately, ER is generally too costly to be used in practice. One reason of its bad behavior is that many intervals are considered as relevant, although most of them should be ignored. In the literature, heuristic approaches have been developed in order to reduce the number of intervals to consider, leading to a loss of filtering. We provide a sharp characterization that allows to reduce the number of intervals by a factor seven without any loss of filtering [38].
- *Fix Point over a Conjunction of Scheduling Constraints*: This research introduces a family of synchronized sweep-based filtering algorithms for handling scheduling problems involving resource and precedence constraints. The key idea is to filter all constraints of a scheduling problem in a synchronized way in order to scale better. In addition to normal filtering mode, the algorithms can run in greedy mode, in which case they perform a greedy assignment of start and end times. The filtering mode achieves a significant speed-up over the decomposition into independent cumulative and precedence constraints, while the greedy mode can handle up to 1 million tasks with 64 resource constraints and 2 million precedences. These algorithms were implemented in both CHOCO and SICStus [21].

6.6. Global constraints

Participants: Nicolas Beldiceanu, Jean-Guillaume Fages, Xavier Lorca, Thierry Petit.

- Scalability becomes more and more critical to decision support technologies. In order to address this issue in Constraint Programming, we introduce the family of self-decomposable constraints. These constraints can be satisfied by applying their own filtering algorithms on variable subsets only. We introduce a generic framework which dynamically decompose propagation, by filtering over variable subsets. Our experiments over the cumulative constraint illustrate the practical relevance of self-decomposition [34].
- Consider a constraint on a sequence of variables functionally determining a result variable that is unchanged under reversal of the sequence. Most such constraints have a compact encoding via an automaton augmented with accumulators, but it is unknown how to maintain domain consistency efficiently for most of them. Using such an automaton for such a constraint, we derive an implied constraint between the result variables for a sequence, a prefix thereof, and the corresponding suffix. We show the usefulness of this implied constraint in constraint solving, both by local search and by propagation-based systematic search [25].
- Constraints over finite sequences of variables are ubiquitous in sequencing and timetabling. This led to general modelling techniques and generic propagators, often based on deterministic finite automata (DFA) and their extensions. We consider counter-DFAs (cDFA), which provide concise models for regular counting constraints, that is constraints over the number of times a regular-language pattern occurs in a sequence. We show how to enforce domain consistency in polynomial time for at-most and at-least regular counting constraints based on the frequent case of a cDFA with only accepting states and a single counter that can be increased by transitions. We also show that the satisfaction of exact regular counting constraints is NP-hard and that an incomplete propagator for

exact regular counting constraints is faster and provides more pruning than existing propagators. Finally, by avoiding the unrolling of the cDFA used by cost regular, the space complexity is reduced[26].

6.7. Optimization

Participants: Salvador Abreu, Alejandro Reyes Amaro, Yves Caniou, Philippe Codognet, Daniel Diaz, Jean-Guillaume Fages, Xavier Lorca, Éric Monfroy, Florian Richoux, Louis-Martin Rousseau.

- The traveling salesman problem (TSP) is a challenging optimization problem for CP and OR that has many industrial applications. Its generalization to the degree constrained minimum spanning tree problem (DCMSTP) is being intensively studied by the OR community. In particular, classical solution techniques for the TSP are being progressively generalized to the DCMSTP. Recent work on cost-based relaxations has improved CP models for the TSP. However, CP search strategies have not yet been widely investigated for these problems. The contributions of this research are twofold. We first introduce a natural generalization of the weighted cycle constraint (WCC) to the DCMSTP. We then provide an extensive empirical evaluation of various search strategies. In particular, we show that significant improvement can be achieved via our graph interpretation of the state-of-the-art Last Conflict heuristic. The work was published in the Constraints journal, see the salesman and the tree: the importance of search in CP.
- In the context of nature inspired metaheuristics and its combination with CP, some new work were conducted in the field of ant colony to solve the software project scheduling problem [19], and in the field of the Manufacturing Cell Design Problem [29].
- We implement new algorithmic methods for constraint problems on massively parallel machines. In [18], we propose an extensive study of homogeneous multi-walk parallel scheme for meta-heuristics both with and without communication. The next step will be to look at heterogeneous portfolio approaches where different solvers are looking in parallel for a solution to a given problem.

6.8. Modelling

Participants: Broderick Crawford, Frédéric Lardeux, Éric Monfroy, Ricardo Soto.

- In the framework of conversion of CST set constraints to SAT instances, a filtering engine has been studied and implemented in order to reduce the size of the generated SAT instances.
- From the one hand, CSP is very expressive. On the other hand, SAT solvers can solve huge instances (millions of variables and clauses). We thus worked on the conversion of CSP set constraints into SAT instances [35]. We then focused on the Social Golfer Problem, in order to easily integrate usual improvements (such as symmetry breaking) using our framework [40].

6.9. AI for real time strategy games

Participants: Santiago Ontanon, Florian Richoux, Alberto Uriarte.

We continue to develop an artificial intelligence, AIUR, to play the real time strategy (RTS) game $StarCraft^{tm}$, using both machine learning and constraint-based techniques. AIUR finished 4^{th} over 18 finalists to the $StarCraft^{tm}$ AI competition organized at the conference AIIDE 2014, and 4^{th} over 13 finalists to the competition at CIG 2014. This year, we wrote an ad-hoc CSPsolver to deal with the wall-in optimization problem [36] for StarCraft, and generalized it as a framework enable to handle any kind of CSP/COPmodels representing a RTS-related problem. This framework, named GHOST, helps the user to implement his CSP/COPmodel before solving it with the ready-to-use, already-tuned embedded solver.

AOSTE Project-Team

6. New Results

6.1. Languages, Models of Computation and Metamodeling using logical clock constraints

Participants: Julien Deantoni, Robert de Simone, Frédéric Mallet, Marie Agnès Peraldi Frati.

A revised and updated version of our previous work on UML MARTE Time Model was written in survey textbook form for a larger audience, and published in [38]. Same was done for the more applied specific findings of the ARTEMIS PRESTO European project [39]. Also, a research report finalizing the denotational semantics of the logical clock constraint languages was issued for reference [44].

6.2. Experiments with Architecture and Application modeling

Participants: Robert de Simone, Émilien Kofman, Jean-Vivien Millo, Amine Oueslati, Mohamed Bergach.

We submitted for publication our theoretical results on formal mapping of an application written as a process network dataflow graph onto an abstract architecture model involving a network-on-chip and manycore processor arrays [24].

In the context of the *FUI Clistine* collaborative project (which aims at building a cheap supercomputer by assembling low-cost, general-purpose and network processors interconnected by a time-predictable, on-board network), we considered the issue of classifying general application types, in the fashion inherited from UC. Berkeley's 13 "dwarfs" [46]. Meanwhile,the modeling of desired architecture was slightly postponed due to hesitations from the main industrial partner (that will build the prototype itself). This work was the topic of Amine Oueslati's first year PhD. The classification, and the use of distinct type properties for efficient and natural encoding, was applied on typical application programs provided by partners (Galerkin methods for electromagnetic simulation by the Nachos Inria team, ray-tracing algorithms by the Optis/Simplysim SME design company).

In the context of Mohammed Bergach's CIFRE PhD contract with Kontron Toulon, we conducted an advanced modeling exercice on how to best fit large DFT (Discrete Fourier Transform) modules onto a specific processor architecture (first Intel Sandybridge, then Haswell) that offers computing compromise costs (in performance vs power) between regular CPUs and GPU hardware accelerators. There were two issues: first, how to best dimension the size of the largest FFT block that may be performed locally on a corresponding GPU compute block; second, how to distribute the many such optimal size FFT block needed in a typical radar application, using the GPU and CPU features at the best of their capacity, with account of the slow data transfer latencies across memory banks (to and from the GPU registers).
As a side-effect, people from Kontron are now using and distributing to their customers the FFT GPU libraries with ad-hoc FFT variants matching the GPU block memory sizes. The development, rather lenghty in the case of Sandybridge, was quickly adjusted and ported for Haswell. A new workshop paper is under submission.

6.3. Multiview modeling with performance and power aspects

Participants: Julien Deantoni, Ameni Khecharem, Robert de Simone, Emilien Kofman, Carlos Gomez.

In the context of the ANR HOPE project and The CIM PACA Design Platform, we continued our work on joint modeling and co-simulation of abstract architecture and application (use case scenario) models, together with non-functional aspect views such as performance, power and temperature. The goal of the HOPE project is to consider *hierarchical* {Power/Performance/Temperature} Management Units (MU), and our target is to connect our IDM modeling with dedicated tools such as Synopsys Platform Architect or Docea Power AcePlorer. The IP_Xact interface format for IP blocks is also aimed for compositional assembly representation, including non-functional properties and timing semantics constraints for co-simulation. This work is mostly continued from the former PhD thesis of Carlos Gomez to a new framework by Ameni Khecharem, as part of her PhD. Practical co-simulation trends are also investigated. Currents results were reported in [29]

6.4. Heterogeneous Languages Coordination with Concurrency and Time

Participants: Julien Deantoni, Matias Vara Larsen, Robert de Simone, Frédéric Mallet.

In the context of the ANR GEMOC project and in closely related to the mutiview approach of the team, we focused on how to deal with analysis and simulation of heterogeneous languages. Supporting coordinated use of heterogeneous domain specific languages leads to what we called the globalization of modeling language [22]. Concretely, we proposed to define a language behavioral interface to exhibit the concurrency and time aspects of the semantics of a language. The concurrency and time aspects are described by a formal extension of CCSL, named MoCCML (Model of Concurrency and Communication Modeling Language [45], [28]). Any models that conform such language exhibit a symbolic representation of all its acceptable schedules. Based on this, we shown that it is possible to coordinate heterogeneous models .To avoid redundant model coordination activities, we reified the know-how about model coordination in BCOoL (Behavioral Coordination Operator Language[34]), a language dedicated to language coordination. This work is mainly realized by Matias Vara Larsen, as part of his PhD. In this context, we organize the community around such subject for the second year in an international workshop [43] with an increasing number of participants.

6.5. Performance study of Massively Parallel Processor Array (MPPA) SoC architecture

Participants: Sid Touati, Franco Pestarini.

From a previous collaboration programme, we (Aoste Sophia) possess a MMPA manycore chip, designed and produced by the company Kalray, in Grenoble. The chip integrates 256 cores, composed of 16 clusters (themselves each with 16 cores), and a powerfull network-on-chip interconnect mesh structure. This architecture is oriented towards high performance embedded application, with real time constraints. The cores and NoC were designed to deliver predictable performance.

Our current project, during Franco Pestarini Inria International Intern period, was to test the performance of the NoC, trying to obtain better knowledge of its behavior. We put up a set of microbenchmarks to exercice the network under different specific scenarios (low overhead network traffic, high traffic), and analyzed the experimental results.

We produced a detailed deliverable report explaining under which conditions the NoC could deliver stable and predictable performances. We identified potential configurations where the network becomes unstable (leading to variable and impredictable performances and bandwith).

Meanwhile, the textbook on low-level code optimization, written between Sid Touati and Kalray CTO, appeared in published form [42]. Its content reports on some of the techniques used inside the MPPA compilation environment, and beyond.

6.6. Parametric and Non Parametric Statistics for Code Performance Analysis

Participant: Sid Touati.

This activity is conducted by Sid Touati in collaboration with Julien Worms, an associate professor in Mathematics at the university of Versailles Saint Quentin. It was started under the consideration that the performances of programs are hardly ever represented by a gaussian distribution. So, our purpose here is to study parametric statistics for analyzing the performances of programs. We are interested in modelling program performances by gaussian mixtures (using mixmod method). After a statistical modeling, we deduce multiple performance tests and performance criteria to decide with a high degree of confidence about the "best" program run version. This is still work-in-progress: we are implementing a free software for analysis based on our approach, and we are writing a rather complete research report prior to further publications in conferences and journals.

6.7. Uniprocessor Real-Time Scheduling

Participants: Falou Ndoye, Yves Sorel, Walid Talaboulma.

6.7.1. Real-Time Scheduling with Exact Preemption Cost

Previous years, we worked on schedulability analyses of dependent tasks, executed on a uniprocessor, which take into account the exact preemption cost and more generally the cost of the real-time operating system. Indeed, this cost is composed of the cost of the preemptions and the cost of the scheduler. Our approach is based on an offline real-time schedulability analysis, proved sustainable, that produces a scheduling table. This latter contains the next instants (activation and completion of tasks) when the scheduler will be called, beeing aware of the instants where tasks are preempted and then resumed. This approach allows the schedulability analysis to account preemption costs involved by other preemptions. The scheduling table contains also the address of the next task to execute preventing the scheduler to choose it in the ready tasks list, unlike with classical on-line scheduler. The theoretical results in the uniprocessor case, are given in the Falou Ndoye's PhD thesis [19] defended in April this year. This approach has been implemented through an offline scheduler that is triggered by a timer when this latter is equal to zero and loaded with the next instant contained in the scheduling table, according to a time trigger approach.

We carried out two kinds of implementations. Actually, the first one is a simulation since our time trigger offline scheduler is modelled as a high priority task running uppon an existing operating system. We experimented this approach with Vanilla Linux (not modified) and Linux/Xenomai, real-time versions of the Linux operating system, with low latency caracteristics. Of course, these implementations were only able to show that the theoterical results were correct, but did not provide good real-time performances nor a robust way to measure time without influencing the usefull code. Therefore, we implemented our scheduler on a bare metal ARM968E-S processor based on an ARM9 architecture since it is widespread in the industry world, and we experimented this processor few years ago to determine the cost of classical online schedulers.

For this purpose we used a MCB2929 developpement board, from Keil, containing the LPC2929 SoC including the ARM968E-S, an accurate timer, and various peripherals. The scheduling table is generated offline for a set of tasks, and stored in the memory as an array of couples, each composed of the task to execute and the duration elapsing until the next scheduler call. This duration is used to set the timer counter. When it hits zero it triggers a high priority interruption that is serviced by calling again the scheduler to choose the next couple (task, duration), and so on up to the end of the scheduling table. This will repeat infinitely from the beginning of the scheduling table.

We tested different set of tasks with multiple preemption scenarios, that can yield to deadlines misses. We measured for a **12Mhz** CPU and timer clock frequencies a value of **28** μ**s** for the scheduler cost, and of **1** μ**s** for the preemption cost.

6.8. Multiprocessor Real-Time Scheduling

Participants: Aderraouf Benyahia, Laurent George, Falou Ndoye, Dumitru Potop-Butucaru, Yves Sorel, Meriem Zidouni.

6.8.1. Multiprocessor Partitioned Scheduling with Exact Preemption Cost

Since we chose a multiprocessor partitioned scheduling approach, we can take advantage of the results we obtained in the case of uniprocessor real-time Scheduling accounting for the cost of the real-time operating system, i.e. the cost of preemptions and of the scheduler. From the point of view of the off-line real-time schedulability analysis we only have to consider in addition to activation and completion instants, reception of data instants. This latter instant is determinated by supposing that the cost of every data transfer is known for every possible communication medium. Indeed, when two dependent tasks are allocated to two different processors, the consuming task will have to wait for the data sent by the producing task. The theoretical results in the multiprocessor case, are given in the Falou Ndoye's PhD thesis [19] defended in April this year. We chose the message passing protocol for interprocessor communications achieved through a switched ethernet network. In order to determine precisely the cost of data transfers, we started to investigate the possible approaches to synchronize the send and receive tasks located in two different processors and to schedule them with the other tasks allocated to the same processor. This synchronization protocol will be taken into account to determine the interprocessor communication costs. Concerning these communication costs, we consider FIFO and FIFO* schedulings in the switches, the later is a FIFO scheduling based on the release time of frames at their source node. We have first corrected the trajectory approach (recently shown to be optimistic for corner cases) with FIFO scheduling to compute worst case end-to-end communication costs. Then, we have extended the trajectory approach to FIFO* scheduling. We want to implement our off-line scheduler on every processor of a multiprocessor architecture composed of at least three processors, communicating through an ethernet switch.

Concerning the delay of communications, we consider FIFO and FIFO* schedulings in the switches, the later is a FIFO scheduling based on the release time of frames at their source node. We have first corrected the trajectory approach (recently shown to be optimistic for corner cases) with FIFO scheduling to computed worst case end-to-end communication delays. Then, we have extended the trajectory approach to FIFO* scheduling.

6.8.2. Mutiprocessor Parallel Directed Acyclic Graph (DAG) scheduling

We are interested in studying the hard real-time scheduling problem of parallel Directed Acyclic Graph (DAG) tasks on multiprocessor systems. In this model, a task is defined as a set of dependent subtasks that execute under precedence constraints. The execution order of these subtasks is dynamic, i.e., a subtask can execute either sequentially or in parallel with its siblings based on the decisions of the real-time scheduler. To this end, we analyze two DAG scheduling approaches to determine the execution order of subtasks: the Model Transformation and the Direct Scheduling approaches. We consider global preemptive multiprocessor scheduling algorithms to be used with the scheduling approaches, such as Earliest Deadline First (EDF) and Deadline Monotonic (DM).

6.8.3. Gateway with Modeling Languages for Certified Code Generation

This work was carried out in the P FUI project 8.2.2 We continued the work on the gateway between the P formalism and SynDEx, started the last two years. We have integrated in the gateway the IF and FOR blocks of Simulink that were missing in the functional specification, except for particular cases where the IF block is nested in the FOR block, or the opposite. The integration of the MERGE and MUX blocks are still to be done. We extended the P formalism with architectural elements that SynDEx needs to perform schedulability analyses on functional specifications. These architectural elements are hardware resources (processor, bus, shared memory, router) and timing characteristics (deadline, period, WCET, WCTT). We developed a new part in the gateway which transforms an architectural model described with the P formalism in the input format of SynDEx. We developed also a third part in the gateway which feedbacks the schedulability analysis results obtained with SynDEx (the scheduling table) and stores them into models described with the P formalism. Finally, we have collaborated with the industrial partners to test our gateway on their use cases.

6.8.4. SynDEx updates

The first tests on the alpha version of SynDEx V8, released last year, shown some bugs that we fixed. This first release did not include a code generator. Thus, we worked to interface the distributed real-time embedded code generator of SynDEx V7 with SynDEx V8.

6.9. Probabilistic Real-Time Systems

Participants: Liliana Cucu-Grosjean, Robert Davis, Adriana Gogonel, Codé Lo, Dorin Maxim, Cristian Maxim.

The advent of complex hardware, in response 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 [23] we have described the vision of FP7 IP PROXIMA, project that is interested in the introduction of randomization of the architectures at cache level. For this type of architecture static probabilistic timing analysis is possible [20] by providing bounds on the probabilistic execution time of a task. An industrial case study from avionics is detailed in [32]. Such distribution is then used as input for probabilistic scheduling as described in [37], [36].

This year we have also provided a complete state of the art of the probabilistic real-time systems in [17].

6.10. Off-line (static) mapping and WCET analysis of real-time applications onto NoC-based many-cores

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

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

In past years we have identified and compared the hardware mechanisms supporting precise timing analysis and efficient resource allocation in existing NoCs. We have determined that the NoC should ideally provide the means of enforcing a global communications schedule that is computed off-line and which is synchronized with the scheduling of computations on CPU cores. Furthermore, if in addition the computation and memory resources of the MPSoC have support for real-time predictability, then parallel applications can be developed that allow very precise WCET analysis of parallel code. WCET analysis of parallel code is joint work with Isabelle Puaut of Inria, EPI ALF.

This year we have completed our mapping (allocation and scheduling) and code generation technique and tool for NoC-based MPSoCs. NoCs pose significant challenges to both on-line (dynamic) and off-line (static) real-time scheduling approaches [25]. They have large numbers of potential contention points, have limited internal buffering capabilities, and network control operates at the scale of small data packets. Therefore, efficient resource allocation requires scalable algorithms working on hardware models with a level of detail that is unprecedented in real-time scheduling.

We considered a static (off-line) scheduling approach, and we targeted massively parallel processor arrays (MPPAs), which are MPSoCs with large numbers (hundreds) of processing cores. We proposed a novel allocation and scheduling method capable of synthesizing such global computation and communication schedules covering all the execution, communication, and memory resources in an MPPA. To allow an efficient use of the hardware resources, our method takes into account the specificities of MPPA hardware and implements advanced scheduling techniques such as pre-computed preemption of data transmissions[26] and pipelined scheduling [21].

Our method has been implemented within the Lopht tool presented in section 5.4 , and first results are presented in [26], [25], and in extenso in the PhD thesis of manel Djemal [18]. One of the objectives of the starting CAPACITES project is the evaluation of the possibility of porting Lopht and the WCET analysis technique for parallel code onto the Kalray MPPA platform.

6.11. Real-time scheduling and code generation for time-triggered platforms

Participants: Dumitru Potop Butucaru, Thomas Carle, Raul Gorcitz, Yves Sorel.

We have continued this year the work on real-time scheduling and code generation for time-triggered platforms. Much of this work was carried out as part of a bilateral collaboration with Airbus DS and the CNES, which fund the post-doctorate of Raul Gorcitz, and in our collaboration with the IRT SystemX, project FSF.

The objective is to facilitate the development of complex time-triggered systems by automating the allocation, scheduling, and code generation steps. We show that full automation is possible while taking into account all the specification elements required by a complex, real-life embedded control system. The main originality of our work is that it takes into account at the same time multiple complexity elements: functional specifications with conditional execution and multiple modes and various types of non-functional properties: real-time (release dates, deadlines, major time frame, end-to-end flows), ARINC 653 partitioning (which we can fully or partially synthesize), task preemptability, allocation. Our algorithms allow the automatic allocation and scheduling onto multi-processor (distributed) systems with a global time base, taking into account communication costs.

While the past years were mainly dedicated to the development of this scheduling and code generation technique, this year the technique and the associated tool have matured enough to allow the publication of the first results concerning the optimized scheduling algorithms [21] and its application on large case studies. Ongoing work by the post-doc Raul Gorcitz, funded by Airbus DS and the CNES aims at evaluating the applicability of our methods on embedded platforms that are being considered for the future european space launchers. The Lopht tool is also used in the IRT SystemX, project FSF as part of the proposed design flow. All extensions have been implemented in the Lopht tool. All this work has been presented *in extenso* in the PhD thesis of Thomas Carle [16].

CONVECS Project-Team

6. New Results

6.1. New Formal Languages and their 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 CONVECS for industrial case studies and applications (see § 6.5) and serves also in university courses on concurrency, in particular at ENSIMAG (Grenoble) and at Saarland University.

6.1.1. Translation from LNT to LOTOS

Participants: Hubert Garavel, Frédéric Lang, Wendelin Serwe.

In 2014, the translator from LNT to LOTOS was further improved. In addition to bug fixes and removal of incorrect warnings emitted by the translator itself or by the C compiler on the generated code, the following enhancements have been brought: the LNT language was extended with a "`!representedby`" pragma for processes, and a "`only if`" statement to concisely express guarded commands; the translator now performs static analysis and warns about unused variables, unused "`in`" or "`in out`" parameters, useless (deterministic or nondeterministic) assignments to variables, "`in out`" parameters that are never assigned, and dubious synchronizations between processes; checks for underflow/overflow on natural and integer numbers are now activated by default. The translator also generates better LOTOS code, and the LNT reference manual was shortened and updated in many places.

6.1.2. Translation from LOTOS to Petri nets and C

Participants: Hubert Garavel, Wendelin Serwe.

The LOTOS compilers CAESAR and CAESAR.ADT, which were once the flagship of CADP, now play a more discrete role since LNT (rather than LOTOS) has become the recommended specification language of CADP. Thus, CAESAR and CAESAR.ADT are mostly used as back-end translators for LOTOS programs automatically generated from LNT or other formalisms such as Fiacre, and are only modified when this appears to be strictly necessary.

In 2014, the CAESAR compiler has been modified to tolerate LOTOS specifications that would be normally rejected under the ISO/IEC 8807 standard definition of LOTOS. The first change extends the visibility scope of local definitions when the global definitions are empty. The second change uses the type information of process definitions to better resolve overloading ambiguities in expressions passed as actual parameters to process calls.

Conversely, CAESAR was made stricter by rejecting at compile-time LOTOS specifications containing out-of-bound constants, even if such constants are never used.

Performance has been increased by adding or strengthening a number of optimizations concerning, e.g., internal data structures, Boolean guards that can be statically evaluated, values belonging to singleton sorts, disconnected or otherwise unreachable Petri net places and transitions, etc.

The CAESAR.BDD tool of CADP, which analyzes hierarchical Petri nets generated from higher-level specifications (e.g., LOTOS or LNT) has been significantly enhanced. The semantic model accepted by CAESAR.BDD has been made more general and given the new name of NUPN (*Nested-Units Petri Nets*). The definition and theoretical properties of NUPN have been formalized.

The textual syntax for NUPN has been extended with pragmas intended to retain useful properties of non-ordinary and/or non-safe Petri nets translated to NUPN. An XML syntax for NUPN (compatible with the ISO standard PNML for the representation of Petri nets) has been defined and adopted by the Model Checking Contest [0]. A translator from PNML to NUPN has been developed at LIP6 (Paris, France).

[0] http://mcc.lip6.fr/nupn.php

The CAESAR.BDD tool has been updated accordingly, and extended to perform stricter checks and compute more structural and behavioral properties of NUPN models. CAESAR.BDD has been intensively used to correct the descriptions of the Model Checking Contest benchmarks: a first campaign (January-February 2014) detected 9 errors in structural properties and 8 errors in behavioral properties, and a second campaign (April 2014) revealed 23 more errors. CAESAR.BDD has also been used to automatically generate new benchmarks, together with their descriptions.

6.1.3. Translation from GRL to LNT

Participants: Fatma Jebali, Frédéric Lang, Eric Léo, Radu Mateescu.

In the context of the Bluesky project (see § 8.1.2.1), we study the formal modeling of GALS (*Globally Asynchronous, Locally Synchronous*) systems, which are composed of several synchronous subsystems evolving cyclically, each at its own pace, and communicating with each other asynchronously. Designing GALS systems is challenging due to both the high level of (synchronous and asynchronous) concurrency and the heterogeneity of computations (deterministic and nondeterministic). To bring our formal verification techniques and tools closer to the GALS paradigm, we designed a new formal language named GRL (*GALS Representation Language*), as an intermediate format between GALS models and purely asynchronous concurrent models. GRL combines the main features of synchronous dataflow programming and asynchronous process calculi into one unified language, while keeping the syntax homogeneous for better acceptance by industrial GALS designers. GRL allows a modular composition of synchronous systems (blocks), environmental constraints (environments), and asynchronous communication mechanisms (mediums), to be described at a level of abstraction that is appropriate to verification. GRL also supports external C and LNT code.

In 2014, we have continued to enhance the syntax and the formal semantics of GRL. We have written a detailed research report (82 pages) [25] containing the complete definition of the syntax, static semantics, and dynamic semantics (in the form of structural operational semantics rules), and also illustrating the checking of dynamic semantics rules on several examples of GRL programs. A paper describing GRL has been published in an international conference [14].

To equip GRL with verification features, we formally defined a translation from GRL to LNT. GRL blocks are translated into LNT functions, possibly encapsulated within LNT wrapper processes to enable asynchronous communication, whereas GRL environments and mediums are directly translated into LNT processes. The asynchronous composition of blocks, environments, and mediums is translated to an LNT parallel composition of the corresponding processes.

Using the SYNTAX and LOTOS NT compiler construction technology [44], we have developed a translator named GRL2LNT (about $25,000$ lines of code), allowing an LNT program to be generated from a GRL specification automatically. GRL2LNT performs the lexical and syntactic analysis of GRL programs, together with almost all static semantic checks specified in its formal definition [25]. A stable version of GRL2LNT has been released in 2014. Additionally, we have developed an OPEN/CAESAR-compliant compiler GRL.OPEN (based on GRL2LNT and LNT.OPEN), which makes possible the on-the-fly exploration of the LTS underlying a GRL specification using CADP. We have also built a test base containing about 250 (correct and incorrect) GRL programs, and used it for non-regression testing of GRL2LNT. The correct GRL programs represent about $7,000$ lines of code and produce about $18,000$ lines of LNT code after translation using GRL2LNT.

A paper describing the formal verification of GALS systems using GRL and CADP, with a focus on the translation from GRL to LNT, has been submitted to an international conference [28].

6.1.4. Coverage Analysis for LNT

Participants: Gwen Salaün, Lina Ye.

In the classic verification setting, the designer has a specification of a system in a value-passing process algebra, a set of temporal properties to be verified on the corresponding LTS model, and a data set of examples (test cases) 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.

We propose a new framework for debugging value-passing process algebra through coverage analysis and we illustrate our approach with LNT. We define several coverage notions before showing how to instrument the specification without affecting original behaviors. Our approach helps one to improve the quality of a data set of examples used for validation purposes, but also to find ill-formed decisions, dead code, and other errors in the specification. We have implemented a tool for automating our debugging approach, and applied it to several real-world case studies in different application areas.

In 2014, a paper has been accepted in an international conference [19].

6.1.5. Other Language Developments

Participants: Hugues Evrard, Hubert Garavel, Frédéric Lang, Eric Léo, Wendelin Serwe.

The ability to compile and verify formal specifications with complex, user-defined operations and data structures is a key feature of the CADP toolbox since its very origins. A long-run effort has been recently undertaken to ensure a uniform treatment of types, values, and functions across all the various CADP tools.

In 2014, convergence between the LOTOS, LNT, BCG, and XTL data-type libraries has been increased by defining common libraries for eight predefined types: Boolean, Natural, Integer, Real, Character, String, Raw, and Gate. These libraries gather in the same place definitions of types, constants, and functions that were previously disseminated across different tools. Additionally, systematic checks for underflows and overflows have been set for the Natural and Integer types. Also, unprintable characters and C-like escape sequences are now uniformly handled by the Character, String, and Raw types.

To support the LNT language in the Emacs/XEmacs, jEdit, and Vim editors, configuration files have been added or updated, which provide for syntax highlighting/coloring, and enable autocompletion in Emacs using YASnippet.

6.2. Parallel and Distributed Verification

6.2.1. Distributed Code Generation for LNT

Participants: Hugues Evrard, Frédéric Lang.

Rigorous development and prototyping of a distributed verification algorithm in LNT involves the automatic generation of a distributed implementation. For the latter, a protocol realizing process synchronization is required. 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.

Using a synchronization protocol that we verified formally in 2013, we developed a prototype distributed code generator, named DLC (*Distributed LNT Compiler*), which takes as input the model of a distributed system described as a parallel composition of LNT processes.

In 2014, we continued the development of DLC. We improved the performances of DLC generated code by reducing the number of protocol messages when one or several processes are ready on a single gate. We experimented this optimization on a set of processes running on different computers and synchronizing all together on a single barrier interaction (i.e., all processes are ready on a single gate). In this situation, DLC now generates code that is faster than Java or Erlang.

The distributed program generated by DLC would be of little interest if it could not interact with its environment (e.g., users through human-computer interfaces, or other systems, such as databases, Web services, etc.). Therefore, we designed a mechanism to embed user-defined C functions, called *hook functions*, into the code generated by DLC. Hook functions are triggered on events related to actions in the system. This allows system actions to be, e.g., monitored by the user or controlled by external conditions. Using hook functions, the code generated by DLC can thus both take an account of and have an effect on its environment.

In order to demonstrate DLC on a real-world example, we applied it to the recent Raft [0] consensus algorithm [60]. We wrote an LNT specification of a simple key-value store made fault tolerant by replication of commands using the Raft consensus algorithm. During the modeling phase, we found a missing transition in the TLA+ specification of the protocol. We signaled it to the authors [0], who corrected the TLA+ specification. We used hook functions to implement interaction with the replicated store from external clients. The distributed implementation generated by DLC was successfully tested on clusters of the Grid5000 platform. We presented an overview of DLC, the hook functions and the Raft experiment in an article that has been accepted for publication in an international conference [12].

6.3. Timed, Probabilistic, and Stochastic Extensions

6.3.1. Model Checking for Extended PCTL

Participants: Hubert Garavel, Radu Mateescu, Jose Ignacio Requeno.

In the context of the SENSATION project (see § 8.2.1.1), we study the specification and verification of quantitative properties of concurrent systems.

In 2014, we defined an extension of PCTL (*Probabilistic Computation Tree Logic*) [49] with the manipulation of data values and actions. This logic is interpreted on extended DTMCs (*Discrete-Time Markov Chains*) containing visible transitions, labeled with channel names and data values, in addition to probabilistic transitions. Extended PCTL makes possible the specification of temporal properties involving discrete time, probabilities, and data values.

We devised a prototype model checker for extended PCTL in the form of an XTL library describing the denotational semantics of all PCTL operators (both primitive and derived ones), accompanied by external C code implementing the algorithms for LTS exploration and numerical computation of probabilities. The high-level programming language constructs of XTL (iterators, sets in comprehension, parameterized macro-definitions) allowed us to easily implement the advanced features (filters on arithmetic and logical operators, computation of probabilities, experiments over data series, etc.) of established probabilistic model checkers, such as PRISM [54]. Also, the manipulation of data values in XTL allows one to specify properties in which probabilities and discrete time deadlines depend on the values of state variables, a feature currently not provided by PRISM.

To experiment and cross-check our extended PCTL library w.r.t. PRISM, we developed an automated translator from the (state-based) DTMCs used by PRISM into the (action-based) DTMCs in BCG format used by CADP. State information is represented by means of special self-looping transitions containing the values of state variables, which are properly handled during the evaluation of probabilistic temporal operators.

The experiments we performed with our extended PCTL library on various examples of DTMCs (produced from communication protocols, chemical reactions, hazard games, etc.) showed a performance comparable to (explicit-state) PRISM for pure PCTL formulas.

Furthermore, in addition to many bug fixes, the XTL compiler and its XTL_EXPAND preprocessor have been strengthened to better detect and report potential mistakes in source XTL specifications. In particular, vacuity checks have been introduced, which warn the user when no label in a BCG graph has the right number of fields or the appropriate field types to satisfy an XTL label match expression (previously, this expression would silently evaluate to false).

The type checking system of XTL and its list of predefined functions have been extended to support the new Natural and Raw types of the BCG format, and to properly distinguish between Natural and Integer values, and Raw and String values, while achieving a high degree of backward compatibility. In particular, XTL now uses type information from the BCG labels to better solve overloading in label offers, so that certain XTL programs that were formerly invalid are now accepted. Finally, it is now possible to use the predefined types and functions of XTL when defining temporal operators directly using external C code.

[0] http://raftconsensus.github.io
[0] https://groups.google.com/forum/#!topic/raft-dev/yu-wOUx-gnA

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

6.4.1. Property-Dependent Reductions for the Modal Mu-Calculus

Participant: Radu Mateescu.

In collaboration with Anton Wijs (Technical University of Eindhoven), we proposed a new method for enhancing the performance of model checking a temporal formula on an LTS by reducing the LTS as much as possible depending on the formula prior to (or simultaneously with) the verification. Given an LTS and a formula, the method consists of two steps:

- The maximal set of actions that one can hide (i.e., rename into the internal action τ) in the LTS without disturbing the interpretation of the formula is computed, and those actions are hidden in the LTS. This works for any formula of the full modal μ-calculus (i.e., of arbitrary alternation depth) and provides the highest potential for reducing the LTS, and hence for improving verification performance, w.r.t. that formula.
- The LTS is reduced modulo an equivalence relation preserving the formula. The reduction can be done before verification, either by constructing the LTS explicitly and using the direct minimization features provided by the BCG_MIN tool, or by constructing the minimized LTS incrementally using the compositional verification features provided by EXP.OPEN and SVL. The reduction can be also done simultaneously during verification, by detecting τ-confluent transitions and prioritizing them against their neighbors.

We defined a μ-calculus fragment, named Lμ-dsbr, and shown its adequacy w.r.t. divergence-sensitive branching bisimulation (divbranching for short). We also shown that Lμ-dsbr is equally expressive to the μ-ACTL$\setminus$X logic, an extension of ACTL$\setminus$X (Action-based CTL without the next time operator) with fixed point operators [39], [40]. This result also implies the adequacy w.r.t. divbranching of μ-ACTL$\setminus$X, which was previously shown to be adequate w.r.t. strong bisimulation.

We experimented our method using the EVALUATOR model checker on various examples of protocols and distributed systems, by specifying the temporal properties in MCL and reducing the LTSs modulo strong and divbranching bisimulation. The experiments showed performance enhancements both in execution time (reduction by a factor 4 for strong bisimulation and 20 for divbranching) and memory consumption (reduction by a factor 2 for strong bisimulation and 5 for divbranching).

We also built a prototype MCL library regrouping the temporal operators of ACTL$\setminus$X (which were already present in CADP) and the modal and temporal operators of Lμ-dsbr (which were newly added). Used in conjunction with the Boolean and fixed point operators of MCL, the operators of this library can be used to specify temporal formulas adequate w.r.t. divbranching, which allows one to reduce the LTS modulo this equivalence (after applying maximal hiding) and to increase the performance of verification accordingly. An article has been published in an international journal [8].

6.4.2. Compositional Verification

Participants: Hubert Garavel, Frédéric Lang.

The CADP toolbox contains various tools dedicated to compositional verification, among which EXP.OPEN, BCG_MIN, BCG_CMP, and SVL play a central role. EXP.OPEN explores on the fly the graph corresponding to a network of communicating automata (represented as a set of BCG files). BCG_MIN and BCG_CMP respectively minimize and compare behavior graphs modulo strong or branching bisimulation and their stochastic extensions. SVL (*Script Verification Language*) is both a high-level language for expressing complex verification scenarios and a compiler dedicated to this language.

In 2014, we corrected 2 bugs in EXP.OPEN, 6 bugs in BCG_MIN and BCG_CMP, and 5 bugs in SVL. We also enhanced these tools as follows:

- We corrected the diagnostic generation algorithm of BCG_CMP, which sometimes generated irrelevant diagnostics.

- We improved the messages displayed by SVL and EXP.OPEN when generating an LTS from a composition expression using the *smart reduction* strategy [38], so that the user can follow more easily the selected composition order.
- Following the recent progress made on the development of the language LNT (see § 6.1), the syntax of the SVL and EXP languages for comments, gate typing, and the "`par`", "`hide`", "`rename`", "`cut`", and "`prio`" operators was extended to be compatible with the syntax of LNT. This enables composition expressions (including comments, channel typing, etc.) copied from LNT programs to be pasted in SVL scripts while requiring as few syntactic changes as possible.
- The "`verify`" operator has been generalized to give access to all three model checkers of CADP (EVALUATOR 3, EVALUATOR 4, and XTL). A new statement "`|=`" has been added to SVL, which enables MCL and XTL formulas to be directly written in an SVL script, rather than being stored in external files.
- To provide for requirements expression and traceability in SVL, we introduced two new statements, "`property`" and "`check`", which increase the readability and good structure of SVL scripts by allowing to define and verify properties, each of which is given a name, instantiable parameters, an informal textual description, and (optionally) an expected truth value.
- We updated several demo examples of CADP in order to illustrate the above extensions.

6.4.3. On-the-Fly Test Generation

Participants: Hubert Garavel, Radu Mateescu, Wendelin Serwe.

In the context of the collaboration with STMicroelectronics, we study techniques for testing if a (hardware) implementation is conform to a formal model described in LNT. Our approach is inspired by the theory of conformance testing [62], as implemented for instance in TGV [53] and JTorX [33]. We have developed two prototype tools to support this approach. The first tool implements a dedicated OPEN/CAESAR-compliant compiler for the particular asymmetric synchronous product between the model and the test purpose. The second tool, based on slightly extended generic components for graph manipulation (τ-compression, τ-confluence reduction, determinization) and resolution of Boolean equation systems, generates the complete test graph (CTG), which can be used to extract concrete test cases or to drive the test of the implementation. The principal advantage of our approach compared to existing tools is the use of LNT for describing test purposes, which facilitates the manipulation of data values.

In 2014, we developed a third prototype tool that takes as input a CTG and extracts either a single test case (randomly chosen or the first encountered one), or the set of *all* test cases. This prototype tool was used in the case study with STMicroelectronics (see § 6.5.1).

The test-generation tool TGV has been streamlined by removing some obsolete options and replacing a large part of its code by calls to the standard CADP libraries. TGV has been made faster, it now supports the latest version of the AUT format, and ensures that test purposes provided in the BCG format are deterministic. The manual page has been updated and completed.

6.4.4. Other Component Developments

Participants: Soraya Arias, Hubert Garavel, Frédéric Lang, Radu Mateescu.

The AUT textual format for CADP for storing LTSs was extended to support recent languages (such as LNT and the PseuCo language developed at Saarland University) that manipulate character-string values. The AUT format, which was defined in the late 80s, did not support such values. A new version 2014 of the AUT format has been defined, which solves this problem and maintains backward compatibility. All the CADP tools that read or write AUT files have been updated accordingly.

The BCG format of CADP for storing LTSs has been upgraded with the advent of a new version 1.2, which replaces version 1.1 released in 2009. New predefined types have been added to BCG to express the difference between unsigned and signed integers, and between character strings and untyped raw-data values. The new version of the BCG format is also more compact and now uses variable-length encoding for strings. The rules for label parsing of the BCG_WRITE interface have been extended, and BCG_IO now supports version 2014 of the AUT format. The intrinsic difficulty of these changes was to preserve the backward compatibility with the BCG files generated over the last twenty years.

To simplify the installation of CADP on Windows systems, we studied an alternative execution environment based on Gnuwin32 and MinGW/Msys rather than Cygwin. Preliminary changes have been brought to CADP scripts to undertake such a migration.

6.5. Real-Life Applications and Case Studies

6.5.1. ACE Cache Coherency Protocol

Participants: Abderahman Kriouile, Radu Mateescu, Wendelin Serwe.

In the context of a CIFRE convention with STMicroelectronics, we study 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 techniques grows exponentially. As an alternative, we study formal verification.

We focused on the ACE (AXI Coherency Extensions) cache coherency protocol, a system-level coherency protocol proposed by ARM [29]. In previous years, we developed a formal LNT model (about $3,400$ lines of LNT) of a system consisting of an ACE-based 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 (currently, a representative subset of 15 operations), including an abstract operation that represents any other ACE operation. We handled the global requirements of the ACE specification using a constraint oriented programming style, i.e., by representing each global requirement as a dedicated process observing the global behavior and inhibiting incorrect executions. We also specified temporal properties expressing cache coherence, data integrity, and successful completion of each transaction. Note that the former property required to transform state-based properties into action-based ones, by adding information about the cache state to the actions executed by the cache.

In 2014, we exploited the formal model to improve the validation of the architecture under design at STMicroelectronics. In a first step, we studied the sanity (soundness and completeness) of an industrial interface verification unit, consisting of a list of so-called *formal checks*. After modeling each check in LNT, we used the BISIMULATOR tool to verify that each check is an overapproximation of the corresponding projection of the formal model. When we tried to establish that the parallel composition of all checks is an overapproximation of the projection of the formal model, we discovered a missing check (a particular channel did not occur in any of the checks).

In a second step, we studied the derivation of system level test cases, using a two-phase approach:

- In the first phase, abstract test cases were extracted automatically from the formal model using a prototype tool (see § 6.4). To circumvent the complexity of extracting test cases from the complete model, we proposed an iterative approach based on the automatic selection of a comprehensive set of interesting scenarios leading to LTSs of tractable size. The selection of the interesting scenarios relies on the counterexamples provided by the EVALUATOR model checker for the properties of coherence and data integrity.
- In the second phase, the abstract test cases were translated into the input format of an industrial test bench in charge of refining them into concrete test cases to be executed on the RTL (*Register Transfer Level*) description of the architecture under study. Experiments with manually translated abstract test cases led to the early discovery of bugs in commercial verification blocks, which could therefore be corrected before their use became critical in the development process.

The tests derived from the formal model increased the coverage of problematic features of some blocks used in the architecture. In particular, our approach was able to detect a limitation concerning data integrity 20 months before it was confirmed by classical methods, and our methodology provides all the scenarios triggering the limitation.

This work led to a publication accepted in an international conference [15]. Also, a large Petri net derived from our LNT model was provided as benchmark example for the Model Checking Contest.

6.5.2. Formal Verification of BPMN Processes

Participants: Radu Mateescu, Gwen Salaün, Lina Ye.

A business process is a set of structured, related activities that aims at fulfilling a specific organizational goal for a customer or market. An important metric when developing a business process is its degree of parallelism, i.e., the maximum number of tasks that are executable in parallel in that process. The degree of parallelism determines the peak demand on tasks, providing a valuable guide for the problem of resource allocation in business processes.

In 2014, we investigated how to automatically measure the degree of parallelism for business processes, described using the BPMN standard notation. To this aim, we defined a formal model for BPMN processes in terms of LTSs, which are obtained through an encoding in LNT. We then proposed an approach for automatically computing the degree of parallelism by using model checking of parameterized MCL formulas and dichotomic search. We developed a prototype tool for automating this check and we applied it successfully to more than one hundred BPMN processes.

This work led to a publication in an international conference [16].

6.5.3. Stability of Asynchronously Communicating Systems

Participants: Gwen Salaün, Lina Ye.

Analyzing communicating systems that interact asynchronously via reliable FIFO buffers is an undecidable problem. A typical approach is to check whether the system is bounded, and if not, the corresponding state space can be made finite by limiting the presence of communication cycles in behavioral models or by fixing buffer sizes.

We followed a different approach, which aims at analyzing communicating systems without restricting them by imposing any arbitrary bounds. These systems are likely to be unbounded and therefore result in infinite state spaces. We introduce a notion of stability and prove that once the system is stable for a specific buffer bound (called stability bound), it remains stable whatever larger bounds are chosen for the buffers. This enables us to check certain properties on the (finite-state) system obtained for the stability bound and to ensure that the system will preserve them whatever larger bounds are used for buffers.

We have also proven that computing the stability bound is in general undecidable, and we proposed a semi-algorithm that successfully computes the stability bounds for many typical examples of communicating systems using heuristics and equivalence checking. This work is described in a research report [27].

6.5.4. Deployment and Reconfiguration Protocols for Cloud Applications

Participants: Rim Abid, Gwen Salaün.

In the context of the OpenCloudware project (see § 8.1.1.1), 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 applications composed of interconnected software components running on distinct virtual machines (VMs). Setting up, (re)configuring, and monitoring these applications involve intricate management protocols, which fully automate these tasks while preserving application consistency as well as some key architectural invariants.

In 2014, we extended the specification of the self-deployment protocol to support VM failures. This led to a publication in an international conference [11], of which an extended version is under preparation for submission to an international journal.

We also worked on the dynamic reconfiguration of cloud applications. As a first attempt, we proposed to design this protocol using a publish-subscribe communication model [32]. In 2014, we improved the protocol to also support VM failures, and drastically validated the corresponding LNT specification using model checking. A paper presenting these results was submitted to an international journal. In parallel, we studied a version of this protocol where the different participants interact asynchronously via FIFO buffers. This led to a publication in an international conference [10].

As a new line of work, we undertook the study of controller synthesis techniques for the coordination of autonomic managers in asynchronous environments. Our approach relies on an encoding into LNT and on the application of several operations on automata (synchronous products, hiding, reduction) for synthesizing the corresponding controller using CADP tools. We also proposed automated techniques for generating Java code from an abstract model of the controller. For validation purposes, we applied our approach to real-world three-tier Web applications and showed that the introduction of a controller allows one to avoid erroneous situations due to the absence of coordination between autonomic managers.

6.5.5. 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 GRL (see § 6.1.3), which enables the connection to testing and verification tools covering the synchronous and asynchronous aspects.

In 2014, we have developed a set of GRL libraries implementing about 40 of the function blocks present in the PLC programming tool of Crouzet, to facilitate the integration of GRL in the PLC software design process. These function blocks include (among others) logic and comparison functions, timers, triggers, and counters. These GRL libraries have been used to model large applications provided by Crouzet. The GRL2LNT and GRL.OPEN tools (see § 6.1.3) provide a direct connection to all verification functionalities of CADP, in particular model checking and equivalence checking.

Regarding model checking, we have studied existing work in the verification of synchronous systems and GALS systems. We have identified a set of typical patterns of temporal properties (e.g., deadlocks, safety, liveness) relevant for GALS systems. These property patterns have been specified using MCL and checked on a set of feature-rich GRL examples using GRL.OPEN and EVALUATOR.

Regarding equivalence checking, the purpose is to compare the behavior of a GALS system with its *service*, which represents its desired observable behavior, modulo a suitable equivalence relation. We have studied existing work in equivalence checking for GALS systems and we have investigated how to formally define the expected service of a GALS system at the appropriate level of expressiveness and abstraction, which requires a careful identification of the observable actions corresponding to the interactions between the GALS system and its physical environment. We have modeled several examples of GALS systems in GRL, and experimented the definition of appropriate services and their usage for equivalence checking by means of GRL.OPEN and BISIMULATOR.

The validation approach we promote, together with our colleagues from the LCIS laboratory (Valence) in the Bluesky project, led to a common publication in a national conference [21].

6.5.6. EnergyBus Standard for Connecting Electric Components

Participants: Hubert Garavel, Wendelin Serwe.

The EnergyBus [0] is an upcoming industrial standard for electric power transmission and management, based on the CANopen field bus. It is developed by a consortium assembling all major industrial players (such as Bosch, Panasonic, and Emtas) in the area of light electric vehicles (LEV); their intention is to ensure interoperability between all electric LEV components. At the core of this initiative is a universal plug integrating a CAN-Bus [0] with switchable power lines. The central and innovative role of the EnergyBus is to manage the safe electricity access and distribution inside an EnergyBus network.

In the framework of the European FP7 project SENSATION (see § 8.2.1.1) a formal specification in LNT of the main EnergyBus protocols is being developed by Alexander Graf-Brill and Holger Hermanns at Saarland University [48], with the active collaboration of CONVECS.

In 2014, our joint work with Saarland University on the modeling, verification, and test case generation for the EnergyBus standard led to a common publication [13].

6.5.7. Graphical User-Interfaces and Plasticity

Participants: Hubert Garavel, Frédéric Lang, Raquel Oliveira.

In the context of the Connexion project (see § 8.1.1.2) and in close collaboration with Gaëlle Calvary, Eric Ceret, and Sophie Dupuy-Chessa (IIHM team of the LIG laboratory), we study the formal description and validation of graphical user-interfaces using the most recent features of the CADP toolbox. The case study assigned to LIG in this project is a prototype graphical user-interface [36] designed to provide human operators with an overview of a running nuclear plant. The main goal of the system is to inform the operators about alarms resulting from faults, disturbances, or unexpected events in the plant. Contrary to conventional control rooms, which employ large desks and dedicated hardware panels for supervision, this new-generation interface uses standard computer hardware (i.e., smaller screen(s), keyboard, and mouse), thus raising challenging questions on how to best provide synthetic views of the plant status. Another challenge is to introduce plasticity in such interface, so as to enable several supervision operators, including mobile ones outside of the control room, to get accurate information in real time.

We formally specified this new-generation interface in LNT, encompassing not only the usual components traditionally found in graphical user-interfaces, but also a model of the physical world (namely, a nuclear reactor with various fault scenarios) and a cognitive model of a human operator in charge of supervising the plant. Also, several desirable properties of the interface have been expressed in MCL and verified on the LNT model using CADP. This led to a publication in an international conference [17].

In 2014, we continued our activity along several directions. The LNT specification was matured in various respects. As a result of several interactions with EDF, the specification was enhanced with a more realistic representation of the plant (currently $5,358$ lines of LNT code). Besides, new desirable properties of the user-interfaces emerged with the evolution of the formal model, making a total of seven complex properties formally specified in MCL.

We initiated an integration of our formal model with an industrial control room prototype, provided by a partner in the project. To this aim, several improvements were done in the formal specification, and the integration is currently in progress.

We started to address the introduction of plasticity in the formal specification, a challenge that was identified in 2013. Plasticity is the capacity of a user-interface to withstand variations of its context of use (i.e., platform, user, environment) while preserving usability. We proposed two approaches to introduce plasticity in the analysis. The first one introduces in the formal model a representation of a plasticity engine (responsible for user-interfaces adaptation) and applies model checking to verify its properties. The second approach consists in formally specifying several versions of the user-interfaces, derived from adaptation, and applying equivalence checking to verify similarity relations on the user-interface models.

[0] http://www.energybus.org
[0] http://www.can-cia.org

6.5.8. Fault-Tolerant Routing for Network-on-Chip Architectures

Participant: Wendelin Serwe.

Fault-tolerant architectures provide adaptivity for on-chip communications, but also increase the complexity of the design, so that formal verification techniques are needed to check their correctness. In collaboration with Chris Myers and Zhen Zhang (University of Utah, USA), we studied an extension of the link-fault tolerant Network-on-Chip (NoC) architecture introduced by Wu *et al* [67] that supports multiflit wormhole routing.

To keep the state space manageable, the formal LNT model of the routing algorithm was constructed in several steps, applying different abstractions (structural and related to data). This modeling process led to several insights. First, it led to the discovery of a package leakage path that could lead to the complete loss of a packet and a deadlock. This error in the design of an arbiter was corrected in the subsequent models. Second, a buffering capacity in an arbiter was found to be crucial; this insight also led to a redesign of the arbiters. The resultant changes on the router and arbiter models uncovered interesting symmetries. Finally, we studied how deadlock freedom and tolerance of a single-link fault can be verified for a NoC architecture.

This work led to a publication in an international conference [20].

6.5.8.1. Other Case Studies

The demo examples of CADP, which have been progressively accumulated since the origins of the toolbox, are a showcase for the multiple capabilities of CADP, as well as a test bed to assess the new features of the toolbox. In 2014, the effort to maintain and enhance these demos has been pursued. The progressive migration to LNT has continued, by translating certain demos from LOTOS to LNT. Many demos have been enriched with value-passing temporal formulas that illustrate the conciseness and expressiveness of MCL and the capabilities of the EVALUATOR 4 model checker. Finally, many demos have been shortened and made more readable by using the new features of SVL, especially the "`property`" and "`|=`" statements that allow formulas to be gathered in a single SVL file rather than disseminated in a collection of MCL or XTL files.

HYCOMES Team

6. New Results

6.1. Highlights of the Year

The main advances in 2014 of the Hycomes team have been as follows:

Causality analysis of hybrid systems with ordinary differential equations (ODE) We have proposed a causality analysis, in the form of a simple type system, rejecting hybrid programs with algebraic circuits — see section 6.2 .

An index theory of DAE hybrid systems with differential algebraic equations (DAE) We have proposed a conservative extension of the notion of differentiation index to hybrid systems with differential algebraic equations — see section 6.3 .

6.2. A Type-Based Analysis of Causality Loops In Hybrid Systems Modelers

Explicit hybrid systems modelers like Simulink / Stateflow allow for programming both discrete- and continuous-time behaviors with complex interactions between them. A key issue in their compilation is the static detection of algebraic or causality loops. Such loops can cause simulations to deadlock and prevent the generation of statically scheduled code. In [6] (also published as a deliverable of the Sys2Soft collaborative project [14], see 7.2), we address this issue for a hybrid modeling language that combines synchronous Lustre-like data-flow equations with Ordinary Differential Equations (ODEs). We introduce the operator last(x) for the left-limit of a signal x. This operator is used to break causality loops and permits a uniform treatment of discrete and continuous state variables. The semantics relies on non-standard analysis, defining an execution as a sequence of infinitesimally small steps. A signal is deemed causally correct when it can be computed sequentially and only progresses by infinitesimal steps outside of discrete events. The causality analysis takes the form of a simple type system. In well-typed programs, signals are proved continuous during integration and can be translated into sequential code for integration with off-the-shelf ODE solvers. The effectiveness of this system is illustrated with several examples written in Zélus, a Lustre-like synchronous language extended with hierarchical automata and ODEs.

6.3. On the index of multi-mode DAE Systems

Hybrid systems modelers exhibit a number of difficulties related to the mix of continuous and discrete dynamics and sensitivity to the discretization scheme. Modular modeling, where subsystems models can be simply assembled with no rework, calls for using Differential Algebraic Equations (DAE). In turn, DAE are strictly more difficult than ODE. In most modeling and simulation tools, before simulation can occur, sophisticated pre-processing is applied to DAE systems based on the notion of differentiation index. Graph based algorithms such as the one originally proposed by Pantelides [47] are efficient at finding the differentiation index of a DAE system, structurally (i.e., outside some exceptional values for the system parameters), solving the consistent initialisation problem and, transforming a DAE system into a statically scheduled system of ordinary differential equations (ODE) and implicit functions. The differentiation index for DAE explicitly relies on everything being differentiable. Therefore, extensions to hybrid systems must be done with caution — to our knowledge, no such extension exists, supported by a rigourous mathematical theory. In [8], we use non-standard analysis for this. Non-standard analysis formalizes differential equations as discrete step transition systems with an infinitesimal time basis. This allows to map hybrid DAE systems to difference Algebraic Equations (dAE), for which the notion of difference index can be used. The difference index of a dAE is an easy transposition of the differentiation index of a DAE, where forward shift in time (using a next() operator) replaces differentiation. We prove that the differentiation index of a DAE is structurally equal to the difference index of the dAE resulting from its non-standard interpretation. We can thus propose

the difference index of the non-standard semantics of a hybrid DAE system, as a consistent extension of both the differentiation index of DAE and the difference index of dAE. It turns out that the index theory for (discrete time) dAE systems is interesting in itself and raises new issues. We have investigated graph based method similar to the Pantelides [47] algorithm for computing the difference index of a dAE.

6.4. A Unifying View of Loosely Time-Triggered Architectures

Cyber-Physical Systems require distributed architectures to support safety critical real-time control. Hermann Kopetz' Time-Triggered Architecture (TTA) has been proposed as both an architecture and a comprehensive paradigm for systems architecture, for such systems. TTA offers the programmer a logical discrete time compliant with synchronous programming, together with timing bounds. A clock synchronization protocol is required, unless the local clocks used themselves provide the recquired accuracy. To relax the strict requirements on synchronization imposed by TTA, Loosely Time-Triggered Architectures (LTTA) have been proposed. In LTTA, computation and communication units are all triggered by autonomous, unsynchronized, clocks. Communication media act as shared memories between writers and readers and communication is non blocking. This is at the price of communication artifacts (such as duplication or loss of data), which must be compensated for by using some "LTTA protocol". In [7] we have pursued our previous work by providing a unified presentation of the two variants of LTTA (token- and time-based), with simplified analyses. We compared these two variants regarding performance and robustness and we provide ways to combine them.

MUTANT Project-Team

6. New Results

6.1. Highlights of the Year

Acoustical Society of America Best Paper Award for [20].

International Computer Music Conference (ICMC) Best Presentation Award for [19].

MuTant TEDx Talk in October 2014 on *Human-Computer Musicianship* that attracted more than 12 thousand podcasts according to organisers.

MuTant in CNRS's 2nd edition of "Les Fondamentales" Science and Society event in Grenoble, in a session dedicated to Science and Music on the same Score.

MuTant Participation in the 2014 edition of *Futur en Seine* festival and showcased collaboration with Orchestre de Paris in a public event.

BEST PAPER AWARD :

[19] **International Computer Music Conference**. C. TRAPANI, J. ECHEVESTE.

6.2. Time-Coherency of Bayesian Priors for Sequential Alignment

In the context of Philippe Cuvillier's PhD project, we aim at increasing the robustness of machine listening in situations where observations from the external environment are extremely noisy or incoherent.

Recent results propose a novel insight to the problem of duration modeling for Information Retrieval problems where a discrete sequence of events is estimated from a time-signal using Bayesian models. Since the duration of each event is unknown, a major issue is setting the right Bayesian prior on each of them. Hidden Semi-Markov models (HSMM) allow choosing explicitly any probability distribution for the durations but learning these statistically is a non-parametric problem. In absence of huge training data sets, most algorithms rely on regularization techniques such as choosing parametric classes of distributions but the justifications of such techniques are often heuristics.

Among the numerous application domains of HMM-like paradigms, music-to-audio alignment brings two interesting properties. Firstly, a music score informs of the ordering among events. Secondly, it assigns to each event a nominal duration. For alignment tasks the Markov models conveniently model the ordering with *transient chains*. But the modeling of these nominal durations is a crucial and undermined problematic. This work investigates the relationship of this prior information of duration with the Bayesian priors of a HSMM. Theoretical insights are obtained through the study of the *prior state probability* of transient semi-Markov chains. Whereas ergodic chain and their convergence to an equilibrium probability are well studied, transient chains constitute an undermined case but of prime importance for real-time inference on HSMM.

On the first hand we prove that the non-asymptotical evolution of the state probability has some particular behaviors if the Bayesian priors fulfill several precise conditions, derived from statistical properties like the hazard rate and the tail decay. Then we say that a model is *time-coherent* if the evolution of the state probability respects the information of ordering and nominal lengths. This leads to several prescriptions on the design of HSMM Bayesian priors. On the other hand we get further prescriptions by comparing the Bayesian priors associated to different nominal lengths. This real-valued parameter comes with a natural ordering; we explain why this ordering among parameters is coherently modeled by some specific stochastic orderings among distributions that are standard in statistics.

Intermediate results have been reported in [12], [13]. This worked allowed the development of *Antescofo* version 0.6 released in November 2014.

6.3. Online Methods for Audio Segmentation and Clustering

Audio segmentation is an essential problem in many audio signal processing tasks, which tries to segment an audio signal into homogeneous chunks. Rather than separately finding change points and computing similarities between segments, we focus on joint segmentation and clustering, using the framework of hidden Markov and semi-Markov models. We introduced a new incremental EM algorithm for hidden Markov models (HMMs) and showed that it compares favorably to existing online EM algorithms for HMMs. Early experimental results on musical note segmentation and environmental sound clustering are promising and will be pursued further in 2015.

This project was done in the context of Alberto Bietti's MS project [26] under co-supervision of Arshia Cont (MuTant) and Francis Bach (SIERRA).

6.4. Model-based Testing an Interactive Music System

In the context of the Phd of Clément Poncelet, and in relation with the developments presented in Section 5.3 , we have been studying the application of model-based timed testing techniques to interactive music systems like Antescofo.

Several formal methods have been developed for automatic conformance testing of critical embedded software. The principle is to execute a real implementation under test (IUT) in a testing framework, black-box, by sending it carefully selected inputs and then observing and analyzing its outputs. In conformance model-based testing (MBT), the input and corresponding expected outputs are generated according to formal models of the IUT and the environment. The models of timed automata with inputs and outputs, and tools like the the Uppaal suite have been developed for extending such techniques to realtime systems [32], [31]. Several procedures have been designed for addressing the task described in Section 5.3 .

The case of IMS presents important originalities compared to other MBT applications to realtime systems. On the one hand, the time model supports several time units, including the wall clock time, measured in seconds, and the time of music scores, measured in number of beats relatively to a tempo. This situation raised several new problems for the generation of test suites and their execution. On the other hand, the formal specification of the IUT's behavior on a given score is produced automatically by a *score compiler*, using an intermediate representation. We rely on the realistic hypotheses that a mixed score specify completely the expected timed behavior of the IMS. Hence, our test method is fully automatic, in contrast with other approaches which generally require experts to write the specification manually. This workflow fits well in a music authoring workflow where scores in preparation are constantly evolving. We have been applying our tools to small benchmark made of characteristic scores, as well as to real mixed scores used in concerts, and some bugs in Antescofo have been identified. These results have been presented in the conference ICMC 2014 [18] and will be presented during the 30th ACM/SIGAPP Symposium On Applied Computing, track Software Verification and Testing [17].

6.5. Antescofo Temporal Pattern

An important enhancement has been made by the introduction of an expressive temporal pattern language [15] in *Antescofo*. Temporal patterns are used to define complex events that correspond to a combination of perceived events in the musical environment as well as arbitrary logical and metrical temporal conditions. The real time recognition of such event is used to trigger arbitrary actions in the style of event-condition-action rules.

The semantics of temporal pattern matching is defined to parallel the well-known notion of regular expression and Brzozowski's derivatives but extended to handle an infinite alphabet, arbitrary predicates, elapsing time and inhibitory conditions.

Temporal patterns are implemented by translation into a core subset of the Antescofo domain specific language. This compilation has proven efficient enough to avoid the extension of the real-time runtime of the language and has been validated with composers in actual pieces.

6.6. OpenMusic reactive Model

In collaboration with Jean Bresson, we have extended the evaluation model of OpenMusic to integrate reactive capabilities [10]. OpenMusic (OM) is a domain-specific visual programming language designed for computer-aided music composition based on Common Lisp. It allows composers to develop functional processes generating or transforming musical data. To extend OM towards reactive applications, we have proposed to integrate its demand-driven evaluation mechanism with reactive data-driven evaluations in a same and consistent visual programming framework. To this end, we have developped the first denotational semantics of the visual language, which gives account for its demand-driven evaluation mechanism and the incremental construction of programs. We then have extended this semantics to enable reactive computations in the functional graphs. The resulting language merges data-driven executions with the existing demand-driven mechanism. This integration allows for the propagation of changes in the programs, and the evaluation of graphically-designed functional expressions as a response to external events, a first step in bridging the gap between computer-assisted composition environments and real-time musical systems.

6.7. Representation of Rhythm and Quantization

Rhythmic data are commonly represented by tree structures (rhythms trees) in assisted music composition environments, such as OpenMusic, due to the theoretical proximity of such structures with traditional musical notation. We are studying the application in this context of techniques and tools for processing tree structure, which were originally developed for other areas such as natural language processing, automatic deduction, Web data ... We are particularly interested in two well established formalisms with solid theoretical foundations: term rewriting and tree automata.

The problem of rhythm transcription, or quantization, is to generate, from a timed sequence of notes (e.g. a file in MIDI format), a score in traditional music notation. The input events can come from an interpretation on a MIDI keyboard or be the result of a computation in OpenMusic. This problem arises immediately as insoluble unequivocally: we shall calibrate the system to fit the musical context, balancing constraints of precision, or of simplicity / readability of the generated scores. For this purpose, we are developing in collaboration with Slawek Staworko (LINKS, currently on leave at University of Edinburgh) for algorithms searching optimums in large sets of weighted trees (tree series), representing possible solutions to a problem quantification. A prototype has been developed and is under evaluation on real case studies. For the construction of appropriate tree series, we turn to semi-supervised systems, where the composer's interactions are predominant in the smooth process. These work have been presented in an invited talk in the workshop of the IFIP working group on term rewriting.

With Prof. Masahiko Sakai (Nagoya University, a specialist in term rewriting), we conduct a complementary work [14] on the representation of rhythmic notation. The goal is to define a structural theory as equations on trees rhythms. This approach can be used for example to generate, by transformation, different notations possible the same rate, with the ability to select in accordance with certain constraints.

PARKAS Project-Team

6. New Results

6.1. Highlights of the Year

The paper *ReactiveML, a reactive extension to ML* of Mandel and Pouzet has been declared to be the *most influential paper of PPDP (Principles and Practice of Declaractive Programming) 2005*. A previous version of the paper, submitted to JFLA'05, has been declared to be "une contribution marquante parmi les articles publiés aux JFLA".

6.2. Quasi-synchrony

Participants: Guillaume Baudart, Timothy Bourke, Marc Pouzet.

We study the implementation of critical control applications on the so-called *quasi-periodic* distributed architectures. These architectures, used in civil avionics (e.g., Airbus A380), consist of a collection of distributed processors running with *quasi-periodic* clocks, that is, un-synchronized physical clocks subject to bounded jitterring. The theory of quasi-synchrony has been introduced by Paul Caspi in the 2000' [29]. Loosely Time-Triggered Architectures (LTTA) denotes such architectures with the prototocol used to implement a synchronous program on top of it.

Over the last ten year two protocols were considered: (1) *Back-Pressure* LTTA [25] based on a acknowledgement mechanism reminiscent of elastic circuit [43]. (2) *Time-Based* LTTA [28] which uses timing constraints of the architecture to mimic a synchronous execution.

During year 2014, we have entirelly reformulated the model of LTTA using synchronous semantics and principles. Compared to previous formalizations based on Petri nets [24], this new presentation is is simpler and more uniform with the same theoretical model used for both the application and the protocol ((1) or (2)). Moreover, it is easier to consider mixed protocols (a whole application with part based on time-based communication and others based on back-pressure). Besides this, we also proposed a new and more flexible Time-Based LTTA, allowing for pipelining by not reconstructing global synchronization, unlike what was done in previous Time-Based LTTA.

6.3. Hybrid Synchronous Languages

Participants: Guillaume Baudart, Timothy Bourke, Marc Pouzet.

During year 2014, we mainly worked on two directions: (a) the design and implementation of causality analysis for hybrid systems modelers; (b) the design and implementation of a new compilation technique producing imperative sequential code.

This research is conducted in collaboration with Albert Benveniste and Benoit Caillaud (Hycomes team at Inria, Rennes), Jean-Louis Colaco, Cédric Pasteur and Bruno Pagano from the SCADE core team of Esterel-Technologies/ANSYS.

Causality analysis In this work, we address the static detection of causality loops for a hybrid modeling language that combines synchronous Lustre-like data-flow equations with Ordinary Differential Equations (ODEs). We introduce the operator *last(x)* for the left-limit of a signal x. This operator is used to break causality loops and permits a uniform treatment of discrete and continuous state variables. The semantics relies on non-standard analysis, defining an execution as a sequence of infinitesimally small steps. A signal is deemed *causally correct* when it can be computed sequentially and only progresses by infinitesimal steps outside of discrete events. The causality analysis takes the form of a simple type system. In well-typed programs, signals are proved continuous during integration.

This analysis has been presented at [4] and is fully implemented in the hybrid synchronous language Zélus.

A Synchronous-based Code Generator For Explicit Hybrid Systems Languages The generation of sequential code is important for simulations to be efficient and to produce target embedded code. While sequential code generation in hybrid modeling tools is routinely used for efficient simulation, it is little or not used for producing target embedded code in critical applications submitted to strong safety requirements. This is a break in the development chain: parts of the applications must be rewritten into either sequential or synchronous programs, and all properties verified on the source model cannot be trusted and have to be re-verified on the target code.

In this work, we present a novel approach for the code generation of a hybrid systems modeling language. By building on top of an existing synchronous language and compiler, it reuses almost all the existing infrastructure with only a few modifications. Starting from an existing synchronous data-flow language extended with Ordinary Differential Equations (ODEs), we detail the translation to sequential code. The translation is expressed as a sequence of source-to-source transformations. A generic intermediate language is introduced to represent transition functions which are turned into C code. The versatility of the compiler organisation is illustrated by considering two classical targets: generation of simulation code complying with the FMI standard and linking with an off-the-shelf numerical solver (Sundials CVODE).

This new code generation has been implemented in two different compilers: the Zélus research prototype and the industrial SCADE Suite KCG code generator, at Esterel-Technologies/ANSYS. Here, SCADE is conservatively extended with ODEs, following previous works by Benveniste et al. and implemented in Zélus. In the SCADE compiler, it was possible to reuse almost all the existing infrastructure like static checking, intermediate languages, and optimisations, with few modifications. The extension to account for hybrid features represents only 5% additional lines of code, which is surprisingly low. Moreover, the proposed language extension is conservative in that regular synchronous functions are compiled as before—the same synchronous code is used both for simulation and for execution on target platforms.

This full-scale validation confirm the interest in building a hybrid systems modeler on top of a existing synchronous language. Moreover, the precise definition of code generation, built on a proven compiler infrastructure of a synchronous language avoids the rewriting of control software and may also increase the confidence in what is simulated.

This work will be presented at the *International Conference on Compiler Construction (CC)*, in April 2015.

6.4. Fidelity in Real-Time Programming

Participants: Guillaume Baudart, Timothy Bourke.

Synchronous languages are a rigorous approach to programming, analyzing, and implementing embedded systems. Real-time aspects are typically handled by discretizing time using either (implicit) ticks or (explicit) named signals, and later verifying that the (necessarily bounded) execution time of a reaction is strictly less than the period of the fastest timing signal. This approach has many advantages: it separates logical behaviour from implementation concerns, yields a simple and precise programming model, and abstracts from eventual run-time environments. For an important subclass of embedded protocols and controllers, however, we believe it advantageous to add constructions that deal more concretely with real-time constraints.

We are pursuing these ideas in the enriched timing model provided by the Zélus programming language (detailed elsewhere). We continue to study the extension and application of this language to the modelling, simulation, analysis, and implementation of real-time embedded software.

This year we developed three case studies: quasi-synchronous architectures (from last year), loosely time-triggered architectures (detailed elsewhere), and a small embedded controller. These case studies motivate and drive our research and implicitly define the subclass of embedded systems that we aim to treat. They have each been modelled in Zélus and can be simulated with the existing compiler.

We made progress on defining a subset of Zélus that is ammenable to discretization techniques for more flexible simulation. A first version of an appropriate algorithm has been sketched and partially implemented. Work continues on developing it with the idea of incorporating it into the Zélus compiler and using it to treat our case studies.

6.5. Mechanization of AODV loop freedom proof

Participant: Timothy Bourke.

The Ad hoc On demand Distance Vector (AODV) routing protocol is described in RFC3561. It allows the nodes in a Mobile Ad hoc Network (MANET) to know where to forward messages so that they eventually reach their destinations. The nodes of such networks are *reactive systems* that cooperate to provide a global service (the sending of messages from node to node) satisfying certain correctness properties (namely 'loop freedom'—that messages are never sent in circles).

This year I finalized both the framework for network invariant proofs [20] and its application to the AODV protocol [21] and submitted them for inclusion in the *Archive of Formal Proof*, an online and open-source repository of formal developements in the Isabelle proof assistant (indexed as a journal). I presented results on the framework at the Vienna 'Summer of Logic' [6] and my colleagues presented the application in Sydney [5]. Together with an intern at NICTA and Sydney, my colleagues and I made preliminary investigations into extending the framework and model with timing details. A journal version of the ITP paper has been submitted.

In collaboration with Peter Höfner (NICTA) and Robert J. van Glabbeek (UNSW/NICTA).

6.6. Reasoning about C11 Program Transformations

Participants: Francesco Zappa Nardelli, Thibaut Balabonski, Robin Morisset.

We have shown that the weak memory model introduced by the 2011 C and C++ standards does not permit many of common source-to-source program transformations (such as expression linearisation and "roach motel" reordering) that modern compilers perform and that are deemed to be correct. As such it cannot be used to define the semantics of intermediate languages of compilers, as, for instance, LLVM aimed to. We consider a number of possible local fixes, some strengthening and some weakening the model. We have evaluated the proposed fixes by determining which program transformations are valid with respect to each of the patched models. We have provided formal Coq proofs of their correctness or counterexamples as appropriate.

A paper on this work has been accepted in [13]. In collaboration with Viktor Vafeiadis (MPI-SWS, Germany).

6.7. Language design on top of JavaScript

Participant: Francesco Zappa Nardelli.

This research project aims at improving the design of the JavaScript language. In [22] we propose a typed extension of JavaScript combining dynamic types, concrete types and like types to let developers pick the level of guarantee that is appropriate for their code. We have implemented our type system and we have explored the performance and software engineering benefits.

With Gregor Richards and Jan Vitek (Purdue University).

6.8. Tiling for Stencils

Participants: Tobias Grosser, Sven Verdoolaege, Albert Cohen.

This research project aims with optimizing time-iterated stencil operations.

Iterative stencil computations are important in scientific computing and more and more also in the embedded and mobile domain. Recent publications have shown that tiling schemes that ensure concurrent start provide efficient ways to execute these kernels. Diamond tiling and hybrid-hexagonal tiling are two tiling schemes that enable concurrent start. Both have different advantages: diamond tiling has been integrated in a general purpose optimization framework and uses a cost function to choose among tiling hyperplanes, whereas the greater flexibility with tile sizes for hybrid-hexagonal tiling has been exploited for effective generation of GPU code.

We undertook a comparative study of these two tiling approaches and proposed a hybrid approach that combines them. We analyzed the effects of tile size and wavefront choices on tile-level parallelism, and formulate constraints for optimal diamond tile shapes. We then extended, for the case of two dimensions, the diamond tiling formulation into a hexagonal tiling one, which offers both the flexibility of hexagonal tiling and the generality of the original diamond tiling implementation. We also showed how to compute tile sizes that maximize the compute-to-communication ratio, and apply this result to compare the best achievable ratio and the associated synchronization overhead for diamond and hexagonal tiling.

One particularly exciting result is the ability to apply tiling to periodic data domains. These computations are prevalent in computational sciences, particularly in partial differential equation solvers. We proposed a fully automatic technique suitable for implementation in a compiler or in a domain-specific code generator for such computations. Dependence patterns on periodic data domains prevent existing algorithms from finding tiling opportunities. Our approach augments a state-of-the-art parallelization and locality-enhancing algorithm from the polyhedral framework to allow time-tiling of stencil computations on periodic domains. Experimental results on the swim SPEC CPU2000fp benchmark show a speedup of $5\times$ and $4.2\times$ over the highest SPEC performance achieved by native compilers on Intel Xeon and AMD Opteron multicore SMP systems, respectively. On other representative stencil computations, our scheme provides performance similar to that achieved with no periodicity, and a very high speedup is obtained over the native compiler. We also report a mean speedup of about $1.5\times$ over a domain-specific stencil compiler supporting limited cases of periodic boundary conditions. To the best of our knowledge, it has been infeasible to manually reproduce such optimizations on swim or any other periodic stencil, especially on a data grid of two-dimensions or higher.

These works resulted in a number of high-profile publications, including a nommination for a best paper award, and culminated with the PhD thesis defense of Tobias Grosser.

6.9. Portable representation for polyhedral compilation

Participants: Riyadh Baghdadi, Michael Kruse, Chandan Reddy, Tobias Grosser, Sven Verdoolaege, Albert Cohen.

Programming accelerators such as GPUs with low-level APIs and languages such as OpenCL and CUDA is difficult, error prone, and not performance-portable. Automatic parallelization and domain specific languages have been proposed to hide this complexity and to regain some performance portability. We proposed PENCIL, a subset of GNU C99 with specific programming rules. A compiler for a Domain-Specific Language (DSL) may use it as a target language, a domain expert may use it as a portable implementation language facilitating the parallelization of real-world applications, and an optimization expert may use PENCIL to accelerate legacy applications.

The design of PENCIL is simultaneously a key research result and a milestone for parallelizing compiler engineering/design. Aspects of its static-analysis-friendly, formal semantics are highly original, for the language's ability to preserve expressiveness and modularity without jeopardizing a (polyhedral) compiler's ability to perform aggressive transformations. We validated its potential as a front-end to a state-of-the-art polyhedral compiler, extending its applicability to dynamic, data dependent control flow and non-affine array accesses. We illustrated this PENCIL-enabled flow on the generation of highly optimized OpenCL code, considering a set of standard benchmarks (Rodinia and SHOC), image processing kernels, and DSL embedding scenarios for linear algebra (BLAS) and signal processing radar applications (SPEAR-DE). We ran experimental results on a variety of platforms, including an AMD Radeon HD 5670 GPU, an Nvidia GTX470 GPU, and an ARM Mali-T604 GPU.

This work is conducted in collaboration with partners from ARM, RealEyes (a computer vision company) and Imperial College.

6.10. Correct and efficient runtime systems

Participants: Nhat Minh Lê, Robin Morisset, Adrien Guatto, Albert Cohen.

Complementing our different compilation efforts for synchronous and task-parallel data-flow languages, we studied the implementation of Kahn process networks, a deterministic parallel programming model, on shared memory multiprocessors. This model is based on a familiar abstraction: blocking communication through bounded, in-order, single-producer single-consumer queues.

We proposed two novel algorithms that construct such blocking queues on top of concurrent ring buffers and user-land scheduling components. We implemented our algorithms in C11, taking advantage of the relaxed memory model of the language, and prove the correctness of this implementation.

We used these algorithms in a complete runtime system for Kahn process networks with applications ranging from linear algebra kernels to stream computing. In particular, our implementations of the Cholesky and LU factorizations outperform state-of-the-art parallel linear algebra libraries on commodity x86 hardware.

6.11. A Functional Synchronous Language with Integer Clocks

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

Synchronous languages in the vein of Lustre are first-order functional languages dedicated to stream processing. Lustre compilers use a type-like static analysis, the clock calculus, to reject programs that cannot be implemented as finite state machines. The broad idea is to assign to each element of a stream a logical computation date in a global, discrete time scale. When this analysis succeeds, the types obtained guide the code generation phase of the compiler, which produces transition functions. In practice, these functions consists in simple, bounded memory C code featuring only assignments and conditional statements.

This research work explores a variation on Lustre and its compilation. Our proposal is twofold. First, we add a new construct that creates a local time scale whose internal steps are invisible from the outside. Second, we change the clock calculus to allow several elements of a stream to be computed during the same time step. The resulting type system comes with a soundness proof, which relies on an elementary form of step-indexed realizability, and with a code generation scheme adapted to the new setting, and featuring nested loops in the target code.

SPADES Team

6. New Results

6.1. Components and Contracts

Participant: Jean-Bernard Stefani.

6.1.1. Location graph model

The design of configurable systems can be streamlined and made more systematic by adopting a component-based structure, as demonstrated with the Fractal component model [2]. However, the formal foundations for configurable component-based systems, featuring higher-order capabilities where components can be dynamically instantiated and passivated, and non-hierarchical structures where components can be contained in different composites at the same time, are still an open topic. We have developed recently the location graph model [15], where components are understood as graphs of locations hosting higher-order processes, and where component structures can be arbitrary graphs. We have developed a compositional operational semantics for the location graph model, which is parametric with respect to the family of processes. We have shown that the location graph model constitutes a conservative extension of a previous model, called CAB, that captures the key features of the BIP component model [5]. We have further worked on the behavioral theory of the location graph model, characterizing contextual equivalence in the model by means of a higher-order bisimularity relation, and begun the study of the encoding of different models, including the Synchronized Hyperedge Replacement model [45].

6.2. Real-Time multicore programming

Participants: Vagelis Bebelis, Adnan Bouakaz, Pascal Fradet, Alain Girault, Gregor Goessler, Jean-Bernard Stefani, Sophie Quinton, Partha Roop, Eugene Yip.

6.2.1. 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 or 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.

Recently, we have introduced the *schedulable parametric data-flow* (SPDF) MoC for dynamic streaming applications [47]. SPDF extends the standard dataflow model by allowing rates to be parametric. Last year, we have proposed the *Boolean Parametric Data Flow* (BPDF) MoC which combines integer parameters (to express dynamic rates) and boolean parameters (to express the activation and deactivation of communication channels). High dynamicity is provided by integer parameters which can change at each basic iteration and boolean parameters which can change even within the iteration. We have presented static analyses which ensure the liveness and the boundedness of BDPF graphs.

Recently, we have proposed a generic and flexible framework to generate parallel schedules for BPDF applications [16]. The parametric dataflow graph is associated with 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. One of the target architectures is the STHORM many-core platform designed by STMicroelectronics.

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

6.2.2. Typical Worst-Case Analysis of real-time systems

Weakly hard time constraints have been proposed for applications where occasional deadline misses are permitted. We have recently developed Typical Worst Case Analysis (TWCA) to exploit similar constraints and bound response times of systems with sporadic overload. This year, we have applied this approach to a real-life automotive network [14]. Additionally, we have extended the approach for static priority preemptive (SPP) and static priority non-preemptive (SPNP) scheduling to determine the maximum number of deadline misses of a given task [21]. The approach is based on an optimization problem which trades off higher priority interference versus miss count. We formally derived a lattice structure for these combinations that lays the ground for an integer linear programming (ILP) formulation. The ILP solution was evaluated and provided far better results than previous TWCA.

In parallel, we have contributed to a systematic co-engineering approach that integrates TWCA into functional analysis [19]. We combine physical, control and timing models by representing them as a network of hybrid automata. Closed-loop properties can then be verified on this hybrid automata network by using standard model checkers for hybrid systems. The use of the Logical Execution Time (LET) semantics where data is written back deterministically at the typical worst-case response time (rather than the usual worst-case bound) is a new and particularly powerful approach for addressing the computational complexity of the model checking problem.

6.2.3. Time predictable programming

In the context of the RIPPES associated team with UC Berkeley and U Auckland, we have finalized ongoing work on our synchronous programming language for time predictability PRET-C [10]. PRET-C extends C with synchronous constructs inspired by ESTEREL, to allow an easy programming of concurrent reactive programs. These constructs allow the programmer to express concurrency, interaction with the environment, looping, and a synchronization barrier (like the `pause` statement in ESTEREL). PRET-C's semantics is deterministic, and it can be efficiently compiled towards sequential code, either executed on a dedicated processor for the best predictability of the program's Worst-Case Reaction Time (WCRT), or executed on a generic processor.

We have also continued our work on FOREC, a time predictable synchronous programming language for multi-core chips. Like PRET-C, it extends C with a small set of ESTEREL-like synchronous primitives. FOREC threads communicate with each other via shared variables, the values of which are combined at the end of each tick to maintain deterministic execution. FOREC is compiled into threads that are then statically scheduled for a target multi-core chip. This is the main difference with PRET-C. We have finalized the semantics of FOREC, which led us to propose several ways to combine shared variables at the tick boundaries, such that the semantics remains deterministic. This part was inspired by the so-called concurrent revisions [38].

Finally, with colleagues from the former ARTISTDESIGN European Network of Excellence, we have also participated in a survey on predictable embedded systems [11].

6.2.4. Tradeoff exploration between energy consumption and execution time

We have continued our work on multi-criteria scheduling, in the particular context of dynamic applications that are launched and terminated on an embedded multi-core chip, under execution time and energy consumption constraints. We have proposed a two layer adaptive scheduling method. In the first layer, each application (represented as a DAG of tasks) is scheduled statically on sets of cores: 2 cores, 3 cores, 4 cores, and so on. For each size of these sets (2, 3, 4, ...), there may be only one topology or several topologies. For instance, for 2 or 3 cores there is only one topology (a "line"), while for 4 cores there are three distinct topologies ("line", "square", and "T shape"). Moreover, for each topology, we generate statically several schedules, each one subject to a different total energy consumption constraint, and consequently with a different Worst-Case Reaction Time (WCRT). Coping with the energy consumption constraints is achieved thanks to Dynamic Frequency and Voltage Scaling (DVFS). In the second layer, we use these pre-generated static schedules to reconfigure dynamically the applications running on the multi-core each time a new application is launched or

an existing one is stopped. The goal of the second layer is to perform a global optimization of the configuration, such that each running application meets a pre-defined quality-of-service constraint (translated into an upper bound on its WCRT) and such that the total energy consumption is minimized. For this, we (1) allocate a sufficient number of cores to each active application, (2) allocate the unassigned cores to the applications yielding the largest gain in energy, and (3) choose for each application the best topology for its subset of cores (i.e., better than the by default "line" topology).

This is a joint work with Ismail Assayad (U. Casablanca, Morocco) who visits the team regularly.

6.3. Language Based Fault-Tolerance

Participants: Dmitry Burlyaev, Pascal Fradet, Alain Girault, Yoann Geoffroy, Gregor Goessler, Jean-Bernard Stefani.

6.3.1. Automatic transformations for fault tolerant circuits

In the past 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 automatic transformations to ensure fault-tolerance properties in digital circuits. To this aim, we consider program transformations for hardware description languages (HDL). We consider both single-event upsets (SEU) and single-event transients (SET) and fault models of the form "at most 1 SEU or SET within n clock signals".

We have expressed several variants of triple modular redundancy (TMR) as program transformations. We have proposed a verification-based approach to minimize the number of voters in TMR [17]. Our technique guarantees that the resulting circuit *(i)* is fault tolerant to the soft-errors defined by the fault model and *(ii)* is functionally equivalent to the initial one. Our approach operates at the logic level and takes into account the input and output interface specifications of the circuit. Its implementation makes use of graph traversal algorithms, fixed-point iterations, and BDDs. Experimental results on the ITC'99 benchmark suite indicate that our method significantly decreases the number of inserted voters, which entails a hardware reduction of up to 55% and a clock frequency increase of up to 35% compared to full TMR. We address scalability issues arising from formal verification with approximations and assess their efficiency and precision.

We have proposed novel fault-tolerance transformations based on time-redundancy. In particular, we have presented a transformation using double-time redundancy (DTR) coupled with micro-checkpointing, rollback and a speedup mode [18]. The approach is capable to mask any SET every 10 cycles and keeps the same input/output behavior regardless error occurrences. Experimental results on the ITC'99 benchmark suite indicate that the hardware overhead is 2.7 to 6.1 times smaller than full TMR with double loss in throughput. It is an interesting alternative to TMR for logic intensive designs.

We have also designed a transformation that allows the circuit to change its level of time-redundancy. This feature permits to dynamically and temporarily give up (resp. increase) fault-tolerance and speed up (resp. slow down) the circuit. The motivations for such changes can be based on the observed change in radiation environment or the processing of (non)critical data. These different time redundancy transformations have been patented [23]

We have started the formal certification of such transformations using the Coq proof assistant [40]. The transformations are described on a simple gate-level hardware description language inspired from μFP [68]. The fault-model is described in the operational semantics of the language. The main theorem states that, for any circuit, for any input stream and for any SET allowed by the fault-model, its transformed version produces a correct output. A TMR and triple time redundancy transformations have already been proved correct. The proof of the DTR transformation is in progress.

6.3.2. Concurrent flexible reversibility

In the recent years, we have been investigating reversible concurrent computation, and investigated various reversible concurrent programming models, with the hope that reversibility can shed some light on the common semantic features underlying various forms of fault recovery techniques (including, exceptions, transactions, and checkpoint/rollback schemes).

We have revisited our encoding of our reversible higher-order π-calculus in (a variant of) the higher-order π-calculus, in order to obtain a much tighter result than our original encoding. In essence, we now have a form of strong bisimilarity (modulo administrative reductions) between a reversible higher-order π-calculus process and its translation in higher-order π. We have also studied the relation between the causality information used in our reversible higher-order π and a causal higher-order π-calculus, inspired by the causal π-calculus [35]. This work has been submitted for publication [24]. This work was done in collaboration with Inria teams FOCUS in Bologna, as part of the ANR REVER project.

6.3.3. Blaming in component-based systems

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 [51] and its instantiations. However, this approach relies on a causal model that may not be known, for instance in presence of black-box components. For such systems, we have been developing a framework for blaming that helps us establish the causal relationship between component failures and system failures, given an observed system execution trace. The analysis is based on a formalization of counterfactual reasoning. We have shown in [12] how our approach can be used for log analysis to help establishing liability in the context of legal contracts.

We have proposed in [6] an approach for blaming in component-based real-time systems whose component specifications are given as timed automata. The analysis is based on a single execution trace violating a safety property P. We have formalized blaming using counterfactual reasoning to distinguish component failures that actually contributed to the outcome from failures that had no impact on the violation of P. We have shown how to effectively implement blaming by reducing it to a model-checking problem for timed automata. The approach has been implemented in LOCA (Section 5.1.1). We have further demonstrated the feasibility of our approach on the model of a dual-chamber implantable pacemaker.

6.3.4. Synthesis and implementation of fault-tolerant embedded systems

We have integrated a complete workflow to synthesize and implement correct-by-construction fault tolerant distributed embedded systems consisting of real-time periodic tasks. Correct-by-construction is provided by the use of discrete controller synthesis [63] (DCS), a formal method thanks to which we are able to guarantee that the synthe-sized controlled system satisfies the functionality of its tasks even in the presence of processor failures. For this step, our workflow uses the Heptagon domain specific language [43] and the Sigali DCS tool [59]. The correct implementation of the resulting distributed system is a challenge, all the more since the controller itself must be tolerant to the processor failures. We achieve this step thanks to the libDGALS real-time library [22] (1) to generate the glue code that will migrate the tasks upon processor failures, maintaining their internal state through migration, and (2) to make the synthesized controller itself fault-tolerant. We have demonstrated the feasibility of our work-flow on a multi-tasks multi-processor fault-tolerant distributed system.

TEA Project-Team

6. New Results

6.1. Highlights of the Year

This year's effort has been mainly devoted to the successful creation of project-team TEA and the definition of its new research perspective on Time, Events and Architectures in CPS design.

The SAE committee on the AADL adopted our recommendations to implement a timed and synchronous behavioural annex [13], [11] for standardisation [20]. The specification and reference implementation of this revised behavioral annex will be the focus of most our attention next year.

Adnan Bouakaz published and implemented more of the original results from his PhD. work on abstract affine scheduling [14], [15].

6.2. Priority-Driven Scheduling of Static Dataflow Graphs through Time Abstraction

Participants: Adnan Bouakaz, Thierry Gautier, Jean-Pierre Talpin.

Static dataflow graph models, such as SDF [0] and CSDF [0], are widely used to design concurrent real-time streaming applications due to their inherent functional determinism and predictable performances. The state of the art usually advocates static-periodic scheduling of dataflow graphs over dynamic scheduling. Through the past decades, a considerable effort has been made to solve this problem [0]. Ensuring boundedness and liveness is the essence of the proposed algorithms in addition to optimizing some nonfunctional performance metrics (e.g. buffer minimization, throughput maximization, etc.).

Nowadays real-time streaming applications on MPSoCs are increasingly complex; and runtime systems are more needed to handle resource sharing, task priorities, etc. Therefore, recent works [000] are considering dynamic scheduling policies (e.g. earliest-deadline first scheduling, deadline monotonic scheduling, etc.) for dataflow graphs. The main motivations of these works are: (1) most existing real-time operating systems support such scheduling policies; (2) applicability of the existing schedulability theory [00]; and (3) with such dynamic approach, multiple and independent applications, each designed as a dataflow graph, can run concurrently on the same platform.

Our work [00] [14], [15] proposes a sequence-based framework in which a large class of priority-driven schedules can be uniformly expressed and analyzed. Infinite sequences are used to describe the dataflow graphs (e.g. rate sequences, execution time sequences) and both concrete and abstract schedules (e.g. activation clocks, priority sequences, activation relations, etc.). The framework can be then easily adapted for specific needs (e.g.

[0] *Synchronous data-flow*. E. A. Lee and D. G. Messerschmitt. Proceedings of the IEEE, 1987.
[0] *Cycle-static data-flow*. Blisen, G. and Engels, M. and Lauwereins, R. and Peperstraete, Transactions on Signal Processing, v.2. 1996.
[0] *Software synthesis from dataflow graphs*. Battacharyya, S. and Lee, E. and Murthy, P. Kluwer Academic Publishers, 1996.
[0] *Affine Data-Flow Graphs for the Synthesis of Hard Real-Time Applications*. International Conference on Application of Concurrency to System Design. IEEE Press, 2012
[0] *Temporal analysis flow based on an enabling rate characterization for multi-rate applications executed on MPSoCs with non-starvation-free schedulers*. Hausmans, J., et al. International Workshop on Software and Compilers for Embedded Systems, 2014.
[0] *Hard-real-time scheduling of data-dependent tasks in embedded streaming applications.* Bamakhrama, M. and Stefanov, T. Embedded Systems Conference. ACM, 2011
[0] *Real time scheduling theory: a historical perspective*. Sha, L. et al. Real-Time Systems Conference. IEEE, 2004
[0] *A survey of hard real-time scheduling for multiprocessor systems.* Davis, R. and Burns, A. ACM Computing Surveys, v. 4, 2011
[0] *Buffer Minimization in Earliest-First Scheduling of Dataflow Graphs*. A. Bouakaz, J-P. Talpin. ACM conference on languages, compilers and tools for embedded systems. ACM Press, 2013.
[0] *Design of Safety-Critical Java Level 1 Applications Using Affine Abstract Clocks*. A. Bouakaz, J-P. Talpin. International Workshop on Software and Compilers for Embedded Systems, 2013.

affine scheduling). Our schedule construction approach is based on two steps. The first step consists in computing an abstract schedule which consists of a set of priority sequences, processor allocation sequences, and activation relations. An activation relation between two actors describes the relative order of their activations, and hence allows us to compute safe sizes of channels between them using worst-case overflow/underflow scenarios. This step must satisfy some correctness constraints such as consistency and exclusion of overflow and underflow exceptions. Once the best abstract schedule (w.r.t. to a performance metric) is computed, the schedule is refined by computing the actual periods and phases that ensure schedulability on the target architecture.

6.3. Formal Verification of a Synchronous Data-flow Compiler: from Signal to C

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

Translation validation [00] is a technique that attempts to verify that program transformations preserve the program semantics. It is obvious to prove globally that the source program and its final compiled program have the same semantics. However, we believe that a better approach is to separate concerns and prove each analysis and transformation stage separately with respect to ad-hoc data-structures to carry the semantic information relevant to that phase.

In the case of the Signal compiler [1], [7][12], the preservation of the semantics can be decomposed into the preservation of clock semantics at the *clock calculation* phase and that of data dependencies at the *static scheduling* phase, and, finally, value-equivalence of variables at the *code generation* phase.

Translation Validation for Clock Transformations in a Synchronous Compiler. In this work, the clock semantics of the source and transformed programs are formally represented as *clock models*. A clock model is a first-order logic formula that characterizes the presence/absence status of all signals in a Signal program at a given instant. Given two clock models, a *clock refinement* between them is defined which expresses the semantic preservation of clock semantics. A method to check the existence of clock refinement is defined as a satisfiability problem which can be automatically and efficiently proved by a SMT solver.

Let Cp^{sig} and Val_{clk} be the functions which define the Signal compiler and a validator, respectively. The following function defines a formally verified compiler for the *clock calculation and Boolean abstraction* phase. We write $C \sqsubseteq_{clk} A$ to denote that there exists a refinement between A and C.

$$Cp^{sig}_{Val_{clk}}(A) = \begin{cases} C & \text{if } Cp^{sig}(A) = C \text{ and } Val_{clk}(A, C) = true \\ Error & \text{if } Cp^{sig}(A) = C \text{ and } Val_{clk}(A, C) = false \\ Error & \text{if } Cp^{sig}(A) = Error \end{cases}$$

where $Val_{clk}(A, C) =$ true if and only if $C \sqsubseteq_{clk} A$.

Precise Deadlock Detection for Polychronous Data-flow Specifications. Dependency graphs are a commonly used data structure to encode the streams of values in data-flow programs and play a central role in scheduling instructions during auto-mated code generation from such specifications. In this work [17], we propose a precise and effective method that combines a structure of dependency graph and first order logic formulas to check whether multi-clocked data-flow specifications are deadlock free before generating code from them. We represent the flow of values in the source programs by means of a dependency graph and attach first-order logic formulas to condition these dependencies. We use an SMT solver [0] to effectively reason about the implied formulas and check deadlock freedom.

[0] *Translation validation.* Pnueli A., Siegel M., and Singerman E. In Proceedings of TACAS'98, 1998.
[0] *Translation validation: From signal to c.* M. Siegel A. Pnueli and E. Singeman. In Correct Sytem Design Recent Insights and Advances, 2000.
[0] *Satisfiability modulo theories: An appetizer.* L. de Moura and N. Bjorner. In Brazilian Symposium on Formal Methods, 2009.

Evaluating SDVG translation validation: from Signal to C. This work focuses on proving that every output signal in the source program and the corresponding variable in the compiled program, the generated C program, have the same values. The computations of all signals and their compiled counterparts are represented by a shared value-graph, called *Synchronous Data-flow Value-Graph* (SDVG).

Given a SDVG, assume that we want to show that two variables have the same value. We simply need to check that they are represented by the same sub-graph, meaning that they point to the same graph node. If all output signals in the source program A and the corresponding variables in the generated C program have the same value, then we say that C refines A, denoted by $C \sqsubseteq_{val} A$.

Implementation and Experiments. At a high level, our tool *SigCert* (https://scm.gforge.inria.fr/svn/sigcert) developed in OCaml checks the correctness of the compilation of Signal compiler w.r.t clock semantics, data dependence, and value-equivalence as given in Figure 3 .

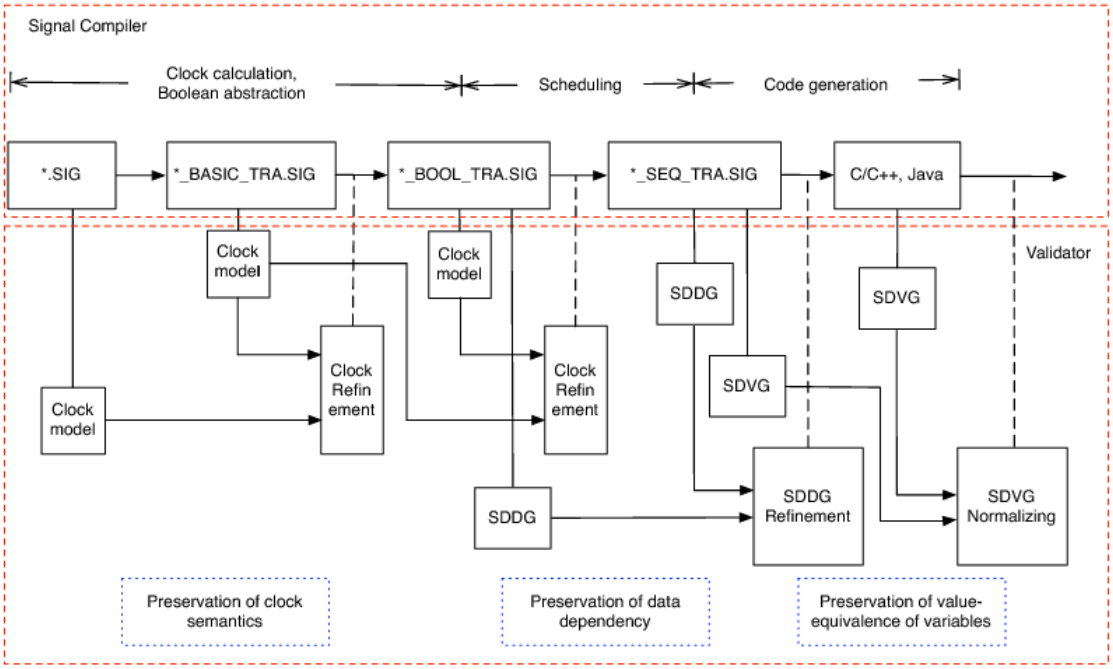

Figure 3. Our Integration within Polychrony Toolset

6.4. Ongoing integration of Polychrony with the P toolset

Participants: Christophe Junke, Loïc Besnard, Thierry Gautier, Paul Le Guernic, Jean-Pierre Talpin.

Current state of P. The FUI project P has been extended until September 2015. Partners in the project now focus on code generation aspects, leaving software architecture aspects aside. The qualifiable model-based code generator, previously known as P toolset, is now named QGen (QGen is developped mostly in Ada 2012 and Python).

Model transformation (P2S). We developped a transformation tool hereafter named `P2S` for expressing P system models as Signal processes. Our work is based on EMF (Eclipse Modelling Framework), taking advantage of the existing Ecore metamodels available for both P and SSME.

The P2S tool is written in Clojure, which is a dialect of Lisp running on the Java Virtual Machine. This approach allows to benefits from a terse and expressive language while remaining fully interoperable with existing Java libraries (including Eclipse plugins and especially Polychrony ones).

SSME abstraction layer. P2S uses an abstraction layer to simplify the creation of SSME elements, while taking into account EMF idioms. For example, the following expression creates a `ProcessModel` instance using the currently registered EMF factory:

```
(process "TestProcess"
         :in '[boolean h integer x]
         :out '[integer y]
         :body (sigdef
                (id 'y)
                (when* (id 'x) (id 'h))))
```

The newly created object can be saved as an XMI file using EMF utilities (the XMI file is 40 lines long and not shown here). This object and its children represent the following Signal process expression [0]:

```
process TestProcess =
  (? boolean h; integer x;
   ! integer y; )
  (| y := (x when h) |);
```

Transformation to P. Conversion from P to Signal relies on Clojure's multimethods. We defined a `convert` multimethod which dispatches on the type of its argument and possibly on additional modifiers. This mechanism allows to convert expressions differently depending on whether we want to produce a Signal declaration or an expression. For example, the following method specializer converts a P port as a signal declaration:

```
(defmethod convert [Port :declaration] [port & _]
  (ssme/signal-declarations
   (convert (.getDataType port))
     (ssme/with-comment
       [(readable-name port :declaration) :post]
       (ssme/id (p-name port)))))
```

Since the specializer contains the `:declaration` keyword, the previous conversion is applied only when called with that keyword given as an extra argument, as follows:

```
(convert some-port :declaration)
```

The more general specializer, which is defined below, is meant to be used inside Signal expressions and, as such, only returns a Signal identifier:

```
(defmethod convert Port [port]
   (ssme/id (p-name port)))
```

Note also that thanks to class inheritance, the above methods are sufficient to convert all kind of P ports (input/output, data/control).

The naming scheme for the resulting SSME elements is handled by the `p-name` multi-method and relies on XMI identifiers of the original P elements: XMI identifiers generated by QGen are string representations of positive integers. Moreover, those identifiers are guaranteed to be unique in a model. These two properties allows to generate valid Signal identifiers while ensuring traceability (e.g. signal `P101` links to the unique port of the original model having 101 as a unique identifier).

Datatypes are currently converted as Signal predefineds types, which do not always match exactly the original types. Another partially implemented option consists in translating them as `external` types in Signal. Some types, like arrays, are converted the same way with both approaches:

[0] Even using the dedicated `signalTreeAPI` utility class, the same example would require many more lines of Java code.

```
(defmethod convert TArray [a]
  (reduce (fn [base dim]
            (ssme/array-type base (convert dim :signal)))
          (convert (.getBaseType a))
          (.getDimensions a)))
```

Conversion of arithmetic operations may also lead to predefined Signal operators (by default) or externally defined functions (incomplete). The current approach has been tested on QGen's test models and successfully translates 208 of the 227 models.

Partial block sequencing. The conversion from P models to Signal takes into account block dependencies as computed by QGen. Unfortunately, QGen's block sequencer produces a total order between blocks, with leads to over-constrained Signal models. We contributed to the model compiler by writing an alternative (Ada) package which provides: (i) a way to parameterize block sequencing, and (ii) partial ordering options.

Our implementation is not part of the qualified compiler, but available as a standalone (non-qualifiable) executable. However, during the development of this block sequencer, we were able to find and correct existing bugs in QGen's sequencer.

Perspectives. From a software development point of view, our current work needs to be packaged and better integrated with the build system of Polychrony. By the way, that existing build process itself could be slightly improved by using Maven configuration files instead of Eclipse manual plug-in management.

The use of a functional language on top of the Java Virtual Machine is an interesting aspect of our work. By allowing the abstraction layer, which currently works at the SSME level, to also access the existing Signal library, we could provide an API for writing and compiling Signal code using a domain-specific language expressed in Clojure (there already exist JNI bindings with the native library). This feature could help developpers hook into, or interact with, the existing Signal compiler in order to customize parts of the code generation strategies.

Regarding the P project, we still need to test code distribution strategies on industrial use-cases and determine how it can be exploited at the system-model level.

6.5. A synchronous annex for the AADL

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

The SAE committee on the AADL adopted our recommendations to implement a timed and synchronous behavioural annex for the standard [20]. The specification and reference implementation of this revised behavioral annex will be the focus of most our attention next year.

We propose a synchronous timing annex for the SAE standard AADL. Our approach consists of building a synchronous model of computation and communication that best fits the semantics and expressive capability of the AADL and its behavioral annex and yet requires little to know (syntactic) extension to it, i.e. to identify a synchronous core of the AADL (which prerequisites a formal definition of synchrony at hand) and define a formal design methodology to use the AADL in a way that supports formal analysis, verification and synthesis.

Our approach first identifies the core AADL concepts from which time events can be described. Then, is considers the behavior annex (BA) as the mean to model synchronous signals and traces through automata. Finally, we consider elements of the constraint annex to reason about abstractions of these signals and traces by clocks and relations among them. To support the formal presentation of these elements, we define a model of automata that comprises a transition system to express explicit transitions and constraints, in the form of a boolean formula on time, to implicitly constraint its behavior. The implementation of such an automaton amounts to composing its explicit transition system with that of the controller synthesised from its specified constraints.

6.6. New features of Polychrony

Participants: Loïc Besnard, Thierry Gautier, Paul Le Guernic.

Reduction of communications. We have developed, as a general functionality of the Signal toolbox, a means to reduce communications between two graphs, using assignment clocks and utility clocks.

For a given signal x, its assignment clock represents the instants at which it may be modified (otherwise than keeping its previous value $x\$$) while its utility clock in a given graph represents the instants at which it is effectively used in this graph.

Considering two graphs G_i and G_j with a signal x sent from G_i to G_j, containers are built above G_i and G_j in order to minimize the clock at which x must be communicated. On the sender side, the signal which has to be sent can be reduced to x_j with $x_j := x \texttt{ when } h$, where h is the lower bound of the assignment clock of x and the utility clock of x in G_j. On the receiver side, x is replaced in G_j by x_r with $x\_r := x_j \texttt{ default } x\_r\$$.

Note that this reduction is not always possible because it may introduce cycles between signals and clocks.

Experiments have been made on programs intended to the distribution of Quartz applications, with a gain of up to 40on some of them [18].

Polychronous automata. We have defined a new model of polychronous constrained automata that has been provided as semantic model for our proposal of an extension of the AADL behavioural annex [20]. An algebra of regular expressions is also defined to represent abstractions of constrained automata or, more specifically, their time constraints.

An experimental implementation of the semantic features of this "timing annex" will be provided through the Polychrony framework. For that purpose, representations of automata are introduced in the Signal toolbox of Polychrony. In a first step, we have decided to provide only a minimal extension of the Signal language itself. A new syntactic category of process model, which is an automaton model, has been introduced. States are described by the association of labels with subprocesses, as it is available in Signal, and transitions between states, at a given clock, are written as calls to *intrinsic* (predefined) processes. Constraints described as regular expressions on events should also be introduced using intrinsic processes.

Automata will be used in different ways related to stategies of compilation. In particular, they will serve as an alternative model for the code generation. For that purpose, polychronous programs are rewritten thanks to valuations of memorized boolean signals. The resulting partially valuated programs are the states of a control automaton.

Such techniques can be applied to implement endo-isochronous programs. Currently, code may be generated only for endochronous programs, for which clock hierarchy is a tree. Endo-isochronous programs are compositions of endochronous programs the "intersection" of which is also endochronous. For example, an automaton can be built to generate code when two signals are known to alternate.

6.7. Optimized Distribution of Synchronous Programs via a Polychronous Model

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

We propose a distribution methodology for synchronous programs [18], applied in particular on programs written in the Quartz language [0]. The given program is first transformed into an intermediate model of guarded actions. After user-specified partitioning, the generated sub-models are transformed into equivalent Signal processes [7]. Then, the unnecessary constraints are eliminated from the processes to avoid unnecessary synchronization. Finally, within the Signal framework, the minimal frequencies of communication and computation are computed via multi-clock calculation. This operation can efficiently reduce the communication quantity and the computation load, with no change to the interface behaviors. Along this way, an optimized data-flow network over desynchronized processing locations can be constructed.

[0] *The Synchronous Programming Language Quartz*. K. Schneider, Technical Report n. 375. University of Kaiserslautern, 2009

The presented methodology has been implemented within the integrated framework Quartz/Averest + Signal/Polychrony. To illustrate and validate this methodology, a series of examples served as case studies. Each of them has been written in the Quartz language and distributed over different processing locations using the presented optimization methodology. These case studies confirm that the optimization can bring in significant communication reduction. In the sequel, the efficient utilization of distributed systems is substantially updated.

6.8. Component-based Design of Multi-rate Systems

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

The Synchronous language Quartz is well suited for modeling mono-clocked systems. However, as based on the model of computation (MoC) synchrony, its parallelism feature excessively strengthens the synchronization. Such synchronous parallelism in particular restricts independent component design. That is, the modeling of connected components should constantly refer to each other to guarantee the achievement of desired system behavior. Hence, Quartz cannot support well the component-based system design, in particular for the distributed systems that are generally deployed over desynchronized processing locations with multi-rate clocks.

In contrast to Quartz, the polychronous language Signal is based on the MoC polychrony. As its name suggests, a polychronous program makes use of multi-rate clocks to drive its execution. One can consider that each component in the program holds its own master clock, and there is no longer a master clock for the whole program. The resulted architecture is named globally asynchronous locally synchronous (GALS) architecture.

Through integrating Quartz with Signal, a component-based methodology is proposed for designing multi-rate systems: at first, components are modeled independently to achieve local behaviors; secondly, inter-component communications are adjusted using Signal to realize intermittent synchronization. In this way, the modeling approach for mono-clocked systems evolves into a component-based modeling methodology. Such significant progress not only facilitates the component coordination, but also enhances the component reusability, in particular for modeling large scale systems.

ANTIQUE Team

6. New Results

6.1. Highlights of the Year

Patrick and Radhia Cousot have received in 2014 the IEEE Computer Society IEEE Computer Society Harlan D. Mills award for the invention of abstract interpretation, developpment of tool support and practical application http://www.computer.org/portal/web/awards/cousots.

6.2. Memory Abstraction

6.2.1. Modular Construction of Shape-Numeric Analyzers

Participants: Xavier Rival [correspondant], Bor-Yuh Evan Chang [University of Colorado, Boulder, USA], Huisong Li, Antoine Toubhans.

Abstract interpretation, Memory abstraction, Shape abstract domains. In [24], we discuss the modular construction of memory abstract domains.

The aim of static analysis is to infer invariants about programs that are tight enough to establish semantic properties, like the absence of run-time errors. In the last decades, several branches of the static analysis of imperative programs have made significant progress, such as in the inference of numeric invariants or the computation of data structures properties (using pointer abstractions or shape analyzers). Although simultaneous inference of shape-numeric invariants is often needed, this case is especially challenging and less well explored. Notably, simultaneous shape-numeric inference raises complex issues in the design of the static analyzer itself. We studied the modular construction of static analyzers, based on combinations of atomic abstract domains to describe several kinds of memory properties and value properties.

6.2.2. An abstract domain combinator for separately conjoining memory abstractions

Participants: Xavier Rival [correspondant], Bor-Yuh Evan Chang [University of Colorado, Boulder, USA], Antoine Toubhans.

Abstract interpretation, Memory abstraction, Shape abstract domains. In [25], we studied the separating combination of heap abstract domains.

The breadth and depth of heap properties that can be inferred by the union of today's shape analyses is quite astounding. Yet, achieving scalability while supporting a wide range of complex data structures in a generic way remains a long-standing challenge. We proposed a way to side-step this issue by defining a generic abstract domain combinator for combining memory abstractions on disjoint regions. In essence, our abstract domain construction is to the separating conjunction in separation logic as the reduced product construction is to classical, non-separating conjunction. This approach eases the design of the analysis as memory abstract domains can be re-used by applying our separating conjunction domain combinator. And more importantly, this combinator enables an analysis designer to easily create a combined domain that applies computationally-expensive abstract domains only where it is required.

6.2.3. Abstraction of Arrays Based on Non Contiguous Partitions

Participants: Xavier Rival [correspondant], Jiangchao Liu.

Abstract interpretation, Memory abstraction, Array abstract domains. In [20], we studied array abstractions.

Array partitioning analyses split arrays into contiguous partitions to infer properties of cell sets. Such analyses cannot group together non contiguous cells, even when they have similar properties. We proposed an abstract domain which utilizes semantic properties to split array cells into groups. Cells with similar properties will be packed into groups and abstracted together. Additionally, groups are not necessarily contiguous. This abstract domain allows to infer complex array invariants in a fully automatic way. Experiments on examples from the Minix 1.1 memory management demonstrated its effectiveness.

6.3. Static analysis of JavaScript applications

6.3.1. Automatic Analysis of Open Objects in Dynamic Language Programs

Participants: Arlen Cox [correspondant], Bor-Yuh Evan Chang [University of Colorado, Boulder, USA], Xavier Rival.

Abstract interpretation, Dynamically typed languages, Verification In [14], we have studied the abstraction of open objects in dynamic language programs (like JavaScript).

In dynamic languages, objects are open: they support iteration over and dynamic addition/deletion of their attributes. Open objects, because they have an unbounded number of attributes, are difficult to abstract without a priori knowledge of all or nearly all of the attributes and thus pose a significant challenge for precise static analysis. To address this challenge, we presented the HOO (Heap with Open Objects) abstraction that can precisely represent and infer properties about open-object-manipulating programs without any knowledge of specific attributes. It achieves this by building upon a relational abstract domain for sets that is used to reason about partitions of object attributes. An implementation of the resulting static analysis is used to verify specifications for dynamic language framework code that makes extensive use of open objects, thus demonstrating the effectiveness of this approach.

6.3.2. Desynchronized Multi-State Abstractions for Open Programs in Dynamic Languages

Participants: Arlen Cox [correspondant], Bor-Yuh Evan Chang [University of Colorado, Boulder, USA], Xavier Rival.

Abstract interpretation, Dynamically typed languages, Verification In [15], we have studied desynchronized multi-state abstractions for open programs in dynamic languages (libraries).

Dynamic language library developers face a challenging problem: ensuring that their libraries will behave correctly for a wide variety of client programs without having access to those client programs. This problem stems from the common use of two defining features for dynamic languages: callbacks into client code and complex manipulation of attribute names within objects. To remedy this problem, we introduced two state-spanning abstractions. To analyze callbacks, the first abstraction desynchronizes a heap, allowing partitions of the heap that may be affected by a callback to an unknown function to be frozen in the state prior to the call. To analyze object attribute manipulation, building upon an abstraction for dynamic language heaps, the second abstraction tracks attribute name/value pairs across the execution of a library. We implemented these abstractions and use them to verify modular specifications of class-, trait-, and mixin-implementing libraries.

6.4. Static analysis of Spreadsheet applications

Participants: Tie Cheng [correspondant], Xavier Rival.

Abstract interpretation, Spreadsheet applications, Verification In [13], we have proposed a static analysis to detect type unsafe operations in spreadsheet applications including formulas and macros.

Spreadsheets are widely used, yet are error-prone: they use a weak type system, allowing certain operations that will silently return unexpected results, like comparisons of integer values with string values. However, discovering these issues is hard, since data and formulas can be dynamically set, read or modified. We defined a static analysis that detects all run-time type-unsafe operations in spreadsheets. It is based on an abstract interpretation of spreadsheet applications, including spreadsheet tables, global re-evaluation and associated programs. Our implementation supports the features commonly found in real-world spreadsheets. We ran our analyzer on the EUSES Spreadsheet Corpus. This evaluation shows that our tool is able to automatically verify a large number of real spreadsheets, runs in a reasonable time and discovers complex bugs that are difficult to detect by code review or by testing.

6.5. Mechanically Verifying a Shape Analysis

Participant: Arnaud Spiwack.

Program verification, Abstract interpretation, Static analysis, Shape analysis, Coq. The result of a static analysis is only as good as the trust put into its correctness. For critical software, the standards are very high, and trusting a complex tool requires costly inspection of its implementation. Mechanically proving the correctness of static analysers is a way to lower these costs: the exigence of trust is moved from various complex dedicated tools to a single simpler general purpose one.

In this context, Arnaud Spiwack worked on an ongoing Coq implementation and certification of a shape abstract domain. The implementation, named Cosa, is based on Evan Chang and Xavier Rival's Xisa. It targets an intermediary language of Xavier Leroy's Compcert C, and interfaces with the domains of the Verasco project.

The development of Cosa lead Arnaud Spiwack to express the abstract interpretation correctness property in term of refinement calculus, which allowed to use interaction structures (a type theoretic variant of the refinement calculus) as a central structuring element of Cosa. Arnaud Spiwack started investigating how the technology of nominal sets could be leveraged to prove the correctness of unfolding (which involves choosing new names) in Cosa.

6.6. Static Analysis of Embedded Critical Concurrent Software

6.6.1. AstréeA: A Static Analyzer for Large Embedded Multi-Task Software

Participant: Antoine Miné.

In [11], we present the design, implementation and experimentation of the ASTRÉEA static analyzer, an extension of the ASTRÉE static analyzer dedicated to analyzing the run-time errors in embedded critical concurrent software. Such software are already present in critical systems and will likely become the norm with the generalization of multi-core processors in embedded systems, leading to new challenging demands in verification. One major challenge is that a concurrent program execution does not follow a fixed sequential order, but one of many interleavings of executions from different tasks chosen by the scheduler. As it is impractical to build a fully flow-sensitive analysis by enumerating explicitly all interleavings, we took inspiration from thread-modular methods: we analyze each thread individually, in an environment consisting of (an abstraction of) the effect of the other threads. This is a form of rely-guarantee reasoning, but in a fully automatic static analysis settings formalized as abstract interpretation: a thread-modular static analysis is viewed as a computable abstraction of a complete concrete, fixpoint-based thread-modular semantics. This permits a fine control between precision and efficiency, and opens the way to analysis specialization: any given safety property of a given program can be theoretically inferred given the right abstract domain. The presentation describes our subsequent work in improving the precision of ASTRÉEA by specialization on our target applications, and the interesting abstractions we developed along the way. For instance, we developed new interference abstractions enabling a limited but controllable (for efficiency) degree of relationality and flow-sensitivity. We also designed abstractions able to exploit our knowledge of the real-time scheduler used in the analysis target: i.e., it schedules tasks on a single core and obeys a strict priority scheme. The end-result is a more precise analyzer on our target applications, with currently around a thousand alarms.

6.6.2. Static Analysis by Abstract Interpretation of Concurrent Programs under the TSO Weak Memory Model

Participants: Thibault Suzanne, Antoine Miné.

In [33], we present an abstract semantics for the Total Store Ordering (TSO) memory model, a weakly consistent memory model used in major multi-core processors. This abstraction forgets some information about the order in which variables are written into by each thread. This results in a much simplified concrete semantics, but which is still not computable. We then express the semantics based on partitioned sets of points in a vector space, which allows applying classic methods from abstract interpretation (such as numeric abstract domains) to achieve a fully computable abstract semantics and automatically infer an over-approximation of the set of reachable states of a program running under the TSO memory model. The method is proved correct and, in certain cases, optimal, using the standard tools of abstraction interpretation (Galois connections).

Moreover, we have written a prototype static analyzer for simple program fragments written in an assembly-like language, and experimented our abstraction on a few small examples.

6.7. Inference of Termination and Liveness properties

6.7.1. A Decision Tree Abstract Domain for Proving Conditional Termination

Participants: Caterina Urban, Antoine Miné.

In [26], we present a new parameterized abstract domain able to refine existing numerical abstract domains with finite disjunctions. The elements of the abstract domain are decision trees where the decision nodes are labeled with linear constraints, and the leaf nodes belong to a numerical abstract domain. The abstract domain is parametric in the choice between the expressivity and the cost of the linear constraints for the decision nodes (e.g., polyhedral or octagonal constraints), and the choice of the abstract domain for the leaf nodes. We describe an instance of this domain based on piecewise-defined ranking functions for the automatic inference of sufficient preconditions for program termination. We have implemented a static analyzer for proving conditional termination of programs written in (a subset of) C and, using experimental evidence, we show that it performs well on a wide variety of benchmarks, it is competitive with the state of the art and is able to analyze programs that are out of the reach of existing methods.

6.7.2. An Abstract Domain to Infer Ordinal-Valued Ranking Functions

Participants: Caterina Urban, Antoine Miné.

The traditional method for proving program termination consists in inferring a ranking function. In many cases (i.e. programs with unbounded non-determinism), a single ranking function over natural numbers is not sufficient. In [30], we propose a new abstract domain to automatically infer ranking functions over ordinals. We extend an existing domain for piecewise-defined natural-valued ranking functions to polynomials in ω, where the polynomial coefficients are natural-valued functions of the program variables. The abstract domain is parametric in the choice of the state partitioning inducing the piecewise-definition and the type of functions used as polynomial coefficients. To our knowledge this is the first abstract domain able to reason about ordinals. Handling ordinals leads to a powerful approach for proving termination of imperative programs, which in particular allows us to take a first step in the direction of proving termination under fairness constraints and proving liveness properties of (sequential and) concurrent programs.

6.7.3. Proving Guarantee and Recurrence Temporal Properties by Abstract Interpretation

Participants: Caterina Urban, Antoine Miné.

We present in [28] a new static analysis methods for proving liveness properties of programs. In particular, with reference to the hierarchy of temporal properties proposed by Manna and Pnueli, we focus on guarantee (i.e., “something good occurs at least once”) and recurrence (i.e., “something good occurs infinitely often”) temporal properties. We generalize the abstract interpretation framework for termination presented by Cousot and Cousot. Specifically, static analyses of guarantee and recurrence temporal properties are systematically derived by abstraction of the program operational trace semantics. These methods automatically infer sufficient preconditions for the temporal properties by reusing existing numerical abstract domains based on piecewise-defined ranking functions. We augment these abstract domains with new abstract operators, including a dual widening. To illustrate the potential of the proposed methods, we have implemented a research prototype static analyzer, for programs written in a C-like syntax, that yielded interesting preliminary results.

6.8. Numeric Invariant Inference

6.8.1. A Numeric Abstract Domain to Infer Octagonal Constraints with Absolute Value

Participants: Liqian Chen [National Laboratory for Parallel and Distributed Processing, National University of Defense Technology, Changsha, P.R.China], Jiangchao Liu, Antoine Miné, Deepak Kapur [University of New Mexico, USA], Ji Wang [National Laboratory for Parallel and Distributed Processing, National University of Defense Technology, Changsha, P.R.China].

The octagon abstract domain, devoted to discovering octagonal constraints (also called Unit Two Variable Per Inequality or UTVPI constraints) of a program, is one of the most commonly used numerical abstractions in practice, due to its quadratic memory complexity and cubic time complexity. However, the octagon domain itself is restricted to express convex sets and has limitations in handling non-convex properties which are sometimes required for proving some numerical properties in a program. In [12], we intend to extend the octagon abstract domain with absolute value, to infer certain non-convex properties by exploiting the absolute value function. More precisely, the new domain can infer relations of the form $\{ \pm X \pm Y \leq c,\ \pm X \pm |Y| \leq d,\ \pm|X| \pm |Y| \leq e \}$. We provide algorithms for domain operations such that the new domain still enjoys the same asymptotic complexity as the octagon domain. Moreover, we present an approach to support strict inequalities over rational or real-valued variables in this domain, which also fits for the octagon domain. Experimental results of our prototype are encouraging; The new domain is scalable and able to find non-convex invariants of interest in practice but without too much overhead (compared with that using octagons).

6.8.2. *A Method to Infer Inductive Numeric Invariants Inspired from Constraint Programming.*

Participant: Antoine Miné.

In [29], we suggest the idea of using algorithms inspired by Constraint Programming in order to infer inductive invariants on numeric programs. Similarly to Constraint Programming solvers on continuous domains, our algorithm approximates the problem from above, using decreasing iterations that may split, discard, and tighten axis-aligned boxes. If successful, the algorithm outputs a set of boxes that includes the initial states and is a post-fixpoint of the abstract semantic function of interest. Our work is very preliminary; many improvements still need to be performed to determine if the method is usable in practice, and in which contexts. Nevertheless, we show that a naive proof-of-concept implementation of our algorithm is already capable of inferring non-trivial inductive invariants that would otherwise require the use of relational or even non-linear abstract domains when using more traditional abstract interpretation iteration methods.

6.9. Bisimulation Metrics

6.9.1. *Bisimulation for Markov Decision Processes through Families of Functional Expressions*

Participants: Norman Ferns, Sophia Knight [LIX, France], Doina Precup [McGill University, Canada].

Markov decision processes, Bisimulation, Metrics.

In [17], we have transfered a notion of quantitative bisimilarity for labelled Markov processes [51] to Markov decision processes with continuous state spaces. This notion takes the form of a pseudometric on the system states, cast in terms of the equivalence of a family of functional expressions evaluated on those states and interpreted as a real-valued modal logic. Our proof amounts to a slight modification of previous techniques [56], [55] used to prove equivalence with a fixed-point pseudometric on the state-space of a labelled Markov process and making heavy use of the Kantorovich probability metric. Indeed, we again demonstrate equivalence with a fixed-point pseudometric defined on Markov decision processes [52]; what is novel is that we recast this proof in terms of integral probability metrics [54] defined through the family of functional expressions, shifting emphasis back to properties of such families. The hope is that a judicious choice of family might lead to something more computationally tractable than bisimilarity whilst maintaining its pleasing theoretical guarantees. Moreover, we use a trick from descriptive set theory to extend our results to MDPs with bounded measurable reward functions, dropping a previous continuity constraint on rewards and Markov kernels.

6.9.2. *Bisimulation Metrics are Optimal Value Functions*

Participants: Norman Ferns, Doina Precup [McGill University, Canada].

Markov decision processes, Bisimulation, Metrics.

Bisimulation is a notion of behavioural equivalence on the states of a transition system. Its definition has been extended to Markov decision processes, where it can be used to aggregate states. A bisimulation metric is a quantitative analog of bisimulation that measures how similar states are from a the perspective of long-term behavior. Bisimulation metrics have been used to establish approximation bounds for state aggregation and other forms of value function approximation. In [18], we prove that a bisimulation metric defined on the state space of a Markov decision process is the optimal value function of an optimal coupling of two copies of the original model. We prove the result in the general case of continuous state spaces. This result has important implications in understanding the complexity of computing such metrics, and opens up the possibility of more efficient computational methods.

6.10. Abstraction of Rule-Based Biological Models

6.10.1. Stochastic fragments: A framework for the exact reduction of the stochastic semantics of rule-based models

Participants: Jérôme Feret, Heinz Koeppl [École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland], Tatjana Petrov [École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland].

Protein-protein interaction networks, Stochastic systems, Backward bisimulations, Model reduction. In [9], we propose an abstract interpretation-based framework for reducing the state space of stochastic semantics for protein-protein interaction networks. Our approach consists in quotienting the state space of networks. Yet interestingly, we do not apply the widely-used strong lumpability criterion which imposes that two equivalent states behave similarly with respect to the quotient, but a weak version of it. More precisely, our framework detects and proves some invariants about the dynamics of the system: indeed the quotient of the state space is such that the probability of being in a given state knowing that this state is in a given equivalence class, is an invariant of the semantics. Then we introduce an individual-based stochastic semantics (where each agent is identified by a unique identifier) for the programs of a rule-based language (namely Kappa) and we use our abstraction framework for deriving a sound population-based semantics and a sound fragments-based semantics, which give the distribution of the traces respectively for the number of instances of molecular species and for the number of instances of partially defined molecular species. These partially defined species are chosen automatically thanks to a dependency analysis which is also described in [9].

6.10.2. An algebraic approach for inferring and using symmetries in rule-based models

Participant: Jérôme Feret.

Graph rewriting, Single-pushout semantics, Symmetries, Bisimulations, Model reduction. Symmetries arise naturally in rule-based models, and under various forms. Besides automorphisms between site graphs, which are usually built within the semantics, symmetries can take the form of pairs of sites having the same capabilities of interactions, of some protein variants behaving exactly the same way, or of some linear, planar, or 3D molecular complexes which could be seen modulo permutations of their axis and/or mirror-image symmetries. In [16], we propose a unifying handling of symmetries in Kappa. We follow an algebraic approach, that is based on the single pushout semantics of Kappa. We model classes of symmetries as finite groups of transformations between site graphs, which are compatible with the notion of embedding (that is to say that it is always possible to restrict a symmetry that is applied with the image of an embedding to the domain of this embedding) and we provide some assumptions that ensure that symmetries are compatible with pushouts. Then, we characterise when a set of rules is symmetric with respect to a group of symmetries and, in such a case, we give sufficient conditions so that this group of symmetries induces a forward bisimulation and/or a backward bisimulation over the population semantics.

6.11. Model checking of Logical Biological Models

6.11.1. Model checking logical regulatory networks

Participants: Pedro T. Monteiro [INESC-ID, Lisboa, Portugal], Wassim Abou-Jaoudé, Denis Thieffry [IBENS, France], Claudine Chaouiya [IGC, Oeiras, Portugal].

Model checking, Regulatory networks. Regulatory and signalling networks control cell behaviours in response to environmental cues. The logical formalism has been widely employed to study these interaction networks, which are modelled as discrete dynamical systems. While biologists identify networks encompassing more and more components, properties of biological relevance become hard to verify.

In [22], we report on the use of model-checking techniques to address this challenge. This approach is illustrated by an application dealing with the modelling of T-helper lymphocyte differentiation.

6.11.2. Model checking to assess T-helper cell plasticity

Participants: Wassim Abou-Jaoudé, Pedro T. Monteiro [INESC-ID, Lisboa, Portugal], Aurélien Naldi [Centre Intégratif de Lausanne, Lausanne, Switzerland], Maximilien Grandclaudon [Institut Curie, Paris, France], Vassili Sommeils [Institut Curie, Paris, France], Claudine Chaouiya [IGC, Oeiras, Portugal], Denis Thieffry [IBENS, France].

Model checking, Logical modeling. Computational modeling constitutes a crucial step toward the functional understanding of complex cellular networks. In particular, logical modeling has proven suitable for the dynamical analysis of large signaling and transcriptional regulatory networks. In this context, signaling input components are generally meant to convey external stimuli, or environmental cues. In response to such external signals, cells acquire specific gene expression patterns modeled in terms of attractors (e.g., stable states).The capacity for cells to alter or reprogram their differentiated states upon changes in environmental conditions is referred to as cell plasticity. In this article, we present a multivalued logical framework along with computational methods recently developed to efficiently analyze large models.We mainly focus on a symbolic model checking approach to investigate switches between attractors subsequent to changes of input conditions. As a case study, we consider the cellular network regulating the differentiation of T-helper (Th) cells, which orchestrate many physiological and pathological immune responses. To account for novel cellular subtypes, we present, in [8], an extended version of a published model of Th cell differentiation.We then use symbolic model checking to analyze reachability properties between Th subtypes upon changes of environmental cues. This allows for the construction of a synthetic view of Th cell plasticity in terms of a graph connecting subtypes with arcs labeled by input conditions. Finally, we explore novel strategies enabling specific Th cell polarizing or reprograming events.

CELTIQUE Project-Team

5. New Results

5.1. Browser randomization against web tracking

Participants: Frédéric Besson, Thomas Jensen.

We have investigated different approaches for dynamically tracking information flows in order to improve web browser security. We have identified the problem of stateless web tracking (fingerprinting) and have proposed a novel approach to hybrid information flow monitoring by tracking the knowledge about secret variables using logical formulae. In a follow-up work we investigated how to enforce browser anonymity in the presence of finger-printing web trackers. One way to protect the users' privacy is to make them switch between different machine and browser configurations. We propose a formalisation of this privacy enforcement mechanism. We use information-theoretic channels to model the knowledge of the tracker and the fingerprinting program, and show how to synthesise a randomisation mechanism that defines the distribution of configurations for each user. This mechanism provides a strong guarantee of *privacy* (the probability of identifying the user is bounded by a given threshold) while maximising *usability* (the user switches to other configurations rarely). To find an optimal solution, we express the enforcement problem of randomisation by a linear program. We investigate and compare several approaches to randomisation and find that more efficient privacy enforcement would often provide lower usability. Finally, we relax the requirement of knowing the fingerprinting program in advance, by proposing a randomisation mechanism that guarantees privacy for an arbitrary program.

5.2. Static analysis of functional programs using tree automata and term rewriting

Participants: Thomas Genet, Barbara Kordy, 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. This is a specific form of, so-called, Regular Tree Model Checking. However, when dealing with infinite-state systems, Regular Tree Model Checking approaches may have some difficulties to represent infinite sets of data. We proposed Lattice Tree Automata, an extended version of tree automata to represent complex data domains and their related operations in an efficient manner. Moreover, we introduce a new completion-based algorithm for computing the possibly infinite set of reachable states in a finite amount of time. This algorithm is independent of the lattice making it possible to seamlessly plug abstract domains into a Regular Tree Model Checking algorithm. These results are part of Valérie Murat's PhD thesis [13]. Now, we aim at applying this technique to the static analysis of programming languages whose semantics is based on terms, like functional programming languages. We already shown that static analysis of first order functional programs can be automated using tree automata completion [28]. Now, one of the objective is to lift those results to the static analysis of higher-order functions. This was precisely the purpose of Yann Salmon's visit to Pr. Luke Ong. Barbara Kordy who joined Celtique in September 2014 is also going to work on this subject.

5.3. Certified JavaScript

Participants: Martin Bodin, Alan Schmitt.

We have completed our first milestone in the development of a certified JavaScript semantics. We have finished a first version of JSCert, a formalization of the current ECMA standard in the Coq proof assistant, and JSRef, a reference interpreter for JavaScript extracted from Coq to OCaml. We have also given a Coq proof that JSRef is correct with respect to JSCert and assessed JSRef using test262, the ECMA conformance test suite. Our methodology ensures that JSCert is a comparatively accurate formulation of the English standard. We have demonstrated that modern techniques of mechanized specification can handle the complexity of JavaScript. This result, obtained in the setting of a collaboration with Philippa Gardner and Sergio Maffeis of Imperial College, and Arthur Charguéraud of Inria Saclay, have been published in the conference Principles of Programming Languages [25].

5.4. SawjaCard: a static analysis tool for certifying Java Card applications

Participants: Frédéric Besson, Thomas Jensen, David Pichardie, Delphine Demange.

We have transfered to the FIME company a static analysis tool for certifying *Java Card* applications, according to security rules defined by the smart card industry. *Java Card* is a dialect of Java designed for programming multi-application smart cards and the tool, called *SawjaCard*, has been specialised for the particular *Java Card* programming patterns. The tool is built around a static analysis engine which uses a combination of numeric and heap analysis. It includes a model of the *Java Card* libraries and the *Java Card* firewall. The tool has been evaluated on a series of industrial applets and is shown to automate a substantial part of the validation process [21].

5.5. Semantics for C programs

Participants: Frédéric Besson, Sandrine Blazy, Pierre Wilke.

Real life C programs are often written using C dialects which, for the ISO C standard, have undefined behaviours. In particular, according to the ISO C standard, reading an uninitialised variable has an undefined behaviour and low-level pointer operations are implementation defined. We propose a formal semantics which gives a well-defined meaning to those behaviours for the C dialect of the CompCert compiler. Our semantics builds upon a novel memory model leveraging a notion of symbolic values. Symbolic values are used by the semantics to delay the evaluation of operations and are normalised lazily to genuine values when needed. We show that the most precise normalisation is computable and that a slightly relaxed normalisation can be efficiently implemented using an SMT solver. The semantics is executable and our experiments show that the enhancements of our semantics are mandatory to give a meaning to low-levels idioms such as those found in the allocation functions of a C standard library [21].

5.6. Fast inference of polynomial invariants

Participants: David Cachera, Thomas Jensen.

We have developed our static analysis techniques for computing polynomial invariants for imperative programs. The analysis is derived from an abstract interpretation of a backwards semantics, and computes preconditions for equalities of the form g = 0 to hold at the end of execution. A distinguishing feature of the technique is that it computes polynomial loop invariants without resorting to Gro¨bner base computations. The analysis uses remainder computations over parameterized polynomials in order to handle conditionals and loops efficiently. The algorithm can analyze and find a large majority of loop invariants reported previously in the literature, and executes significantly faster than implementations using Gro¨bner bases [15].

5.7. Quantitative analysis of security

Participant: Barbara Kordy.

Graphical models for security is a young but rapidly growing research field. Security models based on graphs combine intuitive, visual representation with rigorous, mathematical foundations. In [30] we address the growing need of performing meaningful probabilistic analysis of security using graphical models. We propose a framework that integrates the modeling technique of attack–defense trees with probabilistic information expressed in terms of Bayesian networks. This allows us to perform probabilistic evaluation of attack–defense scenarios involving dependent actions. To improve the efficiency of our computations, we make use of inference algorithms from Bayesian networks and encoding techniques from constraint reasoning. We discuss the algebraic theory underlying our framework and point out several generalizations which are possible thanks to the use of semiring theory

5.8. Formal Verification of an SSA-Based Middle-End for CompCert

Participants: Delphine Demange, David Pichardie.

CompCert is a formally verified compiler that generates compact and efficient code for a large subset of the C language. However, CompCert foregoes using SSA, an intermediate representation employed by many compilers that enables writing simpler, faster optimizers. In fact, it has remained an open problem to verify formally an SSA-based compiler. We report in [14] on a formally verified, SSA-based middle-end for CompCert. In addition to providing a formally verified SSA-based middle-end, we address two problems raised by Leroy in 2009: giving an intuitive formal semantics to SSA, and leveraging its global properties to reason locally about program optimizations. Joint work with Gilles Barthe.

5.9. A verified information-flow architecture

Participants: Delphine Demange, David Pichardie.

SAFE is a clean-slate design for a highly secure computer system, with pervasive mechanisms for tracking and limiting information flows. At the lowest level, the SAFE hardware supports fine-grained programmable tags, with efficient and flexible propagation and combination of tags as instructions are executed. The operating system virtualizes these generic facilities to present an information-flow abstract machine that allows user programs to label sensitive data with rich confidentiality policies. We present a formal, machine-checked model of the key hardware and software mechanisms used to control information flow in SAFE and an end-to-end proof of noninterference for this model in the Coq proof assistant [17]. This work has been obtained in collaboration with colleagues from University of Pennsylvania, Portland State University, and Harvard University, as part of the CRASH-SAFE project, funded by DARPA.

5.10. Formal Verification of Static Analysis

Participants: Sandrine Blazy, Vincent Laporte, David Pichardie.

Static analysis of binary code is challenging for several reasons. In particular, standard static analysis techniques operate over control flow graphs, which are not available when dealing with self-modifying programs which can modify their own code at runtime. We formalized in the Coq proof assistant some key abstract interpretation techniques that automatically extract memory safety properties and control flow graphs from binary code [22], and operate over a small subset of the x86 assembly. Our analyzer is formally proved correct and has been run on several self-modifying challenges, provided by Cai et al. in their PLDI 2007 paper.

DEDUCTEAM Exploratory Action

6. New Results

6.1. Highlights of the Year

In the framework of the *BWare* project, Pierre Halmagrand, David Delahaye, Damien Doligez, and Olivier Hermant designed a new version of the *B* set theory using deduction modulo, in order to automatically verify a large part of the proof obligations of the benchmark of *BWare*, which consists of proof obligations coming from the modeling of industrial applications (about 13,000 proof obligations). Using this *B* set theory modulo with *Zenon Modulo*, as well as some other extensions of *Zenon*, such as typed proof search and arithmetic (implemented by Guillaume Bury), we are able to automatically verify more than 95% of the proof obligations of *BWare*, while the regular version of *Zenon* is only able to prove less than 1% of these proof obligations. This is a real breakthrough for the *BWare* project, but also for automated deduction in general, as it tends to show that deduction modulo is the way to go when reasoning modulo theories.

6.2. Termination

Frédéric Blanqui, together with Jean-Pierre Jouannaud (Univ. Paris 11) and Albert Rubio (Technical University of Catalonia), have finished their work on a new version of the higher-order recursive path ordering (HORPO) [44], [43], a decidable monotone well-founded relation that can be used for proving the termination of higher-order rewrite systems by checking that rules are included in it. This new version, called the computability path ordering (CPO), appears to be the ultimate improvement of HORPO in the sense that this definition captures the essence of computability arguments *à la* Tait and Girard [37], therefore explaining the name of the improved ordering. It has been shown that CPO allows to consider higher-order rewrite rules in a simple type discipline with inductive types, that most of the guards present in the recursive calls of its core definition cannot be relaxed in any natural way without losing well-foundedness, and that the precedence on function symbols cannot be made more liberal anymore. This new result is described in a 41-pages papers available on Frédéric Blanqui's web page which has been submitted to a journal for publication. A Prolog implementation of CPO is also available on Albert Rubio's web page.

Frédéric Blanqui revised his work on the compatibility of Tait and Girard's notion of computability for proving the termination of higher-order rewrite systems when matching is done modulo $\beta\eta$-equivalence. In particular, he showed that computability is preserved by leaf-β-expansion, a key property for dealing with higher-order pattern-matching. This work is described in a 46-pages paper available on his web page which has been submitted to a journal for publication.

Frédéric Blanqui did some historical investigations on fixpoint theorems in posets used for instance for defining the semantics of non-basic inductive types (i.e. types with constructors taking functions as arguments) and the termination of functions defined by induction on such non-basic inductive types. These theorems assume the function either extensive or monotone. However, as shown by Salinas in [48], these two conditions can be subsumed by a more general one. Frédéric Blanqui slightly improved this condition further by using results by Hartogs, Rubin and Rubin, and Abian and Brown. This work is described in a 10-pages note available on his web page [20].

Kim Quyen Ly finished the development of a new version faster, safer (proved correct in Coq) and standalone version of Rainbow, based on Coq extraction mechanism. She defended her PhD thesis [11] on the automated verification of termination certificates in October.

6.3. Proof and type theory modulo rewriting

Ali Assaf defined a sound and complete embedding of the cumulative universe hierarchy of the *calculus of inductive constructions* (CIC) in the $\lambda\Pi$-calculus modulo rewriting [18]. By reformulating universes in the Tarski style, he showed that we can make cumulativity explicit without losing any typing power. This result refines the translation used by Coqine, which was unsound because it collapsed the universe hierarchy to a single type universe. It also sheds some light on the metatheory of Coq and its connection to Martin-Löf's intuitionistic type theory. This work was presented at the TYPES meeting in Paris.

Frédéric Gilbert and Olivier Hermant defined new encodings from classical to intuitionistic first-order logic. These encodings, based on the introduction of double negations in formulas, are tuned to satisfy two purposes jointly: basing their specifications on the definition of *classical connectives* inside intuitionistic logic – which is the property of *morphisms*, and reducing their impact on the shape and size of formulas, by limiting as much as possible the number of negations introduced. This paper has been submitted.

Raphael Cauderlier and Catherine Dubois defined a shallow embedding of an object calculus (formalized by Abadi and Cardelli), in the $\lambda\Pi$-calculus modulo rewriting . The main result concerns the encoding of subtyping. This encoding shows that rewriting is an effective help for handling of subtyping proofs. The implementation in Dedukti, Sigmaid. This work has been presented at the TYPES 2014 meeting in Paris. A paper has been submitted.

Ali Assaf, Olivier Hermant and Ronan Saillard defined a rewrite system such that all strongly normalizable proof term can be typed in Natural Deduction modulo this rewrite system. This work is inspired by Statman's work [49], and can be understood as an encoding of intersection types.

Guillaume Burel showed how to get rewriting systems that admit cut by using standard saturation techniques from automated theorem proving, namely ordered resolution with selection, and superposition. This work relies on a view of proposition rewriting rules as oriented clauses, like term rewriting rules can be seen as oriented equations. This also lead to introduce an extension of deduction modulo with *conditional* term rewriting rules. This work was presented at the RTA-TLCA conference in Vienna [15].

Gilles Dowek, has generalized the notion of super-consistency to the lambda-Pi-calculus modulo theory and proved this way the termination of the embedding of various formulations of Simple Type Theory and of the Calculus of Constructions in the Lambda-Pi calculs modulo theory.

Gilles Dowek and Alejandro Díaz-Caro have finished their work on the extension of Simply Typed Lambda-Calculus with Type Isomorphisms. This work has been presented at the Types meeting and recently accepted for publication in the Theoretical Computer Science journal [26].

Gilles Dowek and Ying Jiang have given a new proof of the decidability of reachability in alternating pushdown systems, based on a cut-elimination theorem.

Vaston Costa presented to the group a new structure to represent proofs through references rather than copy. The structure, called Mimp-graph, was initially developed for minimal propositional logic but the results have been extended to first-order logic. Mimp-graph preserves the ability to represent any Natural Deduction proof and its minimal formula representation is a key feature of the mimp-graph structure, it is easy to distinguish maximal formulas and an upper bound in the length of the reduction sequence to obtain a normal proof. Thus a normalization theorem can be proved by counting the number of maximal formulas in the original derivation. The strong normalization follow as a direct consequence of such normalization, since that any reduction decreases the corresponding measures of derivation complexity. Sharing for inference rules is performed during the process of construction of the graph. This feature is very important, since we intend to use this graph in automatic theorem provers.

6.4. Automated theorem proving

Guillaume Bury defined a sound and complete extension of the tableaux method to handle linear arithmetic. The rules are based on a variant of the simplex algorithm for rational and real linear arithmetic, and a Branch&Bound algorithm for integer arithmetic.

Guillaume Bury defined an encoding of analytical tableaux rules as a theory for smt solvers. The theory acts like a lazy cnf conversion during the proof search and allows to integrate the cnf conversion into the resolution proof for unsatisfiable formulas. This work was implemented in mSAT.

Simon Cruanes added many improvements to Logtk, in particular a better algorithm to reduce formulas to Clausal Normal Form. A presentation of its design and implementation has been made at PAAR 2014[16]. He also used Zipperposition as a testbed for integer linear arithmetic; a sophisticated inference system for this fragment of arithmetic was designed and implemented in Zipperposition, including many redundancy criteria and simplification rules that make it efficient in practice. The arithmetic-enabled Zipperposition version entered CASC-J7, the annual competition of Automated Theorem Provers, in the first-order theorems with linear arithmetic division where it had very promising results (on integer problems only, since Zipperposition doesn't handle rationals).

Another extension of Zipperposition has been performed by Julien Rateau, Simon Cruanes, and David Delahaye, in order to deal with a fragment of set theory in the same vein as the *STR*$\dot{+}$*VE*$\subseteq$ prover [40]. This extension relies on a specific normal form of literal, which only involves the $\subseteq$, $\cap$, $\cup$, and complement set operators. In the future, the idea is to use this extension in the framework of the *BWare* project to verify *B* proof obligations coming from industrial benchmarks.

The current effort of research on Zipperposition focuses on extending superposition to handle structural induction, following the work from [45]. The current prototype is able to prove simple properties on natural numbers, binary trees and lists.

Kailiang Ji defined a set of rewrite rules for the equivalence between CTL formulas (denote them as R_{CTL}), by taking them as terms of designed predicates. For a given transition system model, we transform it into a set of rewrite rules (denote them as R_m). Then any CTL property of the transition system can be proved in deduction modulo $R_{CTL} \cup R_m$, by specifying the model checking problems into designed first-order formulas. This method was implemented in iProver Modulo, and the experimental evaluation was reported in workshop of Locali 2014.

6.5. Algebraic λ-calculus

Ali Assaf, Alejandro Díaz-Caro, Simon Perdrix, Christine Tasson, and Benoit Valiron completed a journal paper covering results on different algebraic extensions of the λ-calculus [12]. These extensions equip the calculus with an additive and a scalar-multiplicative structure, and their set of terms is closed under linear combinations. Two such extensions, the *algebraic λ-calculus* and the *linear-algebraic λ-calculus* arise independently in different contexts – the former is a fragment of the differential λ-calculus, the latter is a candidate λ-calculus for quantum computation – and have different operational semantics. In this paper, the authors showed how the two approaches relate to each other. They showed that the the first calculus follows a call-by-name strategy while the second follows a call-by-value strategy. They proved that the two can simulate each other using algebraic extensions of *continuation passing style* (CPS) translations that are sound and complete.

ESTASYS Exploratory Action

6. New Results

6.1. Highlights of the Year

The Plasma statistical model checker has been made available to other scientists. ESTASYS has open a new branch on verifying the security of complex systems.

6.2. Verification of Heterogeneous Systems

Participants: Axel Legay, Benoît Boyer, Ngo Van-Chan, Jean Quilbeuf.

> This part concerns Tasks 1, 2 and 4 of the action. We characterize and formalize heterogeneous aspects of SoS and then we define efficient monitoring algorithms and representations for their requirements. We then combine the results with Statistical Model Checking (Task 5).

Systems of Systems (SoS) are very large scale systems with particular characteristics. SoS are not directly built from scratch by a single designer or a single team but are obtained as the composition of simpler systems. SoS have strong reliability and dependability requirements, as they aim to provide a service over a long running period. SoS may dynamically modify themselves by connecting to new systems, updating or disconnecting faulty ones, making it impossible to statically know the set of subsystems that are part of the SoS before runtime.

One of the main difficulty arising when developing SoS is the fact that subsystems may have been designed with a different goal in mind. In particular, some subsystems may have their own goal which differs from the global goal of the SoS. Furthermore, each subsystem may be developed in a particular computation model, making it difficult to find a common unifying semantics for the whole SoS. Finally, SoS may exhibit some emergent behaviors that are hardly predictable at design time.

One of the solutions to allow simulation of a SoS is to rely on a common interface for interconnecting the subsystems. The Functional Mockup Interface (FMI) standard is a natural candidate for such an interface. The different components of a SoS developed in different models of computation can be translated to Functional Mockup Units (FMU). Then a so-called master algorithm coordinates the FMUs composing the system. The execution of each FMU is either directly handled by the master algorithm or relies on an external tool for its execution.

Because the subsystems composing a SoS are of heterogeneous nature, it is difficult to find a common semantics model for the whole system. Furthermore, building such a transition system is not tractable due to the complexity of the system. Thus verification through traditional model checking is not possible for SoS. However, since the FMI/FMU framework enables simulation of such systems, the statistical model checking approach can be used.

The DANSE EU project aims to provide a complete tool chain from the modeling to the verification of SoS. At the higher level, the modeling is done in UPDM using the RHAPSODY tool. At the same level, the designer can express requirements over the model using some patterns written in GCSL. The UPDM model can then be translated into a FMI/FMU format that can be simulated by a dedicated tool, named DESYRE. Similarly, the GCSL requirements are transformed into BLTL formulas. Finally, the PLASMA statistical model checker has been integrated with the DESYRE tool chain in order to check the BLTL formulas based on the simulations provided by DESYRE.

6.2.1. Papers:

[45] (W) This report presents some of the results of the first year of Danse, one of the first EU IP projects dedicated to System of Systems. Concretely, we offer a tool chain that allows to specify SoS and SoS requirements at high level, and analyse them using powerful toolsets coming from the formal verification area. At the high level, we use UPDM, the system model provided by the british army as well as a new type of contract based on behavioral patterns. At low level, we rely on a powerful simulation toolset combined with recent advances from the area of statistical model checking. The approach has been applied to a case study developed at EADS Innovation Works.

[51] (W) Exhaustive formal verification for systems of systems (SoS) is impractical and cannot be applied on a large scale. In this paper we propose to use statistical model checking for efficient verification of SoS. We address three relevant aspects for systems of systems: 1) the model of the SoS, which includes stochastic aspects; 2) the formalization of the SoS requirements in the form of contracts; 3) the tool-chain to support statistical model checking for SoS. We adapt the SMC technique for application to heterogeneous SoS. We extend the UPDM/SysML specification language to express the SoS requirements that the implemented strategies over the SoS must satisfy. The requirements are specified with a new contract language specifically designed for SoS, targeting a high-level English-pattern language, but relying on an accurate semantics given by the standard temporal logics. The contracts are verified against the UPDM/SysML specification using the Statistical Model Checker (SMC) PLASMA combined with the simulation engine DESYRE, which integrates heterogeneous behavioral models through the functional mock-up interface (FMI) standard. The tool-chain allows computing an estimation of the satisfiability of the contracts by the SoS. The results help the system architect to trade-off different solutions to guide the evolution of the SoS.

6.3. Formal Models for Variability

Participants: Axel Legay, Rudolf Fahrenberg, Jin Hyun Kim.

> This part of the report is more concerned with task 2. It studies variability aspects in the broad scope. To simplify the study for the first year, we use the concept of software product lines. Later we shall use the results in federation of embedded systems, which is a particular class of Systems of systems.

Variability is ubiquitous in today's systems, be it in the form of configuration options or extensible architectures. By mastering variability, developers can adapt their system to changing requirements without having to develop entirely new applications. Variability is central in the context of SoS whose behaviors depend on interconnected objects. To gain information on managing variability, we have focused on Software Product Lines. Software Product Lines (SPLs) are a popular form of variability-intensive systems. They are families of similar software systems developed together to make economies of scale. SoS can be viewed as examples of product lines with interconnected obbjects. SPL engineering aims to facilitate the development of the members of a family (called *products* or *variants*) by identifying upfront their commonalities and differences. Variability in SPLs is commonly represented in terms of *features*, *i.e.*, units of difference between products that appear natural to stakeholders. Each product of an SPL is therefore defined by its set of features. Hierarchies of features and dependencies between features (*e.g.*, requires, excludes) are typically captured in a *Feature Model* (FM), *i.e.* a tree-like structure that specifies which combinations of features are valid.

6.3.1. Papers:

[15] (C) The model-checking problem for Software Products Lines (SPLs) is harder than for single systems: variability constitutes a new source of complexity that exacerbates the state-explosion problem. Abstraction techniques have successfully alleviated state explosion in single-system models. However, they need to be adapted to SPLs, to take into account the set of variants that produce a counterexample. In this paper, we apply CEGAR (Counterexample-Guided Abstraction Refinement) and we design new forms of abstraction specifically for SPLs. We carry out experiments to

evaluate the efficiency of our new abstractions. The results show that our abstractions, combined with an appropriate refinement strategy, hold the potential to achieve large reductions in verification time, although they sometimes perform worse. We discuss in which cases a given abstraction should be used.

•

[18] (C) In this work, We explore how ideas of statistical testing, based on a usage model (a Markov chain), can be used to extract configurations of interest according to the likelihood of their executions. These executions are gathered in featured transition systems, compact representation of SPL behaviour. We discuss possible scenarios and give a prioritization procedure validated on a web-based learning management software.

6.4. Statistical Model Checking

Participants: Axel Legay, Sean Sedwards, Benoît Boyer, Louis-Marie Traonouez, Kevin Corre.

This section covers Tasks 4 and 5 of the action. It consists in developping Simulation based techniques and efficient statistical algorithms for SoS.

The use of test cases remains the default means of ensuring the correct behaviour of systems in industry, but this technique is limited by the need to hypothesise scenarios that cause interesting behaviour and the fact that a reasonable set of test cases is unlikely to cover all possible eventualities. Static analysis is more thorough and has been successful in debugging very large systems, but its ability to analyse complex dynamical properties is limited. In contrast, model checking is an exhaustive technique that verifies whether a system satisfies a dynamical temporal logic property under all possible scenarios. For nondeterministic and probabilistic systems, numerical model checking quantifies the probability that a system satisfies a property. It can also be used to quantify the expected cost or reward of sets of executions.

Numerical model checking gives precise, accurate and certain results by exhaustively exploring the state space of the model, however the exponential growth of the state space with system size (the ‘state explosion problem) typically limits its applicability to “toy” systems. Symbolic model checking using efficient data structures can make certain very large models tractable. It may also be possible to construct simpler but behaviourally equivalent models using various symmetry reduction techniques, such as partial order reduction, bisimulation and lumping. If a new system is being constructed, it may be possible to guarantee the overall behaviour by verifying the behaviour of its subcomponents and limiting the way they interact. Despite these techniques, however, the size, unpredictability and heterogeneity of real systems usually make numerical techniques infeasible. Moreover, even if a system has been specified not to misbehave, it is nevertheless necessary to check that it meets its specification.

Simulation-based approaches are becoming increasingly tractable due to the availability of high performance parallel hardware and algorithms. In particular, statistical model checking (SMC) combines the simplicity of testing with the formality of numerical model checking. The core idea of SMC is to create multiple independent execution traces of a system and count how many satisfy a property specified in temporal logic. The proportion of satisfying traces is an estimate of the probability that the system satisfies the property. By thus modelling the executions of a system as a Bernoulli random variable, the absolute error of the estimate can be bounded using, for example, a confidence interval or a Chernoff bound. It is also possible to use efficient sequential hypothesis testing, to decide with specified statistical confidence whether the probability of a property is above or below a given threshold. Since SMC requires multiple independent simulations, it may be efficiently divided on parallel computer architectures, such as grids, clusters, clouds and general purpose computing on graphics processors (GPGPU).

Knowing a result with less than 100% confidence is often sufficient in real applications, since the confidence bounds may be made arbitrarily tight. Moreover, a swiftly achieved approximation may prevent a lot of wasted time during model design. For many complex systems, SMC offers the only feasible means of quantifying performance. Historically relevant SMC tools include APMC, YMER and VESTA. Well-established numerical model checkers, such as PRISM and UPPAAL, are now also including SMC engines. Dedicated SMC tools

under active development include COSMOS and our own tool PLASMA. Recognising that SMC may be applied to any discrete event trace obtained by stochastic simulation, we have devised PLASMA-lab, a modular library of SMC algorithms that may be used to construct domain-specific SMC tools. PLASMA-lab has become the main vehicle of our ongoing development of SMC algorithms.

Our group is devising cutting edge techniques for SMC. In particular, we are developing new learning algorithms (Sect. 6.4.3), algorithms for nondeterministic systems (Sect. 6.4.1), and algorithms for rare events (Sect. 6.4.2).

6.4.1. Algorithms for Nondeterminism

Nondeterministic models are of fundamental importance in defining complexity and are useful models of concurrency optimisation problems. This latter application is of particular importance in the context of systems constructed from subsystems ("Systems of Systems") that interact in an unpredictable way. Verifying or optimising such systems is problematic for numerical techniques because the state space is typically intractable. Nondeterminism is challenging for simulation-based techniques because, by definition, an executable semantics is not determined.

We have thus begun a line of research to develop SMC algorithms for nondeterministic systems. Our initial focus is Markov decision processes (MDP), however we are in the process of extending our work to various nondeterministc timed automata. Recent attempts to provide approximative algorithms for MDPs either do not address the standard verification problems, consider only a "spurious" subset of the standard problems or contain significant misconceptions and limitations.

In [28], we presented the first complete set of scalable SMC algorithms for MDPs. Our techniques are based on the idea of encoding a history-dependent scheduler as the seed of a pseudo-randomised hash function. Schedulers are thus chosen at random by selecting random seeds. The possibly infinite behaviour of the scheduler is completely encoded in $\mathcal{O}(1)$ memory. We presented simple sampling algorithms to find optimal schedulers and constructed the statistical confidence bounds necessary to find the optima of multiple estimates.

In [34] we devised the notion of "smart sampling" to dramatically improve the performance of the simple algorithms presented in [28]. The basic idea is to use part of the simulation budget to generate a crude estimate of the optimal scheduler and to use this information to better allocate the remaining budget. We successfully applied our algorithms to a number of standard case studies from the literature. We also highlighted the limitations of our approach.

The algorithms in [28], [34] find schedulers that minimise or maximise the probability of a property. In [37] we have adapted our algorithms to minimise or maximise the expected reward of a system. This adaptation is not entirely straightforward because the standard definition of reward properties assumes an exhaustive exploration of the state space of the MDP. We have included an implicit hypothesis test to include this assumption. In other respects optimising rewards is less challenging than optimising probabilities because rewards are effectively based on properties having probability 1. We demonstrate the accuracy of our rewards-based algorithms on standard case studies from the literature.

6.4.2. Rare Events in SMC

Rare properties are often highly relevant to system performance (e.g., bugs and system failure are required to be rare) but pose a problem for statistical model checking because they are difficult to observe. Fortunately, rare event techniques such as *importance sampling* and *importance splitting* may be successfully applied to statistical model checking.

In a previous work [50], we explicitly considered the use of importance sampling in the context of statistical model checking. We presented a simple algorithm that uses the notion of cross-entropy to find the optimal parameters for an importance sampling distribution. In contrast to previous work, our algorithm uses a low dimensional vector of parameters to define this distribution and thus avoids the often intractable explicit representation of a transition matrix. We showed that our parametrisation leads to a unique optimum and can produce many orders of magnitude improvement in simulation efficiency. We demonstrated the efficacy of our methodology by applying it to models from reliability engineering and biochemistry.

Our contribution [49] was the first attempt to use importance splitting with SMC to overcome the Rare Event problem. The basic idea is to decompose a logical property into nested properties whose probabilities are easier to estimate. Importance splitting achieves this by estimating a sequence of conditional probabilities, whose product is the required result. To apply this idea to model checking it is necessary to define a score function based on logical properties, and a set of levels that delimit the conditional probabilities. We described the necessary and desirable properties of score functions and levels. We illustrated how a score function may be derived from a property and gave two importance splitting algorithms: one that uses fixed levels and one that discovers optimal levels adaptively.

6.4.3. SMC with Changes and Simulink

We have proposed a new SMC algorithm for detecting probability changes in dynamic systems. We have adapted CUSUM, an algorithm that can be used to detect changes in signal monitoring. We show that CUSUM can be used to detect when the probability to satisfy a given property drops below some value. This algorithm offers new possibilities to detect, e.g., emergent behaviors in dynamic systems. Our main contributions has been to extend temporal logic with a change-based operator.

All these SMC algorithms are implemented in PLASMA-Lab, and have been recently exported to MATLAB/Simulink – a widely used environment for modeling, simulating and analyzing multidomain dynamic systems – through an integration of MATLAB/Simulink and PLASMA-lab. This integration exploit MATLAB Control, a library allowing to interact with MATLAB from Java. We have developed two different methods to link the two environments. The first method includes a new plugin for PLASMA-lab that allows to load and execute Simulink models within PLASMA-lab, and therefore apply SMC algorithms to these models. The second method proposes an application that can be launched directly within MATLAB and provide the PLASMA-Lab SMC algorithms.

We have submitted a paper [41] that presents the new CUSUM algorithm and the integration between PLASMA-Lab and Simulink. In this paper, we apply these results to a case-study developed with Simulink that models a temperature controller of a pig shed. We show how to use PLASMA-Lab to check SMC requirements, perform parameters optimisation and detect failures in the model using the new CUSUM algorithm.

6.4.4. Papers

[48] (C) Statistical model checking (SMC) offers the potential to decide and quantify dynamical properties of models with intractably large state space, opening up the possibility to verify the performance of complex real-world systems. Rare properties and long simulations pose a challenge to this approach, so here we present a fast and compact statistical model checking platform, PLASMA, that incorporates an efficient simulation engine and uses importance sampling to reduce the number and length of simulations when properties are rare. For increased flexibility and ef-ficiency PLASMA compiles both model and property into bytecode that is executed on an in-built memory-efficient virtual machine.

[47] (C) We present PLASMA-lab, a statistical model checking (SMC) library that provides the functionality to create custom statistical model checkers based on arbitrary discrete event modelling languages. PLASMA-lab is written in Java for maximum cross-platform compatibility and has already been incorporated in various performance-critical software and embedded hardware platforms. Users need only implement a few simple methods in a simulator class to take advantage of our efficient SMC algorithms. PLASMA-lab may be instantiated from the command line or from within other software. We have constructed a graphical user interface (GUI) that exposes the functionality of PLASMA-lab and facilitates its use as a standalone application with multiple 'drop-in' modelling languages. The GUI adds the notion of projects and experiments, and implements a simple, practical means of distributing simulations using remote clients.

[41] (C; submitted) Statistical Model Checking (SMC) is a powerful and widely used approach that consists in extracting global information on the system by monitoring some of its executions. In this paper, we add two new stones to the cathedral of results on SMC, that are 1. a new algorithm to detect emergent behaviors at runtime, and 2. an integration of Plasma Lab, a powerful SMC checker,

as a library of Simulink. Our results are illustrated on a realistic case study.

[26] (C) In this paper, we make use of the notion of a *score function* to improve the granularity of a logical property. We show that such a score function may take advantage of heuristics, so long as it also rigorously respects certain properties. To demonstrate our importance splitting approach we present an optimal adaptive importance splitting algorithm and an heuristic score function. We give experimental results that demonstrate a significant improvement in performance over alternative approaches.

[43] (C; submitted) We introduce feedback-control statistical system checking (FC-SSC), a new approach to statistical model checking that exploits principles of feedback-control for the analysis of cyber-physical systems (CPS). FC-SSC uses stochastic system identification to learn a CPS model, importance sampling to estimate the CPS state, and importance splitting to control the CPS so that the probability that the CPS satisfies a given property can be efficiently inferred. We illustrate the utility of FC-SSC on two example applications, each of which is simple enough to be easily understood, yet complex enough to exhibit all of FC-SCC's features. To the best of our knowledge, FC-SSC is the first statistical system checker to efficiently estimate the probability of rare events in realistic CPS applications or in any complex probabilistic program whose model is either not available, or is infeasible to derive through static-analysis techniques.

6.5. Quantitative Reasoning

Participants: Axel Legay, Rudolf Fahrenberg, Louis-Marie Traonouez.

This part is concerned with Tasks 1 and 2. Mostly, we focus on quantifying properties of interconnected objects such as CPS (SoS and CPS share a lot of commonalities).

Model checking of systems deals with the question whether a given model of a computer system satisfies the properties one might want to require of it. This is a well-established and successful approach to formal verification of safety-critical computer systems.

When the models of the systems contain quantitative information, which is needed to represent the material on which the SoS is running, the model checking problem becomes complicated by the fact that in most cases, quantitative properties of the systems do not need to be satisfied exactly. Indeed, the model or the properties might be subject to measurement error, or probabilistic information might only be an approximation. In this case, it is of little use to know whether or not a model satisfies a specification precisely; what is needed instead is a notion of *satisfaction distance*: a measure which can assess to which extent a quantitative model satisfies a quantitative specification.

In other words, what is needed is a notion of satisfaction which is robust in the sense that small deviations in the model or the specification only lead to small changes in the outcome of the model checking question.

For reasoning about distributed systems or **systems-of-systems**, an important role is played by specification theories. Such systems are often far too complex to reason about, or model-check, as a whole, and additionally they might be composed of a large number of components which are implemented by different vendors. Hence one needs methods for compositional reasoning, which allow to infer properties of a system from properties of its components, and for incremental design, which allow to synthesize and refine specifications in a step-wise manner.

Such specification theories are by now well-established e.g. in the incarnations of interface theories and (disjunctive) modal transition systems. Additionally to defining a formalism for describing and model-checking specifications, they provide notions of refinement of specifications, logical conjunction of specifications, and structural composition and quotient.

When the models and specifications contain quantitative information, all the above notions need to be made robust. One needs to introduce a quantitative version of refinement, and the operations on specifications need to be continuous with respect to refinement distance: compositions of specifications with small refinement distance need themselves to have small refinement distance.

6.5.1. Theory papers:

[33] (J; submitted) There are two fundamentally different approaches to specifying and verifying properties of systems. The logical approach makes use of specifications given as formulae of temporal or modal logics and relies on efficient model checking algorithms; the behavioural approach exploits various equivalence or refinement checking methods, provided the specifications are given in the same formalism as implementations. In this paper we provide translations between the logical formalism of nu-calculus and the behavioural formalism of disjunctive modal transition systems. The translations preserve structural properties of the specification and allow us to perform logical operations on the behavioural specifications as well as behavioural compositions on logical formulae. The unification of both approaches provides additional methods for component-based stepwise design.

[4] (C) This paper studies a difference operator for stochastic systems whose specifications are represented by Abstract Probabilistic Automata (APAs). In the case refinement fails between two specifications, the target of this operator is to produce a specification APA that represents all witness PAs of this failure. Our contribution is an algorithm that allows to approximate the difference of two APAs with arbitrary precision. Our technique relies on new quantitative notions of distances between APAs used to assess convergence of the approximations, as well as on an in-depth inspection of the refinement relation for APAs. The procedure is effective and not more complex to implement than refinement checking.

[21] (C) We provide a framework for compositional and iterative design and verification of systems with quantitative information, such as rewards, time or energy. It is based on disjunctive modal transition systems where we allow actions to bear various types of quantitative information. Throughout the design process the actions can be further refined and the information made more precise. We show how to compute the results of standard operations on the systems, including the quotient (residual), which has not been previously considered for quantitative non-deterministic systems. Our quantitative framework has close connections to the modal nu-calculus and is compositional with respect to general notions of distances between systems and the standard operations.

[35] (J; submitted) We provide a framework for compositional and iterative design and verification of systems with quantitative information, such as rewards, time or energy. It is based on disjunctive modal transition systems where we allow actions to bear various types of quantitative information. Throughout the design process the actions can be further refined and the information made more precise. We show how to compute the results of standard operations on the systems, including the quotient (residual), which has not been previously considered for quantitative non-deterministic systems. Our quantitative framework has close connections to the modal nu-calculus and is compositional with respect to general notions of distances between systems and the standard operations.

[6] (J) This paper proposes a new theory of quantitative specifications. It generalizes the notions of stepwise refinement and compositional design operations from the Boolean to an arbitrary quantitative setting. Using a great number of examples, it is shown that this general approach permits to unify many interesting quantitative approaches to system design.

[7] (J) We present a distance-agnostic approach to quantitative verification. Taking as input an unspecified distance on system traces, or executions, we develop a game-based framework which allows us to define a spectrum of different interesting system distances corresponding to the given trace distance. Thus we extend the classic linear-time–branching-time spectrum to a quantitative setting, parametrized by trace distance. We also prove a general transfer principle which allows us to transfer counterexamples from the qualitative to the quantitative setting, showing that all system distances are mutually topologically inequivalent.

[25] (C) We introduce a new notion of structural refinement, a sound abstraction of logical implication, for the modal nu-calculus. Using new translations between the modal nu-calculus and disjunctive modal transition systems, we show that these two specification formalisms are structurally equivalent.Using our translations, we also transfer the structural operations of composition and quotient from disjunctive modal transition systems to the modal nu-calculus. This shows that the modal nu-calculus supports composition and decomposition of specifications.

6.5.2. Application papers:

[32] (C; submitted) We suggest a method for measuring the degree to which features interact in feature-oriented software development. We argue that our method is practically feasible, easily extendable and useful from a developer's point of view.

[19] (C) Class diagrams are among the most popular modeling languages in industrial use. In a model-driven development process, class diagrams evolve, so it is important to be able to assess differences between revisions, as well as to propagate differences using suitable merge operations. Existing differencing and merging methods are mainly syntactic, concentrating on edit operations applied to model elements, or they are based on sampling: enumerating some examples of instances which characterize the difference between two diagrams. This paper presents the first known (to the best of our knowledge) automatic model merging and differencing operators supported by a formal semantic theory guaranteeing that they are semantically sound. All instances of the merge of a model and its difference with another model are automatically instances of the second model. The differences we synthesize are represented using class diagram notation (not edits, or instances), which allows creation of a simple yet flexible algebra for diffing and merging. It also allows presenting changes comprehensively, in a notation already known to users.

[20] (C) We propose a new similarity measure between texts which, contrary to the current state-of-the-art approaches, takes a global view of the texts to be compared. We have implemented a tool to compute our textual distance and conducted experiments on several corpuses of texts. The experiments show that our methods can reliably identify different global types of texts.

[23] (C) Reliable model transformations are essential for agile modeling. We propose to employ a configurable-semantics approach to develop automatic model transformations which are correct by design and can be integrated smoothly into existing tools and work flows.

[39] (C; submitted) Nowadays, large software systems are mostly built using existing services. These are not always designed to interact, i.e., their public interfaces often present some mismatches. Checking compatibility of service interfaces allows one to avoid erroneous executions when composing the services and ensures correct reuse and interaction. Service compatibility has been intensively studied, in particular for discovery purposes, but most of existing approaches return a Boolean result. In this paper, we present a quantitative approach for measuring the compatibility degree of service interfaces. Our method is generic and flooding-based, and fully automated by a prototype tool.

6.5.3. Surveys:

[22] Modal transition systems provide a behavioral and compositional specification formalism for reactive systems. We survey two extensions of modal transition systems: parametric modal transition systems for specifications with parameters, and weighted modal transition systems for quantitative specifications.

[24] We survey extensions of modal transition systems to specification theories for probabilistic and timed systems.

6.6. Privacy and Security

Participants: Axel Legay, Fabrizio Biondi, Jean Quilbeuf, Thomas Given-Wilson.

6.6.1. Information-Theoretical Quantification of Security Properties

This part of the work was not foreseen at the beginning of the action. It concerns security aspects, and more precisely quantifying privacy of data. This aspect is in fact central for SoS and all our algorithms developed for Tasks 4 and 5 should be adapted to solve a series of problems linked to privacy in interconnected object and dynamical environment. For now, we only studied the foundations.

Information theory provides a powerful quantitative approach to measuring security and privacy properties of systems. By measuring the *information leakage* of a system security properties can be quantified, validated, or falsified. When security concerns are non-binary, information theoretic measures can quantify exactly how much information is leaked. The knowledge of such informations is strategic in the developments of component-based systems.

The quantitative information-theoretical approach to security models the correlation between the secret information of the system and the output that the system produces. Such output can be observed by the attacker, and the attacker tries to infer the value of the secret by combining this information with its knowledge of the system.

Armed with the produced output and the source code of the system, the attacker tries to infer the value of the secret. The quantitative analysis we implement computes with arbitrary precision the number of bits of the secret that the attacker will expectedly infer. This expected number of bits is the information leakage of the system.

The quantitative approach generalizes the qualitative approach and thus provides superior analysis. In particular, a system respects non-interference if and only if its leakage is equal to zero. In practice very few systems respect non-interference, and for those who don't it is imperative to be able to distinguish between the ones leaking a very small amount of bits and the ones leaking a significant amount of bits, since only the latter are considered to pose a security vulnerability to the system.

Since black box security analyzes are immediately invalidated whenever an attacker gains information about the source code of the system, we assume that the attacker has a white box view of the system, meaning that it has access to the system's source code. This approach is also consistent with the fact that many security protocol implementations are in fact open source.

The scope of modern software projects is too large to be analyzed manually. For this reason we provide tools that can support the analyst and locate security vulnerabilities in large codebases and projects. We work with a variety of tools, including commercial software analysis tools being adapted with our techniques, and tools such as QUAIL developed here by our team.

We applied the leakage analysis provided by QUAIL to several case studies. Our case studies (voting protocol and smart grid coordination) have in common that a publicly disclosed information is computed from the secret of every participant in the model. In the voting example, the vote of a given voter is secret, but the number of votes for each candidates is public. Similarly, in the smart grid example, the consumption of one of the houses is secret, but the consumption of a whole quarter can be deduced. Qualitative analyses are either too restrictive or too permissive on these types of systems. For instance, non-interference will reject them as the public information depends on the secret. Declassification approaches will accept them, even if the number of voters or consumers is 2, in which case the secret can be deduced.

The development of better tools for quantitative security builds upon both theoretical developments in information theory, and development of the tools themselves. These often progress in parallel with each supporting the findings of the other, and increasing the demands and understanding upon each other.

6.6.1.1. Papers:

[3] (J; submitted) The quantification of information leakage provides a quantitative evaluation of the security of a system. We propose the usage of Markovian processes to model deterministic and probabilistic systems. By using a methodology generalizing the lattice of information approach we model refined attackers capable to observe the internal behavior of the system, and quantify the information leakage of such systems. We also use our method to obtain an algorithm for the computation of channel capacity from our Markovian models. Finally, we show how to use the method to analyze timed and non-timed attacks on the Onion Routing protocol.

[46] (C) Quantitative security analysis evaluates and compares how effectively a system protects its secret data. We introduce QUAIL, the first tool able to perform an arbitrary-precision quantitative analysis of the security of a system depending on private information. QUAIL builds a Markov Chain model of the system's behavior as observed by an attacker, and computes the correlation between

the system's observable output and the behavior depending on the private information, obtaining the expected amount of bits of the secret that the attacker will infer by observing the system. QUAIL is able to evaluate the safety of randomized protocols depending on secret data, allowing to verify a security protocol's effectiveness. We experiment with a few examples and show that QUAIL's security analysis is more accurate and revealing than results of other tools.

[40] (C; submitted) Quantitative security techniques have been proven effective to measure the security of systems against various types of attackers. However, such techniques are based on computing exponentially large channel matrices or Markov chains, making them impractical for large programs. We propose a different approach based on abstract trace analysis. By analyzing directly sets of execution traces of the program and computing security measures on the results, we are able to scale down the exponential cost of the problem. Also, we are able to appy statistical simulation techniques, allowing us to obtain significant results even without exploring the full space of traces. We have implemented the resulting algorithms in the QUAIL tool. We compare their effectiveness against the state of the art LeakWatch tool on two case studies: privacy of user consumption in smart grid systems and anonymity of voters in different voting schemes.

[12] (C) In an election, it is imperative that the vote of the single voters remain anonymous and undisclosed. Alas, modern anonymity approaches acknowledge that there is an unavoidable leak of anonymity just by publishing data related to the secret, like the election's result. Information theory is applied to quantify this leak and ascertain that it remains below an acceptable threshold. We apply modern quantitative anonymity analysis techniques via the state-of-the-art QUAIL tool to the voting scenario. We consider different voting typologies and establish which are more effective in protecting the voter's privacy. We further demonstrate the effectiveness of the protocols in protecting the privacy of the single voters, deriving an important desirable property of protocols depending on composite secrets.

[13] (C) In recent years, quantitative security techniques have been providing effective measures of the security of a system against an attacker. Such techniques usually assume that the system produces a finite amount of observations based on a finite amount of secret bits and terminates, and the attack is based on these observations. By modeling systems with Markov chains, we are able to measure the effectiveness of attacks on non-terminating systems. Such systems do not necessarily produce a finite amount of output and are not necessarily based on a finite amount of secret bits. We provide characterizations and algorithms to define meaningful measures of security for non-terminating systems, and to compute them when possible. We also study the bounded versions of the problems, and show examples of non-terminating programs and how their effectiveness in protecting their secret can be measured.

GALLIUM Project-Team

6. New Results

6.1. Formal verification of compilers and static analyzers

6.1.1. Formal verification of static analyzers based on abstract interpretation

Participants: Jacques-Henri Jourdan, Xavier Leroy, Sandrine Blazy [EPI Celtique], Vincent Laporte [EPI Celtique], David Pichardie [EPI Celtique], Sylvain Boulmé [Grenoble INP, VERIMAG], Alexis Fouilhe [Université Joseph Fourier de Grenoble, VERIMAG], Michaël Périn [Université Joseph Fourier de Grenoble, VERIMAG].

In the context of the ANR Verasco project, we are investigating the formal specification and verification in Coq of a realistic static analyzer based on abstract interpretation. This static analyzer handles a large subset of the C language (the same subset as the CompCert compiler, minus recursion and dynamic allocation); supports a combination of abstract domains, including relational domains; and should produce usable alarms. The long-term goal is to obtain a static analyzer that can be used to prove safety properties of real-world embedded C codes.

This year, Jacques-Henri Jourdan continued the developpment of this static analyzer. He finished the proof of correctness of the abstract interpreter, using an axiomatic semantics for the C#minor intermediate language to decompose this proof in two manageable halves. He improved the precision and performance of the abstract iterator and of numerical abstract domains. He designed and verified a symbolic domain that helps analyzing sequential Boolean operators such as `&&` and `||` that are encoded as Boolean variables and conditional constructs in the C#minor intermediate language. As a more flexible alternative to reduced products of domains, Jacques-Henri Jourdan designed, implemented and proved correct a communication system between numerical abstract domains, based on communication channels and inspired by Astrée [56].

In parallel, IRISA and VERIMAG, our academic partners on the Verasco project, contributed a verified abstract domain for memory states and pointer values (Vincent Laporte, Sandrine Blazy, and David Pichardie) and a polyhedric abstract domain for linear numerical inequalities (Alexis Fouilhe, Sylvain Boulmé, Michaël Périn) that uses validation a posteriori. Those various components were brought together by Jacques-Henri Jourdan and Vincent Laporte, resulting in an executable static analyzer.

The overall architecture and specification of Verasco is described in a paper [29] accepted for presentation at the forthcoming POPL 2015 conference.

6.1.2. The CompCert formally-verified compiler

Participants: Xavier Leroy, Jacques-Henri Jourdan.

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.

This year, we improved the CompCert C compiler in several directions:

- The parser, previously compiled to unverified OCaml code, was replaced by a parser compiled to Coq code and validated *a posteriori* by a validator written and proved sound in Coq. This validation step, performed when the CompCert compiler is compiled, provides a formal proof that the parser recognizes exactly the language described by the source grammar. This approach builds on the earlier work by Jacques-Henri Jourdan, François Pottier and Xavier Leroy on verified validation of $LR(1)$ parsers [60]. Jacques-Henri Jourdan succeeded in scaling this approach all the way up to the full ISO C99 grammar plus some extensions.
- Two new static analyses, value analysis and neededness analysis, were added to the CompCert back-end. As described in section 6.1.3 below, the results of these analyses enable more aggressive optimizations over the RTL intermediate form.
- As part of the work on formalizing floating-point arithmetic (see section 6.1.4 below), the semantics and compilation of floating-point arithmetic in CompCert was revised to handle single-precision floating-point numbers as first-class values, instead of systematically converting them to double precision before arithmetic. This increases the efficiency and compactness of the code generated for applications that make heavy use of single precision.
- Previously, the CompCert back-end compiler was assuming a partitioned register set from the target architecture, where integer registers always contain 32-bit integers or pointers, and floating-point registers always contain double-precision FP numbers. This convention on register uses simplified the verification of CompCert, but became untenable with the introduction of single-precision FP numbers as first-class values: FP registers can now hold either single- or double-precision FP numbers. Xavier Leroy rearchitected the register allocator and the stack materialization passes of CompCert, along with their soundness proofs, to lift this limitation on register uses. Besides mixtures of single- and double-precision FP numbers, this new architecture makes it possible to support future target processors with a unified register set, such as the SPE variant of PowerPC.
- We added support for several features of ISO C99 that were not handled previously: designated initializers, compound literals, `switch` statements where the `default` case is not the last case, `switch` statements over arguments of 64-bit integer type, and incomplete arrays as the last member of a `struct`. Also, variable-argument functions and the `<stdarg.h>` standard include are now optionally supported, but their implementation is neither specified nor verified.
- The ARM back-end was extended with support for the EABI-HF calling conventions (passing FP arguments and results in FP registers instead of integer registers) and with generation of Thumb2 instructions. Thumb2 is an alternate instruction set and instruction encoding for the ARM architecture that results in more compact machine code (up to 30% reduction in code size on our tests).

We released three versions of CompCert, integrating these enhancements: version 2.2 in February 2014, version 2.3 in April, and version 2.4 in September.

In June 2014, Inria signed a licence agreement with AbsInt Angewandte Informatik GmbH, a software publisher based in Saarbrucken, Germany, to market and provide support for the CompCert formally-verified C compiler. AbsInt will extend CompCert to improve its usability in the critical embedded software market, and also provide long-term maintenance as required in this market.

6.1.3. Value analysis and neededness analysis in CompCert

Participant: Xavier Leroy.

Xavier Leroy designed, implemented, and proved sound two new static analyses over the RTL intermediate representation of CompCert. Both analyses are of the intraprocedural dataflow kind.

- Value analysis is a forward analysis that tracks points-to information for pointers, constantness information for integer and FP numbers, and variation intervals for integer numbers, using intervals of the form $[0, 2^n)$ and $[-2^n, 2^n)$. This value analysis extends and generalizes CompCert's earlier

constant analysis as well as the points-to analysis of Robert and Leroy [68]. In particular, it tracks both the values of variables and the contents of memory locations, and it can take advantage of points-to information to show that function-local memory does not escape the scope of the function.

- Neededness analysis is a backward analysis that tracks which memory locations and which bits of the values of integer variables may be used later in a function, and which memory locations and integer bits are "dead", i.e. never used later. This analysis extends CompCert's earlier liveness analysis to memory locations and to individual bits of integer values.

Compared with the static analyses developed as part of Verasco (section 6.1.1), value analysis is much less precise: every function is analyzed independently of its call sites, relations between variables are not tracked, and even interval analysis is coarser (owing to CompCert's lack of support for widened fixpoint iteration). However, CompCert's static analyses are much cheaper than Verasco's, and scale well to large source codes, making it possible to perform them at every compilation run.

Xavier Leroy then modified CompCert's back-end optimizations to take advantage of the results of the two new static analyses, thus improving performance of the generated code:

- Common subexpression elimination (CSE) takes advantage of non-aliasing information provided by value analysis to eliminate redundant memory loads more aggressively.
- Many more integer casts (type conversions) and bit masking operations are discovered to be redundant and eliminated.
- Memory stores and block copy operations that become useless after constant propagation and CSE can now be eliminated entirely.

6.1.4. *Verified compilation of floating-point arithmetic*

Participants: Sylvie Boldo [EPI Toccata], Jacques-Henri Jourdan, Xavier Leroy, Guillaume Melquiond [EPI Toccata].

In 2012, we replaced the axiomatization of floating-point numbers and arithmetic operations used in early versions of CompCert by a fully-formal Coq development, building on the Coq formalization of IEEE-754 arithmetic provided by the Flocq library of Sylvie Boldo and Guillaume Melquiond. This verification of FP arithmetic and of its compilation was further improved in 2013 with respect to the treatment of "Not a Number" special values.

This year, Guillaume Melquiond improved the algorithmic efficiency of some of the executable FP operations provided by Flocq. Xavier Leroy generalized the theorems over FP arithmetic used in CompCert's soundness proof so that these theorems apply both to single- and double-precision FP numbers. Jacques-Henri Jourdan and Xavier Leroy proved additional theorems concerning conversions between integers and FP numbers.

A journal paper describing this 3-year work on correct compilation of floating-point arithmetic was accepted for publication at Journal of Automated Reasoning [14].

6.1.5. *Verified JIT compilation of Coq*

Participants: Maxime Dénès, Xavier Leroy.

Evaluation of terms from Gallina, the functional language embedded within Coq, plays a crucial role in the performance of proof checking or execution of verified programs, and the trust one can put in them. Today, Coq provides various evaluation mechanisms, some internal, in the kernel, others external, via extraction to OCaml or Haskell. However, we believe that the specific performance trade-offs and the delicate issues of trust are still calling for a better, more adapted, treatment.

That is why we started in October this year the Coqonut project, whose objective is to develop and formally verify an efficient, compiled implementation of Coq reductions. As a first step, we wrote an unverified prototype in OCaml producing x86-64 machine code using a monadic intermediate form. We started to port it to Coq and to specify the semantics of the source, target and intermediate languages.

6.2. Language design and type systems

6.2.1. The Mezzo programming language

Participants: Thibaut Balabonski, François Pottier, Jonathan Protzenko.

Mezzo is a programming language proposal whose untyped foundation is very much like OCaml (i.e., it is equipped with higher-order functions, algebraic data structures, mutable state, and shared-memory concurrency) and whose type system offers flexible means of describing ownership policies and controlling side effects.

In 2013 and early 2014, Thibaut Balabonski and François Pottier re-worked the machine-checked proof of type soundness for Mezzo. They developed a version of the proof which includes concurrency and dynamically-allocated locks, and showed that well-typed programs do not crash and are data-race free. This work was presented by François Pottier at FLOPS 2014 [24]. The proof was then extended with a novel and simplified account of adoption and abandon, a mechanism for combining the static ownership discipline with runtime ownership tests. A comprehensive paper, which contains both a tutorial introduction to Mezzo and a description of its formal definition and proof, was submitted to TOPLAS.

Minor modifications were carried out by Jonathan Protzenko in the implementation. A version of Mezzo that runs in a Web browser was developed and uploaded online, so that curious readers can play with the language without installing the software locally.

Jonathan Protzenko wrote his Ph.D. dissertation [12], which describes the design of Mezzo and the implementation of the Mezzo type-checker. He defended on September 29, 2014.

Web site: http://protz.github.io/mezzo/

6.2.2. System F with coercion constraints

Participants: Julien Cretin [Trust In Soft], Didier Rémy, Gabriel Scherer.

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 study of such systems is to view type conversions as *coercions* inside terms.

Following a general approach based on System F, Julien Cretin and Didier Rémy earlier introduced a language of *explicit coercions* enabling abstraction over coercions and viewing all type transformations as explicit coercions [57]. To ensure that coercions are erasable, *i.e.*, that they only decorate terms without altering their reduction, they are restricted to those that are parametric in either their domain or codomain. Despite this restriction, this language already subsumed many extensions of System F, including bounded polymorphism, instance-bounded polymorphism, and η-conversions—but not subtyping constraints.

To lift this restriction, Julien Crétin and Didier Rémy proposed a new approach were coercions are left implicit. Technically, we extended System F with a rich language of propositions containing a first-order logic, a coinduction mechanism, consistency assertions, and coercions (which are thus just a particular form of propositions); we then introduced a type-level language using kinds to classify types, and constrained kinds to restrict kinds to types satisfying a proposition. Abstraction over types of a constrained kind amounts to abstraction over arbitrary propositions, including coercions.

By default, type abstraction should be erasable, which is the case when kinds of abstract type variables are inhabited—we say that such abstractions are consistent. Still, inconsistent type abstractions are also useful, for instance, to model GADTs. We provide them as a different construct, since they are not erasable, as they must delay reduction of subterms that depend on them. This introduces a form of weak reduction in a language with full reduction, which is a known source of difficulties: although the language remains sound, we loose the subject reduction property. This work has been described in [28] and is part of Julien Cretin's PhD dissertation [11] defended in January 2014; a simpler, core subset is also described in [45].

Recently, Gabriel Scherer and Didier Rémy introduced *assumption hiding* [32], [50] to restore confluence when mixing full and weak reductions and provide a continuum between consistent and inconsistent abstraction. Assumption hiding allows a fine-grained control of dependencies between computations and the logical hypotheses they depend on; although studied for a language of coercions, the solution is more general and should be applicable to any language with abstraction over propositions that are left implicit, either for the user's convenience in a surface language or because they have been erased prior to computation in an internal language.

6.2.3. Singleton types for code inference

Participants: Gabriel Scherer, Didier Rémy.

We continued working on 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. A decision algorithm for the simply-typed lambda-calculus is still work-in-progress. We presented at the TYPES'14 conference [42] our general approach to such decision procedures, and obtained an independent counting result for intuitionistic logic [52] that demonstrates the finiteness of the search space.

6.2.4. Generic programming with ornaments

Participants: Pierre-Évariste Dagand, Didier Rémy, Thomas Williams.

Since their first introduction in ML, datatypes have evolved: besides providing an organizing *structure* for computation, they are now offering more *control* over what is a valid result. GADTs, which are now part of the OCaml language, offer such a mechanism: ML programmers can express fine-grained, logical invariants of their datastructures. Programmers thus strive to express the correctness of their programs in the types: a well-typed program is correct by construction. However, these carefully crafted datatypes are a threat to any library design: the same data-*structure* is used for many logically incompatible purposes. To address this issue, McBride developed *ornaments*. It defines conditions under which a new datatype definition can be described as an ornament of another one, typically when they both share the same inductive definition scheme. For example, lists can be described as the ornament of the Church encoding of natural numbers. Once a close correspondence between a datatype and its ornament has been established, certain kinds of operations on the original datatype can be automatically lifted to its ornament.

To account for whole-program transformations, we developed a type-theoretic presentation of *functional ornament* [17] as a generalization of ornaments to functions. This work built up on a type-theoretic *universe of datatypes*, a first-class description of inductive types within the type theory itself. Such a presentation allowed us to analyze and compute over datatypes in a transparent manner. Upon this foundation, we formalized the concept of functional ornament by another type-theoretic universe construction. Based on this universe, we established the connection between a base function (such as addition and subtraction) and its ornamented version (such as, respectively, the concatenation of lists and the deletion of a prefix). We also provided support for driving the computer into semi-automatically lifting programs: we showed how addition over natural numbers could be incrementally evolved into concatenation of lists.

Besides the theoretical aspects, we have also tackled the practical question of offering ornaments in an ML setting [33]. Our goal was to extend core ML with support for ornaments so as to enable semi-automatic program transformation and fully-automatic code refactoring. We thus treated the purely syntactic aspects, providing a concrete syntax for describing ornaments of datatypes and specifying the lifting of functions. Such lifting specifications allow the user to declaratively instruct the system to, for example, lift addition of numbers to concatenation of lists. We gave an algorithm that, given a lifting specification, performs the actual program transformation from the bare types to the desired, ornamented types. This work has been evaluated by a prototype implementation in which we demonstrated a few typical use-cases for the semi-automatic lifting of programs.

Having demonstrated the benefits of ornaments in ML, it has been tempting to offer ornaments as first-class citizens in a programming language. Doing so, we wished to rationalize the lifting of programs as an elaboration process within a well-defined, formal system. To describe the liftings, one would like to specify only the local transformations that are applied to the original program. Designing such a language of *patches* and formalizing its elaboration has been the focus of our recent efforts.

6.2.5. Constraints as computations

Participant: François Pottier.

Hindley-Milner type inference–the problem of determining whether an ML program is well-typed–is well-understood, and can be elegantly explained and implemented in two phases, namely constraint generation and constraint solving. In contrast, elaboration–the task of constructing an explicitly-typed representation of the program–seems to have received relatively little attention in the literature, and did not until now enjoy a modular constraint-based presentation. François Pottier proposed such a presentation, which views constraints as computations and equips them with the structure of an applicative functor. This work was presented as a "functional pearl" at ICFP 2014 [31]. The code, in the form of a re-usable library, is available online.

6.2.6. Equivalence and normalization of lambda-terms with sums

Participants: Gabriel Scherer, Guillaume Munch-Maccagnoni [Université 13, LIPN lab].

Determining uniqueness of inhabitants requires a good understanding of program equivalence in presence of sum types. In yet-unpublished work, Gabriel Scherer worked on the correspondence between two existing normalization techniques, one coming from the focusing community [54] and the other using direct lambda-term rewriting [63]. A collaboration with Guillaume Munch-Maccagnoni has also started this year, whose purpose is to present normalization procedures for sums using System L, a rich, untyped syntax of terms (or abstract machines) for the sequent calculus.

6.2.7. Computational interpretation of realizability

Participants: Pierre-Évariste Dagand, Gabriel Scherer.

We are trying to better understand the computational behavior of semantic normalization techniques such as a realizability and logical relation models. As a very first step, we inspected the computational meaning of a normalization proof by realizability for the simply-typed lambda-calculus. It corresponds to an evaluation function; the evaluation order for each logical connective is determined by the definition of the sets of truth and value witnesses. This preliminary work is to be presented at JFLA 2015 [35].

6.3. Shared-memory parallelism

6.3.1. Algorithms and data structures for parallel computing

Participants: Umut Acar, Arthur Charguéraud [EPI Toccata], Mike Rainey.

The ERC Deepsea project, with principal investigator Umut Acar, started in June 2013 and is hosted by the Gallium team. This project aims at developing techniques for parallel and self-adjusting computations in the context of shared-memory multiprocessors (i.e., multicore platforms). The project is continuing work that began at Max Planck Institute for Software Systems between 2010 and 2013. As part of this project, we are developing a C++ library, called PASL, for programming parallel computations at a high level of abstraction. We use this library to evaluate new algorithms and data structures. We obtained two major results this year.

The first result is a sequence data structure that provides amortized constant-time access at the two ends, and logarithmic time concatenation and splitting at arbitrary positions. These operations are essential for programming efficient computation in the fork-join model. Compared with prior work, this novel sequence data structure achieves excellent constant factors, allowing it to be used as a replacement for traditional, non-splittable sequence data structures. This data structure, called *chunked sequence* due to its use of chunks (fixed-capacity arrays), has been implemented both in C++ and in OCaml, and shown competitive with state-of-the art sequence data structures that do not support split and concatenation operations. This work is described in a paper published at ESA [22].

A second main result is the development of fast and robust parallel graph traversal algorithms, more precisely for parallel BFS and parallel DFS. The new algorithms leverage the aformentioned sequence data structure for representing the set of edges remaining to be visited. In particular, it uses the split operation for balancing the edges among the several processors involved in the computation. Compared with prior work, these new algorithms are designed to be efficient not just for particular classes of graphs, but for all input graphs. This work has not yet been published, however it is described in details in a technical report [46].

6.3.2. Weak memory models

Participants: Luc Maranget, Jacques-Pascal Deplaix, Jade Alglave [University College London, then Microsoft Research, Cambridge].

Modern multi-core and multi-processor computers do not follow the intuitive "Sequential Consistency" model that would define a concurrent execution as the interleaving of the execution of its constituting threads and that would command instantaneous writes to the shared memory. This situation is due both to in-core optimisations such as speculative and out-of-order execution of instructions, and to the presence of sophisticated (and cooperating) caching devices between processors and memory.

In the last few years, Luc Maranget took part in an international research effort to define the semantics of the computers of the multi-core era. This research effort relies both on formal methods for defining the models and on intensive experiments for validating the models. Joint work with, amongst others, Jade Alglave (now at Microsoft Research, Cambridge), Peter Sewell (University of Cambridge) and Susmit Sarkar (University of St. Andrews) achieved several significant results, including two semantics for the IBM Power and ARM memory models: one of the operational kind [70] and the other of the axiomatic kind [64]. In particular, Luc Maranget is the main developer of the **diy** tool suite (see section 5.3). Luc Maranget also performs most of the experiments involved.

In 2014 we produced a new model for Power/ARM. The new model is simpler than the previous ones, in the sense that it is based on fewer mathematical objects and can be simulated more efficiently than the previous models. The new **herd** simulator (part of **diy** tool suite) is in fact a generic simulator, whose central component is an interpreter for a domain-specific language. More precisely, memory models are described in a simple language that defines relations by means of a few operators such as concatenation, transitive closure, fixpoint, etc., and performs validity checks on relations such as acyclicity. The Power/ARM model consists of about 50 lines of this specific language. This work, with additional material, including in-depth testing of ARM devices and data-mining of potential concurrency bugs in a huge code base, was published in the journal *Transaction on Programming Languages and Systems* [13] and selected for presentation at the PLDI conference [23]. Luc Maranget gave this presentation.

In the same research theme, Luc Maranget supervised the internship of Jacques-Pascal Deplaix (EPITECH), from Oct. 2013 to May 2014. Jacques-Pascal extended **litmus**, our tool to run tests on hardware. **litmus** now accepts test written in C; we can now perform the conformance testing of C compilers and machines with respect to the C11/C++11 standard. Namely, Mark Batty (University of Cambridge), under the supervision of Jade Alglave, wrote a **herd** model for this standard. The new **litmus** also proves useful to run tests that exploit some machine idiosyncrasies, when our **litmus** assembly implementation does not handle them.

As a part of the **litmus** infrastructure, Luc Maranget designed a synchronisation barrier primitive by simplifying the sense synchronisation barrier published by Maurice Herlily and Nir Shavit in their textbook [58]. He co-authored a JFLA article [34], that presents this primitive and proves it correct automatically by the means of the **cubicle** tool developed under the supervision of Sylvain Conchon (team Toccata, Inria Saclay).

6.4. The OCaml language and system

6.4.1. The OCaml system

Participants: Damien Doligez, Alain Frisch [Lexifi SAS], Jacques Garrigue [University of Nagoya], Fabrice Le Fessant, Xavier Leroy, Luc Maranget, Gabriel Scherer, Mark Shinwell [Jane Street], Leo White [OCaml Labs, Cambridge University], Jeremy Yallop [OCaml Labs, Cambridge University].

This year, we released versions 4.02.0 and 4.02.1 of the OCaml system. Release 4.02.0 is a major release that fixes about 60 bugs and introduces 70 new features suggested by users. Damien Doligez acted as release manager for both versions.

OCaml 4.02.0 introduces a large number of major innovations:

- Extension points: a uniform syntax for adding attributes and extensions in OCaml source code: most external preprocessors can now extend the language without need to extend the syntax and reimplement the parser.
- Improvements to the module system: generative functors and module aliases facilitate the efficient handling of large code bases.
- Separation between text-like read-only strings and array-like read-write byte sequences. This makes OCaml programs safer and clearer.
- An extension to the pattern-matching syntax to catch exceptions gives a short, readable way to write some important code patterns.
- Extensible open datatypes generalize the exception type and make its features available for general programming.
- Several important optimizations were added or enhanced: constant propagation, common subexpression elimination, dead code elimination, optimization of pattern-matching on strings.
- A code generator for the new 64-bit ARM architecture "AArch64".
- A safer and faster implementation of the printf function, based on the GADT feature introduced in OCaml 4.00.0.

This version has also seen a reduction in size: the Camlp4 and Labltk parts of the system are now independent systems. This makes them free to evolve on their own release schedules, and to widen their contributor communities beyond the core OCaml team.

OCaml 4.02.1 fixes a few bugs introduced in 4.02.0, along with 25 older bugs.

In parallel, we designed and experimented with several new features that are candidates for inclusion in the next major releases of OCaml:

- Ephemerons: a more powerful version of weak pointers.
- A parallel extension of the runtime system and associated language features that will let multi-threaded OCaml programs run in parallel on several CPU cores.
- Modular implicits: a typeclass-like extension that will make is easy to write generic code (*e.g.* print functions, comparison predicates, overloaded arithmetic operators, etc).
- "Inlined" records as constructor arguments, which will let the programmer select a packed representation for important data structures.
- Major improvements to the inlining optimization pass.
- Support for debugging native-code OCaml programs with GDB.

6.4.2. Namespaces for OCaml

Participants: Fabrice Le Fessant, Pierrick Couderc.

With the growth of the OCaml community and the ease of sharing code through OPAM, the new OCaml package manager, OCaml projects are using more and more external libraries. As a consequence, conflicts between module names of different libraries are now more likely for big projects, and the need for switching from the current flat namespace to a hierarchical namespace is now real.

We experimented with a prototype of OCaml where the namespaces used by a module are explicitely written in the OCaml module source header, to generate the environment in which the source is typed and compiled [39]. Namespaces are mapped on directories on the disk. This mechanism complements the recent addition of module aliases to OCaml, by providing extensibility at the namespace level, whereas it is absent at the module level, and solves also the problem of exact dependency analysis (the previous tool used for that purpose, ocamldep, provides only an approximation of the dependencies, computed on the syntax tree).

6.4.3. Memory profiling OCaml application

Participants: Fabrice Le Fessant, Çagdas Bozman [ENSTA ParisTech], Grégoire Henry [OCamlPro], Michel Mauny [ENSTA ParisTech].

Most modern languages make use of automatic memory management to discharge the programmer from the burden of allocating and releasing the chunks of memory used by the software. As a consequence, when an application exhibits an unexpected usage of memory, programmers need new tools to understand what is happening and how to solve such an issue. In OCaml, the compact representation of values, with almost no runtime type information, makes the design of such tools more complex.

We have experimented with three tools to profile the memory usage of real OCaml applications. The first tool saves snapshots of the heap after every garbage collection. Snapshots can then be analysed to display the evolution of memory usage, with detailed information on the types of values, where they were allocated and from where they are still reachable. A second tool updates counters at every garbage collection event, it complements the first tool by providing insight on the behavior of the minor heap, and the values that are promoted or not to the major heap. Finally, a third tool samples allocations and saves stacks of function calls at these samples.

These tools have been used on real applications (Alt-Ergo, an SMT solver, or Cumulus, an Ocsigen website), and allowed us to track down and fix memory problems with these applications, such as useless copies of data structures and memory leaks.

6.4.4. OPAM, the OCaml package manager

Participants: Fabrice Le Fessant, Roberto Di Cosmo [IRILL], Louis Gesbert [OCamlPro].

With the growth of the OCaml community, the need for sharing libraries between users has lead to the development of a new package manager, called OPAM. OPAM is based on Dose, a library developed by the Mancoosi team at IRILL, to provide a unified format, CUDF, to query external dependency solvers. The specific needs of OPAM have driven interesting research and improvements on the Dose library, that have consequently opened new opportunities for improvements in OPAM, for the benefit of both software.

We have for example experimented with the design of a specific language [37] to describe optimization criteria, when managing OPAM packages. Indeed, depending on the actions (installation, upgrade, removal), the user might want to reach very different configurations, requiring an expressive power that go far beyond what traditional package managers can express in their configuration options. For example, during installation, the user would probably see as little compilation as possible, whereas upgrading is supposed to move the configuration to the most up-to-date state, with as much compilation as needed.

We have also proposed a new paradigm: multi-switch constraints, to model *switches* used in OPAM to handle different versions of OCaml on the same computer [41]. We proposed this new paradigm as a way to solve multiple problems (cross-compilation, multi-switch packages, per-switch repositories and application-specific switches). However, we expect this new paradigm to challenge the scalability of the current CUDF solvers used by OPAM, and to require important changes and optimization in the Dose library.

6.5. Software specification and verification

6.5.1. Tools for TLA+

Participants: Damien Doligez, Jael Kriener, 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 [61], and to build tools for writing TLA+ specifications and mechanically checking the corresponding formal proofs.

This year, we released two versions of the TLA+ Proof System (TLAPS), the part of the TLA+ tools that handles mechanical checking of TLA+ proofs. This environment is described in [55].

These versions add the propositional temporal logic prover LS4 as a back-end, which allows TLAPS to deal with propositional temporal formulas. This relies on a technique called *coalescing* [40], which allows users to prove arbitrary safety properties, as well as some liveness properties, by translating them into the back-end prover's logic without increasing the complexity of the formulas.

Jael Kriener started a post-doc contract in December 2013, funded by the ADN4SE contract, and left in September 2014. She worked on the theory of temporal proofs in TLA+ and, in collaboration with CEA, on proving some properties of the PharOS real-time operating system.

Web sites:
http://research.microsoft.com/users/lamport/tla/tla.html
http://tla.msr-inria.inria.fr/tlaps

6.5.2. The Zenon automatic theorem prover

Participants: Damien Doligez, David Delahaye [CNAM], Pierre Halmagrand [Equipe DEDUCTEAM], Guillaume Bury [Equipe DEDUCTEAM], Olivier Hermant [Mines ParisTech].

Damien Doligez continued the development of Zenon, a tableau-based prover for first-order logic with equality and theory-specific extensions.

Pierre Halmagrand continued his thesis work, funded by ANR BWare, on integrating Deduction Modulo in Zenon, with emphasis on making it efficient for dealing with B set theory.

Guillaume Bury did an internship, also funded by ANR BWare. He implemented an extension of Zenon, based on the simplex method, to deal with arithmetic formulas.

6.5.3. Well-typed generic fuzzing for module interfaces

Participants: Thomas Braibant, Jonathan Protzenko, Gabriel Scherer.

Property-based testing generates arbitrary instances of inputs to check a given correctness predicate/property. Thomas Braibant proposed that, instead of a random generation function defined from the internals of one's data-structure, one could use the user-exposed interface to generate instances by composition of API calls. GADTs let us reflect/reify a typed API, and program a type-respecting exploration/testing directly in the host language. We developed a prototype library, Articheck, to experiment with this idea. This work was presented at the ML Workshop [38].

6.5.4. Depth-First Search and Strong Connectivity in Coq

Participant: François Pottier.

In 2002, Ingo Wegener published a short paper which sketches a proof of Kosaraju's linear-time algorithm for computing the strongly connected components of a directed graph. At the same time, Wegener's paper helps explain why the algorithm works, which, from a pedagogical standpoint, makes it quite valuable. In 2013 and 2014, François Pottier produced a machine-checked version of Wegener's proof, and wrote a precise informal account of it, which will be presented at JFLA 2015 [36].

6.5.5. Implementing hash-consed structures in Coq

Participants: Thomas Braibant, Jacques-Henri Jourdan, David Monniaux [CNRS, VERIMAG].

Hash-consing is a programming technique used to implement maximal sharing of immutable values in memory, keeping a single copy of semantically equivalent objects. Hash-consed data-structures give a unique identifier to each object, allowing fast hashing and comparisons of objects. This may lead to major improvements in execution time by itself, but it also makes it possible to do efficient memoization of computations.

Hash-consing and memoization are examples of imperative techniques that are of prime importance for performance, but are not easy to implement and prove correct using the purely functional language of a proof assistant such as Coq.

We published an article in Journal of Automated Reasoning [15], explaining our work on this subject during the last 3 years. We gave four different approaches for implementing hash-consed data-structures in Coq. Then, we performed an in-depth comparative study of how our "design patterns" for certified hash-consing fare on two real-scale examples: BDDs and lambda-terms.

MARELLE Project-Team

6. New Results

6.1. Highlights of the Year

In June 2014, Yves Bertot received the ACM Software System award, as one of the main contributors to the Coq System, along with Gérard Huet, Thierry Coquand, Christine Paulin-Mohring, Bruno Barras, Jean-Christophe Filliâtre, Hugo Herbelin, Chet. Murthy, and Pierre Castéran.

6.2. Proof and computation

Participants: Laurent Théry [correspondant], Benjamin Grégoire.

We have been continuing our effort to improve the computing power of Coq. This has led to two "computational proof":

The Erdös conjecture for n = 2 was proved this year using a SAT solver. We succeeded to formally prove this instance in Coq independently checking the 3Gb trace of the SAT solver.

The weak Goldbach conjecture was proved last year by Harald Helfgott. This proof requires a computation that the conjecture holds for numbers less than 10^{28}. This is done in two stages. The first one is to verify Goldbach conjecture for numbers less than 10^{18}. The second one is to verify the weak Goldbach conjecture for numbers less than 10^{28} using a ladder with intervals 10^{18}. The second stage has been completely verified in Coq. We are currently working on improving the computation power of Coq to make it possible to perform the first stage in reasonable time.

6.3. Formal verification of automated proof algorithms

Participant: Laurent Théry [correspondant].

We have been interested in proving that the classic 2-Sat problem can be solved in linear time. This leads to proving two classic algorithms:

1. A version of Kosaraju's algorithm that computes the strongly connected components of a directed graph [21],
2. A more direct algorithm that solves the 2-Sat problem that is using unit propagation, proposed by Alvaro del Val [20].

6.4. Formal study of cryptography

Participants: Gilles Barthe [IMDEA], Sonia Belaid [THALES and ENS], François Dupressoir [IMDEA], Pierre-Alain Fouque [Université de Rennes 1 and Institut universitaire de France], Cédric Fournet [Microsoft Research], Benjamin Grégoire [correspondant], Benedikt Schmidt [IMDEA], Pierre-Yves Strub [IMDEA], Nikhil Swamy [Microsoft Research], Mehdi Tibouchi [NTT Secure Platform Laboratories], Santiago Zanella-Béguelin [Microsoft Research], Jean-Christophe Zapalowicz [Inria].

The goal of this work is to provide a friendly tool easily usable by cryptographers without knowledge of formal proof assistants. The idea is to use the techniques formally proved in Certycrypt and to call SMT-provers. We provide two differents tools EasyCrypt and ZooCrypt.

This year, we worked on the following topics:

- Relational program logics, as used in EasyCrypt, have been used for mechanizing formal proofs of various cryptographic constructions. In [15], we present rF*, a relational extension of F*, a general-purpose higher-order stateful programming language with a verification system based on refinement types. The distinguishing feature of rF* is a relational Hoare logic for a higher-order, stateful, probabilistic language.
- Fault Attacks are attacks in which an adversary with physical access to a cryptographic device, say a smartcard, tampers with the execution of an algorithm to retrieve secret material. In [13] we propose a new approach for finding fault attacks based on fault conditions. Using the method, we discover multiple fault attacks on RSA and ECDSA. Several of the attacks found by our tool are new. In [14], we propose a new counter measure to make RSA-PSS provably secure against non-random faults. We also prove the result using EasyCrypt.
- Many algorithms, particularly in cryptography, admit very efficient batch versions that compute simultaneously the output of the algoritms on a set of inputs. AutoBatch is a tool that computes highly optimized batch verification algorithms for pairing based signature schemes. In [12], we use EasyCrypt to formalise the methods used by AutoBatch and to automatically certify the result of the transformation performed by AutoBatch.
- We study the problem of automatically verifying higher-order masking countermeasures which is used to protect implementations where the attacker can observe intermediate computations (like in a smartcard). We propose an efficient method to check the correctness and the security of masked implementation. This work has been submitted to EuroCrypt 2015. We start the ANR BRUTUS on this subject.

6.5. Formalization of Bourbaki's sets and ordinals

Participant: José Grimm.

In previous years we developed a formal library describing the parts of the Bourbaki books on set theory, cardinals and ordinals. We completed it by adding the definition of real numbers using Dedekind cuts. The important properties we showed that $\mathbf{R}$ is an ordered Archimedean field, that every non-empty bounded subset has a least upper bound, that every Cauchy sequence has a limit, and that the intermediate value theorem holds.

It follows that every positive real number has positive square root. We give a pair of adjacent sequences that converges to this square root. For instance $\sqrt{2}$ is irrational, and we get a pair of rational adjacent sequences that converges to it. This produces an explicit order isomorphism $\mathbf{Q}^* \to \mathbf{Q}$. The number of such isomorphisms is equal to the power of the continuum (the cardinal of $\mathbf{R}$) [18].

6.6. Stern-Brocot and Fibonacci sequences

Participant: José Grimm.

We constructed an explicit bijection $\mathbf{N} \to \mathbf{Q}$, first in the framework of the Bourbaki project (see above), then in Ssreflect. Every positive rational number x can uniquely be written as a quotient s_n/s_{n+1}. This result was established by Dijkstra who stated it in an obfuscated way. It was shown years before by Stern. It is possible to compute s_n/s_{n+1} without computing numerator and denominator separately, by considering the sequences of bits of n from left to right or from right to left. Truncating the binary expansion of n yields a sequence of approximations to s_n/s_{n+1} (this was studied by Brocot, and the so-called Stern-Brocot tree is an alternative representation of rational numbers). We implemented the work of Dijkstra and Stern in Coq [17].

We also studied how a number can be represented by a sequence of other numbers (for instance as a sum of distinct Fibonacci numbers, with or without constraints). The number of ways of writing n as a sum of powers of two, each power of two being used at most twice, is s_{n+1}. These results are presented in [17].

6.7. Formal proof that e and π are transcendental

Participants: Sophie Bernard, Laurence Rideau.

We constructed formal proofs that π is irrational, e is transcendental, and π is transcendental. These proofs share a common initial pattern, where rationality or algebraicity of the mathematical constants are shown to imply the existence of a sequence of positive integers that must decrease indefinitely.

This proof development is an opportunity to study the interplay between several existing libraries about algebraic structures and analysis: the ssreflect library for algebra and the Coquelicot library for calculus. Moreover, the proof that π is transcendental was an occasion to test the newly developed module on symmetric polynomials by P.-Y. Strub at IMDEA.

6.8. Fast computation of π

Participant: Yves Bertot.

In the previous year, we studied a proof that π could be approximated with a fast converging sequence based on arithmetic geometric means. This year we described a proof that rounding errors during this computation could be guaranteed as small as needed, based on a study of derivatives. This approach provides a fruitful alternative to interval-based approaches. The result was published in [16].

We also completed a journal paper on various ways to observe and compute the number π [7].

6.9. Decision procedures for polynomials

Participant: Yves Bertot.

Following up on the work in previous years around Bernstein Polynomials, we implemented a decision procedure for guaranteeing the sign of a polynomial function inside an interval, using Bernstein polynomials and dichotomy. In the long run, we hope to explore two approaches, one based on the off-line computation of certificates for sub-intervals (these certificates are easy to verify), and one based on implementing computational reflection. This approach should also generalize quite easily to multi-variate polynomials.

MEXICO Project-Team

6. New Results

6.1. Highlights of the Year

6.1.1. Active Diagnosis for Probabilistic Systems

Diagnosis fits well with probabilistic systems since it is natural to model the uncertainty about the behaviour of a partially observed system by distributions. We had previously revisited the active diagnosis (which aims at controlling the system to make it diagnosable) in discrete event systems designing optimal decision and synthesis procedures [7]. This year, we have considered active diagnosis for probabilistic discrete event systems, obtaining again optimal procedures [26]. Furthermore we have refined the notion of active diagnosis by introducing the *safe active diagnosis* which ensures that after the control is applied, there is a positive probability that a fault never occurs. Interestingly this problem is undecidable but for finite memory controller we have shown that the problem becomes again decidable and we have designed optimal decision and synthesis procedures. Our approach has raised an issue that has not be observed by previous researchers: while in discrete event system, most variants of diagnosis are in fact equivalent, this is no more the case for probabilistic systems. So in [26], we have undertaken the task of classifying the different versions obtaining a complete landscape of the notions both in terms of relations and complexity. Furthermore we have proposed a new notion of diagnosis, the *prediagnosis* that combines the advantages of diagnosis and prediction.

6.1.2. Weighted automata and weighted logics

Weighted automata are a conservative quantitative extension of finite automata that enjoys applications, e.g., in language processing and speech recognition. Their expressive power, however, appears to be limited, especially when they are applied to more general structures than words, such as graphs. To address this drawback, we have introduced weighted pebble walking automata, which allow to navigate freely in the graph and may use pebbles to mark some positions.

In [20], we have shown 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 have extended Kleene-Schu ¨tzenberger theorem showing that weighted expressions and automata with pebbles have the same expressive power. We focussed on an efficient translation from expressions to automata. We also proved that the evaluation problem for weighted automata can be done very efficiently if the number of reusable pebbles is low.

In [18], we have studied the expressive power of these automata on words. We have proved that two-way pebble weighted automata, one-way pebble weighted automata, and our weighted logic with transitive closure are expressively equivalent. We also gave new logical characterizations of standard recognizable series.

In [30], we addressed the more general case of graphs such as nested words, trees, pictures, Mazurkiewicz traces, ... We established that weighted pebble walking automata have the same expressive power as weighted first order logic with transitive closure logic, lifting a similar result by Engelfriet and Hoogeboom from the Boolean case to a quantitative setting.

6.1.3. Verification of concurrent recursive programs

Distributed systems form a crucially important but particularly challenging domain. Designing correct distributed systems is demanding, and verifying its correctness is even more so. The main cause of difficulty here is concurrency and interaction (or communication) between various distributed components. Hence it is important to provide a framework that makes easy the design of systems as well as their analysis. There are two schools of thought on reasoning about distributed systems: one following the interleaving based semantics, and one following the visual partial-order/graph based semantics. In [23], we compare these two approaches and argue in favour of the latter. An introductory treatment of the split-width technique is also provided.

In [34], we develop a general technique based on split-width for the verification of networks of multi-threaded recursive programs communicating via reliable FIFO channels. We extend the approach of [6] to this setting. Split-width offers an intuitive visual technique to decompose our behaviour graphs such as MSCs and nested words. The decomposition is mainly a divide-and-conquer technique which naturally results in a tree decomposition. Every behaviour can now be interpreted over its decomposition tree. Properties over the behaviour naturally transfer into properties over the decomposition tree. This allows us to use tree-automata techniques to obtain decision procedures for a range of problems such as reachability, model checking against logical formalisms etc. In this way, we obtain simple, uniform and optimal decision procedures for various verification problems parametrised by split-width. Furthermore, the simple visual mechanism of split-width is as powerful as yardstick graph measures such as tree-width or clique-width. Hence it captures any class of distributed behaviours with a decidable MSO theory.

Multi-threaded recursive programs communicating via channels are turing powerful, hence their verification has focussed on under-approximation techniques. Any error detected in the under-approximation implies an error in the system. However the successful verification of the under-approximation is not as useful if the system exhibits unverified behaviours. In [24], we study controllers that observe/restrict the system so that it stays within the verified under-approximation. We identify some important properties that a good controller should satisfy. We consider an extensive under-approximation class, construct a distributed controller with the desired properties and also establish the decidability of verification problems for this class.

6.1.4. Regulation in Systems Biology

6.1.4.1. Rare events in Signalling Cascades

The visit in 2013 of Professor Monika Heiner from Cottbus University has led to a fruitful collaboration related to statistical model checking of rare events in signalling cascades (a regulatory biological system) [25]. This work has received one of the five top paper awards of the conference. In addition, we have improved the statistical methods used in our tool Cosmos.

6.1.4.2. Characterization of Reachable Attractors Using Petri Net Unfoldings

Attractors of network dynamics represent the long-term behaviours of the modelled system. Their characterization is therefore crucial for understanding the response and differentiation capabilities of a dynamical biological system. In the scope of qualitative models of interaction networks, the computation of attractors reachable from a given state of the network faces combinatorial issues due to the state space explosion.

In [33], we have presented a new algorithm that exploits the concurrency between transitions of parallel acting components in order to reduce the search space. The algorithm relies on Petri net unfoldings that can be used to compute a compact representation of the dynamics. We have illustrated the applicability of the algorithm with Petri net models of cell signalling and regulation networks, boolean and multi-valued. The proposed approach aims at being complementary to existing methods for deriving the attractors of Boolean models, while being generic since it applies to any safe Petri net.

6.2. Diagnosis

6.2.1. Diagnosability under Weak Fairness

In partially observed Petri nets, diagnosis is the task of detecting whether or not the given sequence of observed labels indicates that some unobservable fault has occurred. Diagnosability is an associated property of the Petri net, stating that in any possible execution an occurrence of a fault can eventually be diagnosed. In [35] we consider diagnosability under the weak fairness (WF) assumption, which intuitively states that no transition from a given set can stay enabled forever; it must eventually either fire or be disabled. Following our previous work [71] on how to perform *weak diagnosis* by exploiting the fact that weak fairness reveals faults in parallel with the current observation, sometimes even before their actual accurrence, we turn to the associated *diagnosability* problem in [35]. First, we show that a previous approach to WF-diagnosability in the literature has a major flaw, and present a corrected notion. Moreover, we present an efficient method for verifying WF-diagnosability based on a reduction to LTL-X model checking. An important advantage of this

method is that the LTL-X formula is fixed ? in particular, the WF assumption does not have to be expressed as a part of it (which would make the formula length proportional to the size of the specification), but rather one exploits the ability of existing model checkers to handle weak fairness directly.

6.3. Asynchronous Testing

In the final year of the TECSTES project, we have extended and completed the co-ioco - based conformance and testing theory that we had developed thus far and published in [21], in several directions:

- The testing framework now provides a test generation algorithm [21] for concurrent systems specified with true concurrency models, such as Petri nets or networks of automata. The semantic model of computation of such formalisms are labeled event structures, which allow to represent concurrency explicitly.
- Our test generation algorithm based on Petri net unfolding is able to build a complete test suite w.r.t our co-ioco conformance relation [22]. In addition we propose several coverage criteria that allow to select finite prefixes of an unfolding in order to build manageable test suites.
- We propose an extension of the *ioco* conformance relation, a standard for labeled event structures, named co-ioco, allowing to deal with strong and weak concurrency. We extend the notions of test cases and test execution to labeled event structures, and give a test generation algorithm building a complete test suite for co-ioco. Further, we have introduced and exploited [21] the notions of *strong* and *weak* concurrency: strongly concurrent events must be concurrent in the implementation, while weakly concurrent ones may eventually be ordered, leading to refine *co-ioco* into the *wsc-ioco* relation accounting for weak and strong concurrency.
- The *co-ioco* relation assumes a global control and observation of the system under test, which is not usually realistic in the case of physically distributed systems. Such systems can be partially observed at each of their points of control and observation by the sequences of inputs and outputs exchanged with their environment. Unfortunately, in general, global observation cannot be reconstructed from local ones, so global conformance cannot be decided with local tests. We showed in [39] how appending time stamps to the observable actions of the system under test in order to regain global conformance, via vector clock information, from local testing.
- The MOLE - based testing tool TOURS [42] has been developed with the help of intern Konstantinos Athanasiou, jointly supervised by Hernán Ponce de León and Stefan Schwoon of the MExICo team at LSV), and successful experiments have been conducted with a scalable benchmark example (elevator control). The results show clearly how the true-concurrency approach leads to the test case required being not only smaller individually, but also that *fewer* such test cases are necessary. In addition to the conceptual and analytical enrichment, the results obtained in TECSTES thus also allow to obtain important speedups and reductions in storage space.

Hernán Ponce de León has completed his thesis [40] reporting on the above results, and very successfully defended on Nov. 7, 2014, at ENS Cachan, before the PhD committee consisting of reviewers Rob Hierons and Alex Yakovlev, examiners Thierry Jeron, Remi Morin and Pascal Poizat, and the two supervisors.

6.4. Reachability in MDPs

Markov decision process (MDP) provide the appropriate formalism for the control of fully observable probabilistic systems. There are three kinds of methods for their analysis: linear programming, policy iteration and value iteration. However for large scale systems, only value iteration is still available as it requires less memory than the other methods. For quantitative problems like optimal control for maximizing the discounted reward of an MDP, value iteration is equipped with a stopping criterion that ensures an error bound provided by the user. Value iteration algorithms have also been proposed for the central problem of reachability. However neither stopping criterion nor convergence rate were known for such algorithms. In [37], we have solved these two problems and based on it we have also improved the bound on the number of iterations in order to adapt the value iteration for an exact computation.

6.5. Parameterized Communicating Automata

As a part of our research program on concurrent systems with variable communication topology, we studied system models where the topology is *static* but *unknown*, so that it becomes a parameter of the system. In [28], we introduced parameterized communicating automata (PCAs), where finite-state processes exchange messages via rendez-vous or through bounded FIFO channels. Unlike classical communicating automata, a given PCA can be run on any network topology of bounded degree. We presented various Büchi-Elgot-Trakhtenbrot theorems for PCAs, which roughly read as follows: Let φ be an existential MSO formula and T be any of the following topology classes: pipelines, ranked trees, grids, or rings. There is a PCA that is equivalent to φ on all topologies from T. In the case where each process executes a bounded number of contexts (each context restricting communication in a suitable way), we could show that PCAs are closed under complementation, are expressively equivalent to full MSO logic [29], and have a decidable emptiness problem [31]. The papers [29], [31] are a result of a collaboration with Akshay Kumar (IIT Kanpur) and Jana Schubert (TU Dresden).

6.6. Quantitative behaviours

Several measures have been proposed in literature for quantifying the information leaked by the public outputs of a program with secret inputs. In [32] we studied how to quantify the information leaked by a deterministic or probabilistic program when the measure of information is based on min-entropy or Shannon entropy. A direct computation of these quantities is often infeasible because of the state-explosion problem. In our paper, we model the program as a pushdown system equipped with multi-terminal decision diagrams (ADDs) and propose algorithms to compute said entropies.

The advantage of this approach is that the resulting algorithms can be easily implemented in any BDD-based model-checking tool that checks for reachability in deterministic non-recursive programs by computing program summaries. We demonstrate the validity of our approach by implementing these algorithms in a tool Moped-QLeak.

PARSIFAL Project-Team

6. New Results

6.1. Highlights of the Year

Dale Miller's 1994 LICS paper titled "A Multiple-Conclusion Meta-Logic" [67] was a co-recipient of the LICS Test of Time Award.

6.2. Modular Systems for Classical and Intuitionistic Logic

Participants: Sonia Marin, Lutz Straßburger.

Last year we have shown deductive systems for all intuitionistic modal logics in the modal S5-cube using logical rules in nested sequents [75]. This year we managed to exhibit fully modular systems. That is to say that there is a bijective correspondence between the modal axioms and the inference rules in the deductive system. This is achieved by using a combination of structural and logical rules. This result has been presented at AiML 2014 [24].

6.3. Nested Sequents for Constructive Modal Logics

Participants: Ryuta Arisaka, Anupam Das, Lutz Straßburger.

In the propositional case, "constructive" and "intuitionistic" logic are usually considered the same. However, in the presence of the modalities $\Box$ and $\diamond$ this situation changes because there are several choice of which variants of the k-axiom (which are all equivalent in the classical case) are to be included. Whereas in [75] the intuitionistic variant of the S5-cube has been studied, we studied in this years work [34] the constructive variant of the logics in the S5-cube.

6.4. Intuitionistic Logic in the Calculus of Structures

Participants: Nicolas Guenot, Lutz Straßburger.

The calculus of structures has mainly be used for "classical" logics that come with a De Morgan duality. The reason is that all normalization procedures developed so far for the calculus of structures rely on this De Morgan duality.

In this work, we give two proof systems for implication-only intuitionistic logic in the calculus of structures. The first is a direct adaptation of the standard sequent calculus to the deep inference setting. It comes with a cut elimination procedure that is similar to the one from the sequent calculus, using a non-local rewriting. The second system is the symmetric completion of the first, as normally given in deep inference for logics with a De Morgan duality: all inference rules have duals, as cut is dual to the identity axiom. For this symmetric system we prove a generalization of cut elimination, that we call symmetric normalization, where all rules dual to standard ones are permuted up in the derivation. The result is a decomposition theorem having cut elimination and interpolation as corollaries. This work has been presented at the CSL-LICS 2014 conference [22].

6.5. Free Theorems for Curry

Participant: Lutz Straßburger.

Free theorems [79] are a means of type-based reasoning and are being successfully applied for typed functional programming languages like Haskell, e.g., for program transformation and generally establishing semantic properties [53], [78]. As a simple example, for every polymorphic function $f :: [\alpha] \to [\alpha]$ from lists to lists, arbitrary types τ_1 and τ_2, and a function $g :: \tau_1 \to \tau_2$, we have $f \circ (\mathsf{map}\ g) = (\mathsf{map}\ g) \circ f$, for the standard function $\mathsf{map} :: (\alpha \to \beta) \to [\alpha] \to [\beta]$ which takes a function and a list and applies that function to every entry of the list. It would be of interest to also have such free theorems available for typed functional-logic languages like Curry.

Previous work [48] has investigated free theorems for such a language, Curry [60], phenomenologically and provides intuition for premises of free theorems as well as counterexamples. Proof of the positive claims has been elusive so far, mainly because Curry's type system fails to reflect the key feature: nondeterminism. This avoidance is convenient for programmers, as they do not have to distinguish between deterministic and nondeterministic values. However, it is a hindrance to formal reasoning: the conditions identified in [48] include a notion of determinism, and hence it is a serious weakness of the type system not to capture this.

In a joint work with colleagues at the University of Bonn, published in [25], we have developed an intermediate language, called SaLT, that allowed us to prove a *Parametricity Theorem* which could be used to derive free theorems for Curry.

This work is the result of the PHC Procope collaboration with the University of Bonn (duration 2012-2013).

6.6. A logical basis for quantum evolution and entanglement

Participant: Lutz Straßburger.

In discrete quantum causal dynamics, quantum systems are viewed as discrete structures, namely directed acyclic graphs. In such a graph, events are considered as vertices and edges depict propagation between events. Evolution is described as happening between a special family of space-like slices, which were referred to as locative slices in [41]. Such slices are not so large as to result in acausal influences, but large enough to capture nonlocal correlations. It was an open problem whether such slices can be captured by a deductive system, such that proof search corresponds to quantum evolution. In a joint work with Blute, Guglielmi, Ivanov, and Panangaden, Straßburger has shown that the logic BV with its mix of commutative and noncommutative connectives, is precisely the right logic for such analysis. More precisely, it was shown that the commutative tensor encodes (possible) entanglement, and the noncommutative *seq* encodes causal precedence. With this interpretation, the locative slices are precisely the derivable strings of formulas. Several new technical results about BV are developed as part of this analysis, which is published in [28]

6.7. On the Pigeonhole and Related Principles in Deep Inference and Monotone Systems

Participant: Anupam Das.

The size of proofs of the propositional pigeonhole principle over various systems is a topic of much interest in the proof complexity literature. In particular, it has received notable attention in recent years from the deep inference community, where its classification over the system KS appears as an open problem in numerous publications. In [21] we construct quasipolynomial-size proofs of the propositional pigeonhole principle in the deep inference system KS, addressing this question by matching the best known upper bound for the more general class of monotone proofs.

We make significant use of monotone formulae computing boolean threshold functions, an idea previously considered in works of Atserias et al. The main construction, monotone proofs witnessing the symmetry of such functions, involves an implementation of merge-sort in the design of proofs in order to tame the structural behavior of atoms, and so the complexity of normalization. Proof transformations from previous work on atomic flows are then employed to yield appropriate KS proofs.

As further results we show that our constructions can be applied to provide quasipolynomial-size KS proofs of the parity principle and the generalized pigeonhole principle. These bounds are inherited for the class of monotone proofs, and we are further able to construct $nO(\log\log n)$-size monotone proofs of the weak pigeonhole principle, thereby also improving the best known bounds for monotone proofs.

6.8. A multi-focused proof system isomorphic to expansion proofs

Participants: Kaustuv Chaudhuri, Stefan Hetzl [Vienna University of Technology, Vienna, Austria], Dale Miller.

The sequent calculus is often criticized for requiring proofs to contain large amounts of low-level syntactic details that can obscure the essence of a given proof. Because each inference rule introduces only a single connective, sequent proofs can separate closely related steps—such as instantiating a block of quantifiers—by irrelevant noise. Moreover, the sequential nature of sequent proofs forces proof steps that are syntactically non-interfering and permutable to nevertheless be written in some arbitrary order. The sequent calculus thus lacks a notion of *canonicity*: proofs that should be considered essentially the same may not have a common syntactic form. To fix this problem, many researchers have proposed replacing the sequent calculus with proof structures that are more parallel or geometric. Proof-nets, matings, and atomic flows are examples of such *revolutionary* formalisms. In [13], we propose, instead, an *evolutionary* approach to recover canonicity within the sequent calculus, which we illustrate for classical first-order logic. The essential element of our approach is the use of a *multi-focused* sequent calculus as the means for abstracting away low-level details from classical cut-free sequent proofs. We show that, among the multi-focused proofs, the *maximally multi-focused* proofs that collect together all possible parallel foci are canonical. Moreover, if we start with a certain focused sequent proof system, such proofs are isomorphic to *expansion proofs*—a well known, minimalistic, and parallel generalization of Herbrand disjunctions—for classical first-order logic. This technique appears to be a systematic way to recover the "essence of proof" from within sequent calculus proofs.

6.9. Equality and fixpoints in the calculus of structures

Participants: Kaustuv Chaudhuri, Nicolas Guenot [IT University of Copenhagen, Denmark].

The standard proof theory for logics with equality and fixpoints suffers from limitations of the sequent calculus, where reasoning is separated from computational tasks such as unification or rewriting. We propose in [20] an extension of the calculus of structures, a deep inference formalism, that supports incremental and contextual reasoning with equality and fixpoints in the setting of linear logic. This system allows deductive and computational steps to mix freely in a continuum which integrates smoothly into the usual versatile rules of multiplicative-additive linear logic in deep inference.

6.10. Automatically deriving schematic theorems for dynamic contexts

Participants: Kaustuv Chaudhuri, Olivier Savary-Bélanger [Princeton University, USA].

Hypothetical judgments go hand-in-hand with higher-order abstract syntax for meta-theoretic reasoning. Such judgments have two kinds of assumptions: those that are statically known from the specification, and the *dynamic assumptions* that result from building derivations out of the specification clauses. These dynamic assumptions often have a simple regular structure of repetitions of *blocks* of related assumptions, with each block generally involving one or several variables and their properties, that are added to the context in a single backchaining step. Reflecting on this regular structure can let us derive a number of structural properties about the elements of the context.

In [26], we present an extension of the Abella theorem prover, which is based on a simply typed intuitionistic reasoning logic supporting (co-)inductive definitions and generic quantification. Dynamic contexts are represented in Abella using lists of formulas for the assumptions and quantifier nesting for the variables, together with an inductively defined *context relation* that specifies their structure. We add a new mechanism for defining particular kinds of regular context relations, called *schemas*, and *tacticals* to derive theorems from these schemas as needed. Importantly, our extension leaves the trusted kernel of Abella unchanged. We show that these tacticals can eliminate many commonly encountered kinds of administrative lemmas that would otherwise have to be proven manually, which is a common source of complaints from Abella users.

6.11. A two-level logic approach for reasoning about typed specification languages

Participants: Kaustuv Chaudhuri, Mary Southern [University of Minnesota, USA].

The *two-level logic approach* (*2LLA*) to reasoning about computational specifications, as implemented by the Abella theorem prover, represents derivations of a *specification language* as an inductive definition in a *reasoning logic*. This approach has traditionally been formulated with the specification and reasoning logics having the *same* type system, and only the formulas being translated. However, requiring identical type systems limits the approach in two important ways: (1) every change in the specification language's type system requires a corresponding change in that of the reasoning logic, and (2) the same reasoning logic cannot be used with two specification languages at once if they have incompatible type systems. In [27], we propose a technique based on *adequate* encodings of the types and judgments of a typed specification language in terms of a simply typed higher-order logic program, which is then used for reasoning about the specification language in the usual *2LLA*. Moreover, a single specification logic implementation can be used as a basis for a number of other specification languages just by varying the encoding. We illustrate our technique with an implementation of the LF dependent type theory as a new specification language for Abella, co-existing with its current simply typed higher-order hereditary Harrop specification logic, without modifying the type system of its reasoning logic.

6.12. Undecidability of multiplicative subexponential logic

Participant: Kaustuv Chaudhuri.

Subexponential logic is a variant of linear logic with a family of exponential connectives—called *subexponentials*—that are indexed and arranged in a pre-order. Each subexponential has or lacks associated structural properties of weakening and contraction. In [18], we show that classical propositional multiplicative linear logic extended with one unrestricted and two incomparable linear subexponentials can encode the halting problem for two register Minsky machines, and is hence undecidable.

6.13. Meta-theoretic results on type isomorphisms in the presence of sums

Participant: Danko Ilik.

Type isomorphisms are a pervasive notion of Theoretical Computer Science. In functional programming, two data types being isomorphic means that we can coerce data and programs back-and-forth between two specifications without loss of information. In Constructive Mathematics, two sets are of the same cardinality exactly when they are isomorphic as types. In the proof theory of intuitionistic logic, two formulas are strongly equivalent precisely when they are isomorphic as types.

However, the theory of simple types made from functions, products, and sums, is well understood only when we do not treat functions and sums at the same time. Fiore, Di Cosmo, and Balat [50], presented a "negative" results: the theory of those type isomorphisms is not finitely axiomatizable. To establish the result, they used the work around the Tarski High School Algebra Problem from Mathematical Logic.

We showed that the picture is not so dark by presenting a positive result: the theory is recursively axiomatizable and decidable. The proofs exploit further the deep theory around Tarski's Problem. This work was presented at the Joint Meeting of the Twenty-Third EACSL Annual Conference on Computer Science Logic (CSL) and the Twenty-Ninth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS) in Vienna, Austria [23].

6.14. Towards proof canonicity in presence of disjunction and induction

Participants: Hichem Chihani, Danko Ilik.

The previous work on type isomorphisms showed a way to treat the problem of identity/canonicity of proofs for intuitionistic logic with disjunction, or, equivalently, the problem of the (non-)existence of a canonical eta-long normal form for lambda calculus with if-expressions, which is a long standing open question.

One can see this from the perspective of focusing sequent calculi. The asynchronous phase of proof search is an oriented application of type isomorphisms (by the formulas-as-types correspondence). As we already know that, in the absence of disjunction (sum types), a cut-free focused derivation is eta-long and unique (when the data provided by the synchronous phase is the same), what is necessary in order to handle disjunction is to propagate isomorphisms further than what usual sequent calculus allows. This is related in spirit to deep inference, but more conservative. An implementation of a canonical normalizer and a paper on the topic is under way.

We also intend to use the method to give a proof of focused cut-elimination for the sequent calculi LJF and LKF (at least, for the Sigma-2 fragment) extended with induction. A formal proof in Agda is under development.

6.15. Interpretation of the Sigma-2-classical Axiom of Choice in System T

Participant: Danko Ilik.

Updating previous work, we showed that one can develop a realizability interpretation for the Σ^0_2-fragment of classical Analysis in System T only [36].

This is known to be possible, in principle, by a 1979 result of Schwichtenberg. However, up to day no method that avoids both bar recursion (Spector) and control operators (Krivine) has been known. In fact, we propose to treat control operators as a meta-mathematical technique, rather than to have them in the language of realizers as classical realizability does; we provide a formal proof in Agda that control operators can be completely normalized away from System T while preserving essential equations. [15]

6.16. Axiomatization of constraint systems for first-order reasoning modulo a theory

Participants: Damien Rouhling, Stéphane Graham-Lengrand, Assia Mahboubi, Jean-Marc Notin, Mahfuza Farooque.

This result is part of a work in theorem proving, whose purpose is to provide a theoretical basis for the handling of quantifiers in presence of a theory for which we have specific decision procedures. Inspired by the way first-order unifiers are generated and propagated in automated reasoning techniques such as *tableaux* methods, we sought to generalise these mechanisms to the presence of a theory: We introduced a axiomatic notion of constraint system and a sequent calculus introducing meta-variables and propagating constraints. We then identified the axioms that should be satisfied by the theory's decision procedure, in order for the sequent calculus to be sound and complete. This provides the theoretical basis for the development of Psyche 2.0. This result is submitted for publication.

6.17. Realisability models for cut-elimination in focused systems

Participant: Stéphane Graham-Lengrand.

This result is part of the effort to build meaningful semantics for classical proofs, here based on a polarisation of logical formulae: positive or negative.

Following work by Zeilberger [80], a computational interpretation of cut-elimination in the focused systems LJF and LKF can be given: proofs of positive formulae provide structured data, while proofs of negative formulae consume such data; focusing allows the description of the interaction between the two kinds of proofs as pure pattern-matching.

First, we showed this at a level of abstraction where formulae are no longer made of syntax, yet we also extended the approach so that it could treat quantifiers.

Second, we connected this interpretation to realisability semantics, more precisely orthogonality models, where positive formulae are interpreted as sets of data, and negative formulae are interpreted as their orthogonal sets.

Our construction of orthogonality models for the focused systems LKF and LJF describe the pattern-matching process of cut-elimination in terms of orthogonality. This result has been proved in the Coq proof assistant and forms the second part of [11].

6.18. Refining the FPC framework

Participants: Roberto Blanco, Zakaria Chihani, Quentin Heath, Dale Miller, Fabien Renaud.

We have continued to develop our approach to Foundational Proof Certificates (FPCs). This framework allows defining proof evidence in a general fashion. Proofs in both intuitionistic and classical logics are definable in this framework. We originally have written two different kernels for checking these results but more recently we have found that we can exploit an encoding due to Chaudhuri [43] that enables us to only implement the intuitionistic kernel and then simply encode the classical formulas so that they operator directly on the intuitionistic kernel. This encoding allows for a much more precise and simple means for encoding classical logic into intuitionistic logic than the more familiar double negation translations.

We have also started to develop the second phase of defining proof evidence that was proposed in the ProofCert proposal: the definition of proofs that require fixed points (induction / co-induction). We now have two different kernels being developed on top of the Bedwyr model checker that are checking (and in some cases, proving) theorems involving induction, reachability, and bisimulation.

6.19. Structuring a refinement engine using logic programming

Participants: Dale Miller, Claudio Sacerdoti Coen [University of Bologna], Enrico Tassi [MSR Inria Joint Lab].

The Matita theorem prover is an implementation of the Calculus of Inductive Constructions that is meant to be more accessible (as an implementation) than the Coq system. In an effort to make the Matita kernel more accessible and more flexible, the implementers of that system are experimenting with using a logic programming language similar to λProlog as the control system of the refinement mechanism. In order to use such a logic programming language in this capacity, the notion of flexible goal suspension and *when* declarations are needed. Such a λProlog re-implementation has been written and some experiments in deploying such a system are underway. Formal aspects of λProlog specifications have also been performed using the Abella theorem prover.

PI.R2 Project-Team

5. New Results

5.1. Highlights of the Year

We successfully organised the thematic trimester Semantics of Proofs and Certified Mathematics (IHP, April-July 2014). The trimester attracted over two hundred participants altogether (with about 60 "resident" participants staying a month or more), hosted 5 special workshops, as well as other related regevents such as Types, MAP (Mathematics, Algorithms, and Proofs). It was the first thematic trimester in the history of IHP to feature computer science prominently. There was a kick-off day on April 22, with talks of Georges Gonthier, Thomas Hales, Xavier Leroy, and Vladimir Voevodsky, with the presence of some science journalists. During the trimester, the Bourbaki Seminar devoted an afternoon (June 21) to these themes, with talks of Thomas Hales and Thierry Coquand.

Shortly before, Coq has received the Software System Award 2013 from the Association for Computing Machinery (ACM). Hugo Herbelin is one of the recipients of this prize.

5.2. Proof-theoretical and effectful investigations

Participants: Pierre Boutillier, Guillaume Claret, Pierre-Louis Curien, Amina Doumane, Hugo Herbelin, Etienne Miquey, Ludovic Patey, Pierre-Marie Pédrot, Yann Régis-Gianas, Alexis Saurin.

5.2.1. Proving with side-effects

In 2012, 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. Étienne Miquey is currently working to get a presentation of this work in Curien-Herbelin's μ-$\widetilde{\mu}$-calculus, with the aim of getting in the end a CPS-translation. Such a translation would provide a strong argument of normalisation for the calculus, as well as a better undertanding of the mechanisms of the calculus, especially the side-effect part and the meaning of the existential quantifier restriction.

Hugo Herbelin and Danko Ilik carried on their work on the computational content of completeness proofs and in particular of the computational content of Gödel's completeness theorem. Hugo Herbelin presented their work at the workshop PSC 2014.

5.2.2. Reverse mathematics

Ludovic Patey studied with Laurent Bienvenu and Paul Shafer the provability strength of Ramsey-type versions of theorems like König's lemma. The corresponding paper is submitted to the Journal of Mathematical Logic. Ludovic Patey studied with Laurent Bienvenu the constructions of diagonal non-computable functions by probabilistic means. They submitted a paper to Information and Computation. Ludovic Patey worked on the existence of universal instances in reverse mathematics, and submitted a paper to Annals of Pure and Applied Logic. He worked on the relations between diagonal non-computability and Ramsey-type theorems and submitted a paper to the Archive for Mathematical Logic. He studied the links between the iterative forcing framework developed by Lerman, Solomon & Towsner and the notion of preservation of hyperimmunity and submitted a paper to Computability in Europe 2015.

5.2.3. Gödel's functional interpretation

Pierre-Marie Pédrot kept developing the proof-as-program interpretation of Gödel's Dialectica translation, as seen through the prism of classical realisability. This work was presented at TYPES 2014 and later published at LICS 2014 [26].

5.2.4. Logical foundations of call-by-need evaluation

Alexis Saurin and Pierre-Marie Pédrot developed a structured reconstruction of call-by-need based on linear head reduction which arose in the context of linear logic. This opens new directions both to extend call-by-need to control and to apply linear logic proof-theory (and particularly proof-nets) to call-by-need evaluation. This work was presented at JFLA 2014 [30] early 2014 and later expanded to the classical case, encompassing $\lambda\mu$-calculus.

5.2.5. Streams and classical logic

Alexis Saurin and Fanny He have been working on transfinite term rewriting in order to model stream calculi and their connections with lambda-calculi for classical logic. Their work gave rise to a presentation at the Workshop on Infinitary Rewriting that took place in Vienna last July as part of FLOC 2014.

5.2.6. Alternative syntaxes for proofs

Amina Doumane and Alexis Saurin, in a joint work with Marc Bagnol, studied the structure of several correctness criteria for linear logic proof-nets and could relate them through a new primitive notion of dependency. This work was first presented at JFLA 2014 [29] early 2014 and later at Structure and Deduction in Vienna as part of FLOC 2014. An expanded version has recently been accepted at FOSSACS 2015 [19].

5.3. Type theory and the foundations of Coq

Participants: Pierre Boutillier, Pierre-Louis Curien, Hugo Herbelin, Pierre-Marie Pédrot, Yann Régis-Gianas, Matthieu Sozeau, Arnaud Spiwack.

5.3.1. Description of type theory

Hugo Herbelin and Arnaud Spiwack completed and published their characterisation of the type constructions of Coq in terms of atomic constructions rather than their usual description as a monolithic scheme [23]. This work permitted both a more pedagogical presentation of Coq's type system, and a more tractable and composable mathematical model of Coq on which meta-properties can be stated and proved.

5.3.2. Models of type theory

Simplicial sets and their extensions as Kan complexes can serve as models of homotopy type theory. 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. This is in the process of being published in a special issue of MSCS on homotopy type theory [9].

The technique scales to provide type-theoretic constructions for arbitrary presheaves on Reedy categories, thus including simplicial sets.

5.3.3. Proof irrelevance, eta-rules

During his master's internship supervised by Matthieu Sozeau, Philipp Haselwarter studied a formulation of proof-irrelevance based on the rooster and the syntactic bracket presentation by Spiwack and Herbelin [23]. This resulted in a decomposition of the calculus cleanly showing the use of smashing and a better understanding of the restricted elimination rules of propositions. It also clearly shows that the inductive type for accessibility, used to justify general wellfounded definitions, can not be interpreted as a proof-irrelevant proposition in this calculus.

5.3.4. Unification

Matthieu Sozeau is continuing work in collaboration with Beta Ziliani (PhD at MPI-Saarbrücken) on formalising the unification algorithm used in Coq, which is central for working with advanced type inference features like Canonical Structures. This is the first precise formalisation of all the rules of unification including the ones used for canonical structure resolution. The presentation currently excludes some heuristics that were added on top of the core algorithm in Coq, until they can be studied more carefully. This work, part of B. Ziliani's thesis, was presented at the UNIF'14 workshop [28] and the Coq workshop in Vienna. A submission is in preparation.

5.3.5. Foundations and paradoxes

Arnaud Spiwack generalised previous works by Herman Geuvers and Hugo Herbelin to implement Hurkens's paradox of the impredicative system U^-. The resulting Coq implementation, which is completely independent from the impredicative features of Coq, generalises the two special cases which were previously used to prove negative results about impredicativity in Coq.

5.4. Homotopy of rewriting systems

Participants: Cyrille Chenavier, Pierre-Louis Curien, Yves Guiraud, Maxime Lucas, Philippe Malbos, Jovana Obradović.

5.4.1. Coherent presentations of Artin monoids

With Stéphane Gaussent (ICJ, Univ. de Saint-Étienne), Yves Guiraud and Philippe Malbos have used higher-dimensional rewriting methods for the study of Artin monoids, a class of monoids 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 monoid [65], and one by Deligne about the actions of Artin monoids on categories [47], both proved by geometrical 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 and to extend both theorems. This work will appear in Compositio Mathematica [18] and has been implemented in a Python library.

The next objective of this collaboration is to extend those results in every dimension, first to Artin monoids, then to Artin groups, with a view towards two well-known open problems in the field: the word problem of Artin groups and the so-called $K(\pi, 1)$ conjecture.

5.4.2. New methods for the computation of polygraphic resolutions

Maxime Lucas, supervised by Pierre-Louis Curien and Yves Guiraud, develops Squier's theory in the setting of cubical ω-categories. This will allow easier and more explicit computations of polygraphic resolutions than in the globular setting of [5], and the use of new effective methods such as the reversing algorithm from Garside theory [46].

Yves Guiraud currently collaborates with Patrick Dehornoy (Univ. de Caen) and Matthieu Picantin (LIAFA, Univ. Paris 7) to extend the constructions of [18] to other important families of monoids, such as the plactic monoid, the Chinese monoid and the dual braid monoids.

5.4.3. Higher-dimensional linear rewriting

Cyrille Chenavier, Pierre-Louis Curien, Yves Guiraud and Philippe Malbos investigate with Eric Hoffbeck (LAGA, Univ. Paris 13) and Samuel Mimram (LIX, École Polytechnique) the links between set-theoretic rewriting theory and the computational methods known in symbolic algebra, such as Gröbner bases [39]. This interaction is supported by the Focal project of the IDEX Sorbonne Paris Cité.

With Eric Hoffbeck (LAGA, Univ. Paris 13), Yves Guiraud and Philippe Malbos have introduced the setting of linear polygraphs to formalise a theory of linear rewriting (in the sense of linear algebra), generalising Gröbner bases. They have adapted to algebras the procedure of [5] that computes polygraphic resolutions from convergent presentations of monoids, with applications to the decision of an important homological property called Koszulness. This work is contained in [35] and it has been presented at IWC 2014 [31].

Cyrille Chenavier, supervised by Yves Guiraud and Philippe Malbos, explores the use of Berger's theory of reduction operators [38] to design new methods for the study of linear rewriting systems, and to promote the use of rewriting techniques in combinatorial algebra.

5.4.4. Homotopical and homological finiteness conditions

Yves Guiraud and Philippe Malbos have written a comprehensive introduction [36] on the links between higher-dimensional rewriting, the homotopical finiteness condition "finite derivation type" and the homological finiteness condition "FP_3", from the point of view of higher categories and polygraphs. The purpose of this work is to provide an introduction to the field, formulated in a contemporary language, and with new, more formal proofs of classical results.

5.4.5. Wiring structure of operads and operad-like structures

Building on recent ideas of Marcelo Fiore on the one hand, and of François Lamarche on the other hand, Pierre-Louis Curien and Jovana Obradović develop a syntactic approach, using some of the kit of Curien-Herbelin's duality of computation and its polarised versions of Munch and Curien, to the definition of various structures that have appeared in algebra under the names of operads, cyclic operads, dioperads, properads, modular and wheeled operads, permutads, etc.... These structures are defined in the literature in different flavours. We seek to formalise the proofs of equivalence between these different styles of definition, and to make these proofs modular, so as not to repeat them for each variation of the notion of operad. Preliminary results are being presented in January 2015 at the Mathematical Institute of the Academy of Sciences (Belgrade).

5.5. Coq as a functional programming language

Participants: Pierre Boutillier, Guillaume Claret, Lourdes Del Carmen González Huesca, Thibaut Girka, Hugo Herbelin, Pierre Letouzey, Matthias Puech, Yann Régis-Gianas, Matthieu Sozeau, Arnaud Spiwack.

5.5.1. Type classes and libraries

Type Classes are heavily used in the HoTT/Coq library (http://github.com/HoTT/coq) started by the Univalent Foundations program at the IAS, to which Matthieu Sozeau participated. To ease the development of this sophisticated library, Matthieu Sozeau implemented a number of extensions to type class resolution to make it more predictable and efficient. These are now part of the Coq 8.5 release.

5.5.2. Dependent pattern-matching

The dissertation of Pierre Boutillier presents and formalises a new algorithm to compile dependent pattern-matching into a chain of Coq case analyses. It avoids the use of the "uniqueness of identity proofs" axiom in more cases than the former proposal by McBride and McKinna.

5.5.3. Incrementality in proof languages

Lourdes del Carmen González Huesca and Yann Régis-Gianas developed a new variant of the differential lambda calculus that has two main features: (i) it is deterministic ; (ii) it is based on a notion of a first-class changes. A paper is in preparation.

5.5.4. Proofs of programs in Coq

In collaboration with David Mentre (Mitsubishi Rennes), Thibaut Girka and Yann Régis-Gianas worked on a certified generator for correlating programs. A correlating program is a program that represents the semantic difference between two (close) versions of a program by performing a static scheduling of their instructions. Performing an abstract interpretation on the correlating program provides a representation of the semantic differences between the two versions of a program. A paper is written and should be submitted soon.

5.5.5. Typed tactic language

In collaboration with Beta Ziliani (MPI) and Thomas Refis (master 2 student at University Paris Diderot), Yann Régis-Gianas starts the development of the version 2 of Mtac, a tactic language for Coq. Mtac is a DSL embedded in the Coq proof assistant. Roughly speaking, it allows Coq to be used as a tactic language for itself. With this work, Mtac 2 now includes first class goals. A paper is in preparation.

5.5.6. Tactic engine

Arnaud Spiwack joined the team for two months (Sept—Oct 2014) to finalise the integration and documentation of his re-engineering of Coq's interactive proof engine for the v8.5 version. The new perspective taken by this new engine is to shift the primary focus from how tactics (proof instructions) can modify goals (proof obligations) to focus on the way tactics compose. By making sure that composition of tactics has good mathematical properties, the new engine makes it possible to combine tactics in a more predictable and more powerful way. This new engine is also notable for the introduction of an abstract interface for tactics and tactic composition which makes it easy to augment tactics with new capabilities. The most notable such features are so-called dependent subgoals, which makes more fine-grained proofs possible and significantly improves the support for dependent types; and backtracking which gives the possibility to deploy very modular proof-search components. During his two months in the team, Arnaud Spiwack also added support for tracing tactic execution (Info), again taking advantage of his modular design.

5.5.7. Effectful programming

Guillaume Claret and Yann Régis-Gianas developed a compiler from a subset of OCaml with effects to Coq. Possible effects are the exceptions, the global references and the non-termination. Guillaume Claret and Yann Régis-Gianas developed Pluto, a concurrent HTTP web server written in Gallina. They worked on techniques to certify such interactive programs, formalising the reasoning by use cases. Use cases are proven correct giving a scenario, a typed schema of interactions between a program and an environment, built using the tactic mode of Coq as a symbolic debugger.

5.5.8. Libraries

Sébastien Hinderer and Pierre Letouzey contributed an extended library of lists. Pierre Letouzey contributed an extended library about Peano numbers, that takes advantages of the "Numbers" modular framework done earlier.

SUMO Project-Team

6. New Results

6.1. Highlights of the Year

We started our first industrial collaboration "Project P22" with Alstom Transport, in the context of a common laboratory between Inria and Alstom. The project started in March 2014 and tackles robustness issues and regulation in urban train systems. The second phase of the project will start in march 2015, for a duration of three years. Most of the researchers of Sumo are involved in this project.

6.2. Control and enforcement

6.2.1. Runtime enforcement of timed properties

Participants: Thierry Jéron, Hervé Marchand, Srinivas Pinisetty.

Runtime enforcement is a powerful technique to ensure that a running system satisfies some desired properties. Using an enforcement monitor, an (untrustworthy) input execution (in the form of a sequence of events) is modified into an output sequence that complies with a property. Over the last decade, runtime enforcement has been mainly studied in the context of untimed properties. The contributions [26] and [34] deal with runtime enforcement of timed properties by revisiting the founda-tions of runtime enforcement when time between events matters. We propose a new enforcement paradigm where enforcement mechanisms are time retardants: to produce a correct output sequence, additional delays are introduced between the events of the input sequence. We consider runtime enforcement of any regular timed property defined by a timed automaton. We prove the correctness of enforcement mechanisms and prove that they enjoy two usually expected features, revisited here in the context of timed properties. The first one is soundness meaning that the output sequences (eventually) satisfy the required property. The second one is transparency, meaning that input sequences are modified in a minimal way. We also introduce two new features, i) physical constraints that describe how a time retardant is physically constrained when delaying a sequence of timed events, and ii) optimality, meaning that output sequences are produced as soon as possible. To facilitate the adoption and implementation of enforcement mechanisms, we describe them at several complementary abstraction levels. Our enforcement mechanisms have been implemented and our experimental results demonstrate the feasibility of runtime enforcement in a timed context and the effectiveness of the mechanisms. Finally, in [33], we considered more practical applications. Indeed, in network security, RE monitors can detect and prevent Denial-of-Service attacks. In resource allocation, RE monitors can ensure fairness. Specifications in these domains express data-constraints over the received events where the timing between events matters. To formalize these requirements, we introduce Parameterized Timed Automata with Variables (PTAVs), an extension of Timed Automata (TAs) with internal and external variables. We then extend enforcement for TAs to enforcement for PTAVs for safety properties. We model requirements from the considered application domains and show how enforcement monitors can ensure system correctness w.r.t. these requirements.

6.2.2. Enforcing opacity

Participant: Hervé Marchand.

In [22], we have been interested in enforcing opacity of regular predicates on modal transition systems. Intuitively, a labelled transition system $\mathcal{T}$ partially observed by an attacker, and a regular predicate S over the runs of $\mathcal{T}$, enforcing opacity of the secret S in $\mathcal{T}$ means computing a supervisory controller K such that an attacker who observes a run of the controlled system $K \setminus \mathcal{T}$ cannot ascertain that the trace of this run belongs to S based on the knowledge of $\mathcal{T}$ and K. We lift the problem from a single labelled transition system $\mathcal{T}$ to the class of all labelled transition systems specified by a *Modal Transition System*$\mathcal{M}$. The lifted problem is to compute the maximally permissive controller K such that S is opaque in $K/\mathcal{T}$ for every labelled transition system $\mathcal{T}$ which is a model of $\mathcal{M}$. The situations of the attacker and of the controller are asymmetric: at run time, the attacker may fully know $\mathcal{T}$ and K whereas the controller knows only $\mathcal{M}$ and the sequence of actions executed so far by the unknown $\mathcal{T}$.

In [23], we provided a different solution by enforcing and validate ay runtime various notion of opacity. More specically, we studied how we can model-check, verify and enforce at system runtime, several levels of opacity. Besides existing notions of opacity, we also introduce K-step strong opacity, a more practical notion of opacity that provides a stronger level of confidentiality.

6.2.3. Discrete Controller Synthesis for Infinite State Systems with ReaX

Participants: Nicolas Berthier, Hervé Marchand.

This year, we investigated the control of infinite reactive synchronous systems modeled by arithmetic symbolic transition systems for safety properties handling numerical variable. We provide effective algorithms allowing to solve the safety control problem, and report on experiments based on ReaX, our tool implementing these algorithms [28].

6.3. Model expressivity and quantitative verification

6.3.1. Diagnosis

Participants: Nathalie Bertrand, Sébastien Chédor, Éric Fabre, Loïc Hélouët, Blaise Genest, Hervé Marchand, Christophe Morvan.

Diagnosis of a system consists in providing explanations to a supervisor from a partial observation of the system and a model of possible executions. This year, we have extended results on diagnosis algorithm from scenarios. Systems are modeled using High-level Message Sequence Charts (HMSCs), and the diagnosis is given as a new HMSC, which behaviors are all explanations of the partial observation. The results published this year are first an offline centralized diagnosis algorithm (a single process in a network collects an observation, and emits a diagnosis) that has then been extended to a decentralized version of this algorithm. This allows us to give a complete diagnosis framework for infinite state systems, with a strong emphasis on concurrency and causal ordering in behaviors. HMSC-based diagnosis showed nice properties w.r.t. compositionality. We have also considered solutions for online diagnosis from scenarios, but came to the conclusion that online solutions are memory consuming, and need too many restrictions to run with finite memory. The last contribution of this work is an application of diagnosis techniques to anomaly detection, that is a comparison of observation of the system with a model of usual behaviors to detect security attacks. This work has been published this year [24].

In [21] we have been interested in the analysis of discrete event systems under partial observation which is an important topic, with major applications such as the detection of information flow and the diagnosis of faulty behaviors. These questions have, mostly, not been addressed for classical models of recursive systems, such as pushdown systems and recursive state machines. In this paper, we consider recursive tile systems, which are recursive infinite systems generated by a finite collection of finite tiles, a simplified variant of deterministic graph grammars (slightly more general than pushdown systems). Since these systems are infinite-state in general 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 state-based opacity and diagnosability (off-line verification of opacity and diagnosability, and also run-time monitoring of these properties). For a decidable subclass of recursive tile systems, we also establish the decidability of the problems of state-based opacity and diagnosability.

In discrete event systems prone to unobservable faults, a diagnoser must eventually detect fault occurrences. The diagnosability problem consists in deciding whether such a diagnoser exists. We laid the foundations of diagnosis and predicatability for probabilistic systems represented by partially observed Markov chains (denoted pLTS) [32]. In particular, we studied different specifications of diagnosability and establish their relations both in finite and infinite pLTS. Then we analyzed the complexity of the diagnosability problem for finite pLTS: we showed that the polynomial time procedure proposed earlier is erroneous and that in fact for all considered specifications, the problem is PSPACE-complete. We also established tight bounds for the size of diagnosers. Afterwards we considered the dual notion of predictability which consists in predicting that in a safe run, fault will eventually occur. Predictability is easier than diagnosability: it is NLOGSPACE-complete. Yet the predictor synthesis is as hard as the diagnoser synthesis.

When a system is not diagnosable, the active diagnosis problem consists in controlling the system in order to ensure its diagnosability. In the same probabilistic setting, the active diagnosis problem consists in deciding whether there exists some observation-based strategy that makes the system diagnosable with probability one. We proved that this problem is EXPTIME-complete, and that the active diagnosis strategies are belief-based. The *safe* active diagnosis problem is similar, but aims at enforcing diagnosability while preserving a positive probability to non faulty runs, i.e. without enforcing the occurrence of a fault. We prove that this problem requires non belief-based strategies, and that it is undecidable. However, it belongs to NEXPTIME when restricted to belief-based strategies. Our work also refines the decidability/undecidability frontier for verification problems on partially observed Markov decision processes [30].

6.3.2. Probabilistic model checking

Participants: Nathalie Bertrand, Blaise Genest, Paulin Fournier.

In [16], we considered the verification of Markov chains against properties talking about distributions of probabilities. Even though a Markov chain is a very simple formalism, by discretizing in a finite number of classes the space of distributions through some symbolics, we proved that the language of trajectories of distribution (one for each initial distribution) is not regular in general, even with 3 states. We then proposed a parametrized algorithm which approximate what happens to infinity, such that each symbolic block in the approximate language is at most ϵ away from the concrete distribution.

Parameterized verification aims at validating a model of a system irrespective of the value of a parameter. This year, we studied verification problems for a model of network with the following characteristics: the number of entities is parametric, communication is performed through broadcast with adjacent neighbors, entities can change their internal state probabilistically and reconfiguration of the communication topology can happen at any time. The semantics of such a model is given in term of an infinite state system with both non deterministic and probabilistic choices. We are interested in qualitative problems like whether there exists an initial topology and a resolution of the non determinism such that a configuration exhibiting an error state is almost surely reached. We showed in [44] that all the qualitative reachability problems are decidable and some proofs are based on solving a 2 player game played on the graphs of a reconfigurable network with broadcast with parity and safety objectives.

On a different topic, we considered a control problem for stochastic sytems specified by timed automata with distributions over delays. In [29] we considered reachability objectives on such decision stochastic timed automata (DSTA). Given a reachability objective, the value 1 problem asks whether a target can be reached with probability arbitrarily close to 1. Simple examples show that the value can be 1 and yet no strategy ensures reaching the target with probability 1. In this paper, we prove that, the value 1 problem is decidable for single clock DSTA by non-trivial reduction to a simple almost-sure reachability problem on a finite Markov decision process. The ε-optimal strategies are involved: the precise probability distributions, even if they do not change the winning nature of a state, impact the timings at which ε-optimal strategies must change their decisions, and more surprisingly these timings cannot be chosen uniformly over the set of regions.

6.3.3. Distributed timed systems

Participants: Blaise Genest, Loïc Hélouët.

We have proposed and considered properties of a new timed variant of Petri nets [42], namely Timed Petri Nets with Urgency, that extend Timed Petri Nets with the main features of TPNs. Time Petri Nets (TPN) [52] and Timed Petri Nets [45] are two incomparable classes of concurrent models with timing constraints: urgency cannot be expressed using Timed Petri Nets, while TPNs can only keep track of a bounded number of continuous values (clocks). The work performed this year provides up to-our-knowledge the first decidability results for Petri Net variants combining time, urgency and unbounded places. We have obtained decidability of control-state reachability for the subclass of Timed Petri Nets with Urgency where urgency constraints can only be used on bounded places. By restricting this class to use a finite number of clocks, we have shows decidability of (marking) reachability. Formally, this class corresponds to TPNs under a new, yet natural, timed semantics where urgency constraints are restricted to bounded places. Further, under their original semantics, reachability for a more restricted class of TPNs is decidable.

6.3.4. Test Generation from Recursive Tile Systems

Participants: Sébastien Chédor, Christophe Morvan, Thierry Jéron.

In [20] we explore the generation of conformance test cases for *Recursive Tile Systems* 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. Test generation for this kind of infinite state labelled transition systems is seldom explored in the literature. The first part presents an off-line test generation algorithm for *Weighted* RTSs, a determinizable subclass of RTSs, and the second one, an on-line test generation algorithm for the full RTS model. Both algorithms use test purposes to guide test selection through targeted behaviours. Additionally, essential properties relating verdicts produced by generated test cases with both the soundness with respect to the specification, and the precision with respect to a test purpose, are proved.

6.4. Management of large distributed systems

6.4.1. Distributed optimal planning

Participant: Éric Fabre.

Planning problems consist in organizing actions in a system in order to reach one of some target states. The actions consume and produce resources, can of course take place concurrently, and may have costs. We have a collection of results addressing this problem in the setting of distributed systems. This takes the shape of a network of components, each one holding private actions operating over its own resources, and shared/synchronized actions that can only occur in agreement with its neighbors. The goal is to design in a distributed manner a tuple of local plans, one per component, such that their combination forms a consistent global plan of minimal cost.

Our previous solutions to this problem modeled components as weighted automata. In collaboration with Loig Jezequel (TU Munich) and Victor Khomenko (Univ. of Newcastle), we have extended this approach to the case of components modeled as safe Petri nets [50]. This allows one to benefit from the internal concurrency of actions within a component. Benchmarks have shown that this method can lead to significant time reductions to find feasible plans, in good cases. In the least favorable cases, performances are comparable to those obtained with components modeled as automata. The method does not apply to all situations however, as computations require to perform ϵ-reductions on Petri-nets (our work also contains a contribution to this difficult question). This work has been accepted by the ACM Transactions in Embedded Computing Systems, to appear in 2015.

6.5. Data driven systems

6.5.1. Web services

Participants: Blaise Genest, Loïc Hélouët.

This year, we considered transactional properties (ACID) for web services. In particular, we focused on the atomicity (A of ACID) property, obtained in case of a failure inside an atomic block through compensation of the executed actions of the block. To do so, logs need to be kept. We were interested in maintaining the maximal amount of privacy. We proposed modular algorithms [19] which maintain privacy between modules, with minimal information shared among modules, both in the logging and the compensation phases. Furthermore, each module logs a small number of information, such that the sum of all actions logged is guaranteed minimal. Last, modularity allows fast algorithms, as they need to consider only what happens in the module itself, and not the exact structure of its parent module nor of its sub-modules.

We also published results on our model of sessions systems [27]. This models allows for the modeling of distributed web-based systems that are running an arbitrary number of transactions among arbitrarily many participants. We have shown how simple restrictions can guarantee decidability of simple coverability properties, and then be used to detect violation of buisness rules such as conflict of interest, or a more complex property called the chinese wall.

We are currently considering new models that manage at the same time explicit workflows and structured data. This model can be seen as a combination of AXML [46] and Petri nets.

6.5.2. An Artifact-centric Process Model

Participants: Éric Badouel, Loïc Hélouët, Christophe Morvan.

In [37] we present a purely declarative approach to artifact-centric case management systems, and a decentralization scheme for this model. Each case is presented as a tree-like structure; nodes bear information that combines data and computations. Each node belongs to a given stakeholder, and semantic rules govern the evolution of the tree structure, as well as how data values derive from information stemming from the context of the node. Stakeholders communicate through asynchronous message passing without shared memory, enabling convenient distribution.

TEMPO Team

6. New Results

6.1. Highlights of the Year

The project was created.

6.2. Approximately Timed Simulation

Participants: Vania Joloboff, Shenpeng Wang.

Existing fast simulators such as SimSoC are Loosely Timed. They evaluate the time taken by instructions executed based on an average model. Typically, the clock value is increased by a constant K every N instructions. This is sufficient to test application software with time-outs or to synchronize multicore applications, but it cannot provide a reasonable performance estimate of the embedded software.

To obtain precise parformance estimate, a common practice is to run the software on Cycle Accurate simulators, which provides a performance measure absolutely correct, but take a very long time. This is becoming a bottleneck. In fact, in many cases the software developers need some performance estimate, but do not require cycle precision. The idea of "Approximately Timed" simulation is to provide a fast simulation that can be used by software developers, and yet provide performance estimate. The goal of approximately timed simulation is to provide estimates that are within a small margin error from the real hardware, but at a simulation speed that is an order of magnitude faster than a cycle accurate one.

It is possible to maintain fast simulation, (though slower than Loosely Timed) whereas predicting reasonably accurate performance. The challenge is to come up with an abstract model of the processor that does not simulate the processor at cycle level but simulate enough to measure elapsed time with good precision. The approach is the following: a modern processor in nominal mode executes at least one instruction per clock cycle. If it does not do so, it is because there is a delay, whether a cache miss, a pipe line stall, etc. If one can simulate enough of the system so that the cause of the delays can be reproduced in the simulation and the delays evaluated, although the details of the system are not reproduced exactly, then the delays estimate may be accurate enough to provide an acceptable margin error. Moreover some of these computation can be done only once, not for each iteration of a loop.

In our work, we are considering only the processor model and we rely upon TLM interface to the interconnect for peripheral access to provides us with timing delays. We estimate the performance by using static analysis of the application control flow graph combined with a minimum of dynamic computation in order to maintain a reasonable simulation speed. We have developed such a fast Approximately Timed ISS, that does not fully simulate the hardware, yet provides good precision estimates, and does not use stastistical methods. Our approach consists in developing a higher abstraction model of the processor (than the CA models) that still executes instructions using fast SystemC/TLM code, but in parallel maintains some architecture state to measure the delays introduces by cache misses and pipe line stalls, although the pipe line is not really simulated. This work will be published in 2015 in volume 68 of the WIT Transactions on Information and Communication Technologies (ISBN 978-1-78466-054-3) [10].

6.3. Automated generation of simulator

Participants: Vania Joloboff, Shengpeng Liu.

Developing a simulator for a complete processor represents a lot of work when it is manual coding, and it is error-prone. Several efforts have been made to generate partly or entirely simulators. The dominant approach in the past years has been to use a high level description language of the processor and to generate code with the language compiler, such as LISA [18], MIMOLA [15], EXPRESSION-ADL [17]. But still, the architecture is described manually into the high level language. It is interesting to explore new architectures, but has the same issues as manual coding for simulation of commercial off-the-shelf processors such as ARM an Power architectures. Of course this approach only makes sense if the vendor has at least some semi-formal description of the architecture, which is not the case for Intel, but is the case for ARM, PowerPC and SH.

In order to automatically generate simulators from the vendor specification, we have initiated a new approach: generating the simulator from the specification of the hardware vendor as available from their web site as .pdf document. After a relatively successful initiative using ad-hoc tools, we wanted to pursue this work in a more robust and industrial context, using XML to generate an XML model of the instruction set from the vendor specification in .pdf, formalize some XML model transformations and finally generate directly the simulator code in C++. In addition, we wanted the translator to be architecture independent, not making any assumption during the translation process. However, this work is hitting difficult issues due to the fact that the vendor specification is incomplete and we have to do more manual architecture specific complements to the specification that we anticipated, which seriously weakens the project objective.

6.4. Automated Test

Participants: Mingsong Chen, Haifeng Gu, Xinqian Zhang.

Under the increasing complexity together with the time-to-market pressure, functional validation is becoming a major bottleneck of smart applications running on mobile platforms (e.g., Android, iOS).Unlike traditional software, smartphone applications are reactive and GUI (Graphical User Interface) intensive. The execution of smartphone applications heavily relies on the interactions with users. Manual GUI testing is extremely slow and unacceptably expensive in practice. However, the lack of formal models of user behaviors in the design phase hinders the automation of GUI testing (i.e., test case generation and test evaluation). While thorough test efforts are required to ensure the consistency between user behavior specifications and GUI implementations, few of existing testing approaches can automatically utilize the design phase information to test complex smartphone applications. Based on UML activity diagrams, we propose an automated GUI testing framework called ADAutomation, which supports user behavior modeling, GUI test case generation, and post-test analysis and debugging. The experiments using two industrial smartphone virtual prototypes demonstrate that our approach [16] can not only drastically reduce overall testing time, but also showed to improve the quality of designs.

6.5. SAT based bounded model checking

Participants: Mingsong Chen, Haifeng Gu, Xinqian Zhang.

SAT-based Bounded Model Checking (BMC) is promising for automated generation of directed tests. Due to the state space explosion problem, SAT-based BMC is unsuitable to handle complex properties with large SAT instances or large bounds. In this work, we propose a framework to automatically scale down the SAT falsification complexity by utilizing the decision ordering based learning from decomposed sub-properties. Our framework makes three important contributions: i) it proposes learning-oriented decomposition techniques for complex property falsification, ii) it proposes an efficient approach to accelerate the complex property falsification using the learning from decomposed sub-properties, and iii) it combines the advantages of both property decomposition and property clustering to reduce the overall test generation time. The experimental results [11] using both software and hardware benchmarks demonstrate the effectiveness of our framework.

TOCCATA Project-Team

6. New Results

6.1. Highlights of the Year

- The ACM Software System Award 2013 was given, during a ceremony in June 2014 in San Francisco, to the Coq proof assistant (http://awards.acm.org/software_system/). The prestigious ACM price was previously awarded to the LLVM compiler infrastructure (2012) and to the Eclipse IDE (2011). Among the 9 recipients of the 2013 award are Christine Paulin and Jean-Christophe Filliâtre, from the Toccata team.
- The *Concours Castor informatique* (http://castor-informatique.fr/) had an even larger success than in the previous years. In November 2014, more than 228,000 teenagers from over 1500 schools participated and solved the interactive tasks of the contest. Arthur Charguéraud and Sylvie Boldo, from the Toccata team, significantly contributed to the prepation of the tasks and to the organization of the contest.

6.2. Deductive Verification

- J.-C. Filliâtre, L. Gondelman, and A. Paskevich have formalized the notion of ghost code implemented in *Why3*, in a paper *The Spirit of Ghost Code* [35] presented at CAV 2014. This is an outcome of L. Gondelman's M2 internship (spring/summer 2013).
- M. Clochard published at the POPL conference a paper presenting a work done during an internship at Rice University (Houston, TX, USA) with S. Chaudhuri and A. Solar-Lezama [29]. It is a new technique for parameter synthesis under boolean and quantitative objectives. The input to the technique is a "sketch"—a program with missing numerical parameters—and a probabilistic assumption about the program's inputs. The goal is to automatically synthesize values for the parameters such that the resulting program satisfies: (1) a boolean specification, which states that the program must meet certain assertions, and (2) a quantitative specification, which assigns a real valued rating to every program and which the synthesizer is expected to optimize.
- J.-C. Filliâtre, C. Marché, and A. Paskevich, together with F. Bobot (CEA LIST), took part in the VerifyThis program verification competition, held at the 18th FM symposium in August 2012. They used *Why3* to solve three challenges (which can be found at http://fm2012.verifythis.org/challenges/), and their solutions have been published in a special issue of the journal *Software Tools for Technology Transfer* [16].
- M. Clochard developed, using *Why3*, verified implementations of several data structures, including random-access lists and ordered maps. These are derived from a common parametric implementation of self-balancing binary trees in the style of Adelson-Velskii and Landis trees (so-called AVLs). This work appeared at the VSTTE conference [30]. Its originality relies on the genericity of the specifications and the code, and the very high level of proof automation. Such a case study is aimed at illustrating the capabilities of *Why3* for designing certified libraries. Development is available from our gallery at http://toccata.lri.fr/gallery/avl.fr.html.
- S. Conchon and A. Mebsout have extended the core algorithm of the Cubicle model checker with a mechanism for inferring invariants. This new algorithm, called BRAB, is able to automatically infer invariants strong enough to prove industrial cache coherence protocols. BRAB computes over-approximations of backward reachable states that are checked to be unreachable in a finite instance of the system. These approximations (candidate invariants) are then model-checked together with the original safety properties. Completeness of the approach is ensured by a mechanism for backtracking on spurious traces introduced by too coarse approximations. Details can be found in A. Mebsout's PhD thesis [15].

- A. Charguéraud extended his tool CFML to support, in addition to the verification of the full functional correctness of a piece of code, the verification of the asymptotic complexity of the code. Even though it had been previously established that, in theory, amortized analysis can be explained as the manipulation of *time credits*, and that time credits can be encoded as resources in Separation Logic, CFML is the first practical tool to support the formal verification of amortized analyses for arbitrarily-complex pieces of code. The *time-credit* extension to CFML was put to practice to verify dynamic arrays (Julien Grangier's internship), and to verify a *chunked sequence* data structure [26], particularly challenging due to its use of Tarjan's data structural bootstrapping technique. The latter piece of work was presented in July at the workshop *Semantics of proofs and certified mathematics*, which took place at the Institut Henri Pointcaré. A paper describing the time-credit extension to CFML is under preparation.

6.3. Floating-Point and Numerical Programs

- C. Marché published in the *Science of Computer Programming* journal [22] a detailed description of an industrial research initially conducted in the context of the U3CAT project (ended in 2012) on static analysis of critical C code. The code involves floating-point computations on quaternions that should be of norm 1. Because of the round-off errors, a drift of this norm is observed over time. In this work a bound on this drift is determined and formally proved correct, using *Frama-C*, *Jessie* and *Why3*. Proofs are done using automated provers and in a few complex cases the Coq proof assistant. The published version is up to date with the recent versions of those tools, and the development is available on our gallery at http://toccata.lri.fr/gallery/quat.en.html
- S. Boldo, C. Lelay, and G. Melquiond worked on the Coquelicot library, designed to be a user-friendly Coq library about real analysis. An easier way of writing formulas and theorem statements is achieved by relying on total functions in place of dependent types for limits, derivatives, integrals, power series, and so on. To help with the proof process, the library comes with a comprehensive set of theorems and some automation. We have exercised the library on several use cases: in an exam at university entry level, for the definitions and properties of Bessel functions, and for the solution of the one-dimensional wave equation. These results are published in the journal *Mathematics in Computer Science* [19].
- S. Boldo and G. Melquiond, with J.-H. Jourdan and X. Leroy (Gallium team, Inria Paris - Rocquencourt) extended the CompCert compiler to get the first formally verified C compiler that provably preserves the semantics of floating-point programs This work, published in the *Journal of Automated Reasoning* [18], also covers the formalization of numerous algorithms of conversion between integers and floating-point numbers.
- S. Boldo, C. Lelay, and G. Melquiond, have conducted a survey on the formalization of real arithmetic and real analysis in various proof systems. This work, published in the journal *Mathematical Structures in Computer Science* [20], details the axioms, definitions, theorems, and methods of automation, available in these systems.
- É. Martin-Dorel and G. Melquiond worked on integrating the CoqInterval and CoqApprox libraries into a single package. The CoqApprox library is dedicated to computing verified Taylor models of univariate functions so as to compute approximation errors. The CoqInterval library reuses this work to automatically prove bounds on real-valued expressions. A large formalization effort took place during this work, so as to get rid of all the holes remaining in the formal proofs of CoqInterval. It was also the chance to perform a comparison between numerous decision procedures dedicated to proving nonlinear inequalities involving elementary functions. A report is available [43].
- S. Boldo, J.-C. Filliâtre, and G. Melquiond, with F. Clément and P. Weis (POMDAPI team, Inria Paris - Rocquencourt), and M. Mayero (LIPN), completed the formal proof of a numerical analysis program: the second-order centered finite-difference scheme for the one-dimensional acoustic wave. This proof was published with a focus towards numerical analysts, in the journal *Computers and Mathematics with Applications* [17].

- P. Roux formalized the influence of double rounding on the accuracy of floating-point arithmetic operators. In particular, this includes all the corner cases that were ignored from Figueroa's original pen-and-paper proof. Results appeared in the *Journal of Formalized Reasoning* [24].
- P. Roux formalized a theory of numerical analysis for bounding the round-off errors of a floating-point algorithm. This approach was applied to the formal verification of a program for checking that a matrix is semi-definite positive. The challenge here is that testing semi-definiteness involves algebraic number computations, yet it needs to be implemented using only approximate floating-point operations. A report is available [45].

6.4. Automated Reasoning

- In the context of the BWare project, aiming at using *Why3* and Alt-Ergo for discharging proof obligations generated by Atelier B, we made progress into several directions. New drivers have been designed for *Why3*, in order to use new back-end provers Zenon modulo and iProver modulo. A notion of rewrite rule was introduced into *Why3*, and a transformation for simplifying goals before sending them to back-end provers was designed. Intermediate results obtained so far in the project were presented both at the French conference AFADL [38] and at the international conference on Abstract State Machines, Alloy, B, VDM, and Z [34].

  On the side of Alt-Ergo, recent developments have been made to efficiently discharge proof obligations generated by Atelier B. This includes a new plugin architecture to facilitate experiments with different SAT engines, new heuristics to handle quantified formulas, and important modifications in its internal data structures to boost performances of core decision procedures. Benchmarks realized on more than 10,000 proof obligations generated from industrial B projects show significant improvements [33].
- C. Dross defended her PhD thesis in April 2014 [14], on the topic of automated reasoning modulo theories, and in particular the handling of quantifiers in the SMT approach. The main results of the thesis are: (1) a formal semantics of the notion of *triggers* typically used to control quantifier instantiation in SMT solvers, (2) a general setting to show how a first-order axiomatization with triggers can be proved correct, complete, and terminating, and (3) an extended DPLL(T) algorithm to integrate a first-order axiomatization with triggers as a decision procedure for the theory it defines. Significant case studies were conducted on examples coming from SPARK programs, and on the benchmarks on B set theory constructed within the BWare project.

6.5. Certification of Languages, Tools and Systems

- M. Clochard, C. Marché, and A. Paskevich developed a general setting for developing programs involving binders, using *Why3*. This approach was successfully validated on two case studies: a verified implementation of untyped lambda-calculus and a verified tableaux-based theorem prover. This work was presented at the PLPV conference in January 2014 [32].
- M. Clochard, J.-C. Filliâtre, C. Marché, and A. Paskevich developed a case study on the formalization of semantics of programming languages using *Why3*. This case study aimed at illustrating recent improvements of *Why3* regarding the support for higher-order logic features in the input logic of *Why3*, and how these are encoded into first-order logic, so that goals can be discharged by automated provers. This case study also illustrates how reasoning by induction can be done without need for interactive proofs, via the use of *lemma functions*. This works was presented at the VSTTE conference [31].
- M. Clochard and L. Gondelman developed a formalization of a simple compiler in *Why3*. It compiles a simple imperative language into assembler instructions for a stack machine. This case study was inspired by a similar example developed using Coq and interactive theorem proving. The aim is to improve significantly the degree of automation in the proofs. This is achieved by the formalization of a Hoare logic and a Weakest Precondition Calculus on assembly programs, so that the correctness of compilation is seen as a formal specification of the assembly instructions generated. This work conducted in 2014 will be presented at the JFLA conference in January 2015 [75].

- S. Dumbrava and É. Contejean, with V. Benzaken (VALS team, at LRI) proposed a *Coq* formalization of the relational data model which underlies relational database systems. More precisely, they have presented and formalized the data definition part of the model including integrity constraints. They have modelled two different query language formalisms: relational algebra and conjunctive queries. They also present logical query optimization and prove the main "database theorems": algebraic equivalences, the homomorphism theorem and conjunctive query minimization. This work has been published at ESOP 2014 [27].
- A. Charguéraud, together with the other members of the *JsCert* team have developed this year the first complete formalization of the semantics of the JavaScript programming language. 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 (see http://jscert.org). The formalization consists of a set of inductive rules translating the prose from the *ECMAScript Language Specification, version 5*, using the pretty-big-step semantics [74]. These rules can be used to formally reason about program behaviors or to establish the correctness of program transformations. In addition to the inductive rules, a reference interpreter has been proved correct. This interpreter may be used to run actual JavaScript program following the rules of the formal semantics. It has been used in particular to validate the formal semantics against official JavaScript test suites. The formalization of JavaScript has been published at POPL [28].

6.6. Miscellaneous

A. Charguéraud worked together with Umut Acar and Mike Rainey, as part of the ERC project *DeepSea*, on the developement of efficient data structures and algorithms targeting modern, shared memory multicore architectures. Two major results were obtained this year.

The first result is a sequence data structure that provides amortized constant-time access at the two ends, and logarithmic time concatenation and splitting at arbitrary positions. These operations are essential for programming efficient computation in the fork-join model. Compared with prior work, this novel sequence data structure achieves excellent constant factors, allowing it to be used as a replacement for traditional, non-splittable sequence data structures. This data structure, called *chunked sequence* due to its use of chunks (fixed-capacity arrays), has been implemented both in C++ and in OCaml. It is described in a paper published at ESA [26].

Another result by A. Charguéraud and his co-authors is the development of fast and robust parallel graph traversal algorithms, more precisely for parallel BFS and parallel DFS. The new algorithms leverage the aforementioned sequence data structure for representing the set of edges remaining to be visited. In particular, it uses the split operation for balancing the edges among the several processors involved in the computation. Compared with prior work, these new algorithms are designed to be efficient not just for particular classes of graphs, but for all input graphs. This work has not yet been published, however it is described in details in a technical report [40]. Note that these two graph algorithms, which involve nontrivial use of concurrent data structures, will be very interesting targets for formal verification.

VERIDIS Project-Team

6. New Results

6.1. Highlights of the Year

The veriT solver (section 5.1) participated in the SMT competition 2014, part of the Vienna Summer Of Logic Olympic Games, and received the gold medal for the SMT category.

6.2. Automated and Interactive Theorem Proving

Participants: Pascal Fontaine, Marek Košta, Manuel Lamotte Schubert, Stephan Merz, Thomas Sturm, Hernán Pablo Vanzetto, Uwe Waldmann, Daniel Wand, Christoph Weidenbach.

6.2.1. Combination of Satisfiability Procedures

Joint work with Christophe Ringeissen from the CASSIS project-team at Inria Nancy Grand Est, and Paula Chocron, a student at the University of Buenos Aires.

A satisfiability problem is often expressed in a combination of theories, and a natural approach consists in solving the problem by combining the satisfiability procedures available for the component theories. This is the purpose of the combination method introduced by Nelson and Oppen. However, in its initial presentation, the Nelson-Oppen combination method requires the theories to be signature-disjoint and stably infinite (to ensure the existence of an infinite model). The design of a generic combination method for non-disjoint unions of theories is clearly a hard task but it is worth exploring simple non-disjoint combinations that appear frequently in verification. An example is the case of shared sets, where sets are represented by unary predicates. Another example is the case of bridging functions between data structures and a target theory (e.g., a fragment of arithmetic).

The notion of gentle theory has been introduced in the last few years as one solution to go beyond the restriction of stable infiniteness, in the case of disjoint theories. In [26], [43], we adapt the notion of gentle theory to the non-disjoint combination of theories sharing only unary predicates, constants, and equality. As in the disjoint case, combining two theories, one of them being gentle, requires some minor assumptions on the other one. We show that major classes of theories, i.e., Loewenheim and Bernays-Schoenfinkel-Ramsey, satisfy the appropriate notion of gentleness introduced for this particular non-disjoint combination framework.

We have also considered particular non-disjoint unions of theories connected via bridging functions [27]. We present a combination procedure which is proved correct for the theory of absolutely free data structures. We consider the problem of adapting the combination procedure to obtain a satisfiability procedure for the standard interpretations of the data structure. We present an enumeration procedure that allows us to revisit the case of lists with length.

6.2.2. Type Synthesis for Set-Theoretic Proof Obligations

TLA$^+$ is a language for the formal specification of systems and algorithms whose first-order kernel is a variant of untyped Zermelo-Fraenkel set theory. Typical proof obligations that arise during the verification of TLA$^+$ specifications mix reasoning about sets, functions, arithmetic, tuples, and records. One of the challenges in designing an efficient encoding of TLA$^+$ proof obligations for the input languages of first-order automatic theorem provers or SMT solvers is to synthesize appropriate sorts for the terms appearing in a proof obligation, matching the type system of the target prover. We base this synthesis on the detection of "typing hypotheses" present in the proof obligations and then propagate this information throughout the entire formula. An initial type system [53] similar to the multi-sorted discipline underlying SMT-lib was not expressive enough for representing constraints such as domain conditions for function applications. We therefore developed a more expressive type system that includes dependent types, predicate types, and subtyping. Type synthesis in this system is no longer decidable but generates constraints that are submitted to SMT solvers during type

reconstruction. When the constraints are valid, the translation of the formula becomes simpler, and checking it becomes correspondingly more efficient. When type construction does not succeed, the translator locally falls back to a sound, but inefficient "untyped" encoding where interpreted sorts such as integers are injected into the SMT sort representing TLA$^+$ values. In practice, this approach is found to behave significantly better than the original type system, and it extends easily to ATP proof backends. The results have been published at NFM 2014 [29], full details appear in Vanzetto's PhD thesis [11].

6.2.3. Syntactic Abstractions in First-Order Modal Logics

Joint work with Damien Doligez, Jael Kriener, Leslie Lamport, and Tomer Libal within the TLA$^+$ project at the MSR-Inria Joint Centre.

TLA$^+$ proofs mix first-order and temporal logics, and few (semi-)automatic proof tools support such languages. Moreover, natural deduction and sequent calculi, which are standard underpinnings for reasoning in first-order logic, do not extend smoothly to modal or temporal logics, due to the presence of implicit parameters designating the current point of evaluation. We design a syntactic abstraction method for obtaining pure first-order, respectively propositional modal or temporal, formulas from proof obligations in first-order modal or temporal logic, and prove the soundness of this "coalescing" technique. The resulting formulas can be passed to existing automatic provers or decision procedures for first-order logic (possibly with theory support), respectively for propositional modal and temporal logic. The method is complete for proving safety properties of specifications. This work was presented at the workshop on Automated Reasoning in Quantified Non-Classical Logic organized as part of Vienna Summer of Logic [33], and it has been implemented within TLAPS (section 5.2).

6.2.4. Satisfiability of Propositional Modal Logics via SMT Solving

Joint work with Carlos Areces from the National University of Córdoba, Argentina, and Clément Herouard, a student at ENS Rennes.

Modal logics extend classical propositional logic, and they are robustly decidable. Most existing decision procedures for modal logics are based on tableau constructions. Within our ongoing cooperation with members of the National University of Córdoba supported by the MEALS and MISMT projects (sections 8.3 and 8.4), we are investigating the design of decision procedures based on adding custom instantiation rules to standard SAT and SMT solvers. Our constructions build upon the well-known standard translation of modal logics to the guarded fragment of first-order logic. The idea is to let the solver maintain an abstraction of the quantified formulas, together with corresponding models. The abstraction is refined by lazily instantiating quantifiers, until either it is found to be unsatisfiable or no new instantiations need to be considered. We prove the soundness, completeness, and termination of the procedure for basic modal logic and several extensions. In particular, a smooth extension to hybrid logic makes use of the decision procedures for equality built into SMT solvers, yielding surprisingly simple correctness proofs. A presentation of this work has been accepted for publication in 2015.

6.2.5. First-Order Extensions to Support Higher-Order Reasoning

In contrast to higher-order logic, first-order logic provides automation and completeness. In order to increase the success rate of first-order proof procedures on translations of higher-order proof obligations, we developed two extensions to first-order logic:

- a polymorphic type system and
- declarations for inductive data types.

While the former can be seen as "just some kind of complication" to standard first-order reasoning procedures, the latter is an extension beyond first-order logic. We have shown how to keep first-order completeness in the presence of inductive data types while making use of the declarations for inferences and reductions that cannot be justified at the first-order level. The result is a superposition calculus extended with induction that shows impressive performance on standard benchmark sets when compared to existing approaches.

6.2.6. Decidability of First-Order Recursive Clause Sets

Recursion is a necessary source for first-order undecidability of clause sets. If there are no cyclic, i.e., recursive definitions of predicates in such a clause set, (ordered) resolution terminates, showing decidability. In this work we present the first characterization of recursive clause sets enabling non-constant function symbols and depth increasing clauses but still preserving decidability. For this class called BDI (Bounded Depth Increase) we present a specialized superposition calculus. This work has been published in the Journal of Logic and Computation [18].

6.2.7. Finite Quantification in Hierarchic Theorem Proving

Joint work with Peter Baumgartner and Joshua Bax from NICTA, Canberra, Australia.

Many applications of automated deduction require reasoning in first-order logic modulo background theories, in particular some form of integer arithmetic. A major unsolved research challenge is to design theorem provers that are "reasonably complete" even in the presence of free function symbols ranging into a background theory sort. For the case when all variables occurring below such function symbols are quantified over a finite subset of their domains, we have developed and implemented a non-naive decision procedure for extended theories on top of a black-box decision procedure for the EA-fragment of the background theory. In its core, it employs a model-guided instantiation strategy for obtaining pure background formulas that are equi-satisfiable with the original formula. Unlike traditional finite model finders, it avoids exhaustive instantiation and, hence, is expected to scale better with the size of the domains [25].

6.2.8. Developing Learning Strategies for Virtual Substitution

Joint work with Konstantin Korovin from the University of Manchester, UK.

During the past twenty years there have been a number of successful applications of real quantifier elimination methods based on virtual substitution. On the other hand, recently there has been considerable progress in (linear and non-linear) real arithmetic SMT-solving triggered by the idea to adopt from Boolean SAT-solving conflict analysis and learning techniques. In this work we do the first steps towards combining these two lines of research.

We consider linear real arithmetic SMT-solving. Inspired by related work for the Fourier-Motzkin method, we develop learning strategies for linear virtual substitution. For the first time, we formalize a virtual substitution-based quantifier elimination method—with and without our learning strategies—as formal calculi in the style of abstract DPLL [55]. We prove soundness and completeness for these calculi. Some standard linear programming benchmarks computed with an experimental implementation of our calculi show that the novel learning techniques combined with linear virtual substitution give rise to considerable speedups. Our implementation is part of the Reduce package Redlog, which is open-source and freely available.

This work gave rise to a publication at the CASC 2014 international workshop [28].

6.2.9. Efficient Cell Construction in Cylindrical Algebraic Decomposition

Joint work with Christopher W. Brown from the United States Naval Academy.

In their 2012 paper, de Moura and Jovanović [51] give a novel procedure for non-linear real SMT solving. The procedure uses DPLL-style techniques to search for a satisfying assignment. In case of a conflict, Cylindrical Algebraic Decomposition (CAD) is used to guide the search away from the conflicting state: On the basis of one conflicting point, the procedure learns to avoid in the future an entire CAD cell containing that point. The crucial part of this "model-based" approach is a function realizing this cell learning. Unfortunately, it is the main computational bottleneck of the whole procedure.

In 2014, we improved our cell learning procedure developed in 2013 by further theoretical investigation, which led to optimizations of the cell construction algorithm. This work gave rise to a publication in the Journal of Symbolic Computation [14].

In this publication we present an algorithm for the cell construction problem. Given a point and a set of polynomials, the algorithm constructs a single cylindrical cell containing the point, such that the polynomials are sign-invariant in the constructed cell. To represent a single cylindrical cell, a novel data structure is introduced. The algorithm, which is based on McCallum's projection operator, works with this representation and proceeds incrementally: First a cell representing the whole real space is constructed, and then refinement with respect to a single input polynomial is done to ensure the sign-invariance of this polynomial in the refined cell. We prove that our algorithm is correct and efficient in the following sense: First, the set of polynomials computed by our algorithm is a subset of the set constructed by the "model-based" approach, and second, the cell constructed by our algorithm is bigger than the cell constructed by the "model-based" approach.

6.3. Formal Methods for Developing Algorithms and Systems

Participants: Manamiary Andriamiarina, Jingshu Chen, Marie Duflot-Kremer, Dominique Méry, Stephan Merz.

6.3.1. Incremental Development of Distributed Algorithms

Joint work with Mohammed Mosbah and Mohammed Tounsi from the LABRI laboratory in Bordeaux, France, and with Neeraj Kumar Singh from the Department of Computing and Software, McMaster University, Hamilton, Canada.

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 work helps formalizing pre-existing algorithms, developing new algorithms, as well as developing models for distributed systems.

Our research was initially supported by the ANR project RIMEL (see http://rimel.loria.fr). More concretely, we aim at an integration of the correct-by-construction refinement-based approach into the *local computation* programming model underlying the VISIDIA toolkit developed at LABRI for designing distributed algorithms expressed as a set of rewriting rules over graph structures.

In particular, we show how state-based models can be developed for specific problems [22] and how they can be simply reused by controlling the composition of state-based models through the refinement relationship. Traditionally, distributed algorithms are supposed to run on a fixed network, whereas we consider a network with a changing topology.

The contribution is related to the development of proof-based patterns providing effective help to the developer of formal models of applications [24], [12], [42]. Our patterns simplify the development of distributed systems using refinement and temporal logic.

6.3.2. Modeling Medical Devices

Formal modelling techniques and tools [30] 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. 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.

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. The analysis of guidelines using formal methods is a promising approach for improving them.

In [32], we give the semantics of refinement diagrams that are used in a refinement-based methodology for complex medical systems design, which possesses all the required key features. A refinement-based approach relying on formal verification, model validation using a model-checker, and refinement charts is proposed in this methodology for designing a high-confidence medical device. We show the effectiveness of this methodology for the design of a cardiac pacemaker system. Moreover, we organized a Dagstuhl seminar on the Pacemaker Challenge [20].

6.3.3. Analysis of Real-Time Concurrent Programs

Joint work with Nadezhda Baklanova, Jan-Georg Smaus, Wilmer Ricciotti, and Martin Strecker at IRIT Toulouse, France, and master student Jorge Ibarra Delgado, funded by the Airbus Foundation (see also section 7.1).

We investigate techniques for the formal verification of multi-threaded real-time programs. We assume that programs contain annotations that indicate the times for executing basic blocks, and that these annotations are enforced by the execution platform. Inspired by Safety-Critical Java [49], our partners in Toulouse developed a formal semantics for a fragment of Java in Isabelle/HOL. We designed techniques for formally ensuring the absence of concurrent accesses to shared resources in bounded-length executions of such programs. Specifically, we generate constraints that characterize the possible execution orders of the program, and then invoke an SMT solver in order to verify that no execution violates precedence constraints that ensure absence of conflicts. In the case where such an execution exists, we obtain a trace that exhibits the access conflict. Our technique has been implemented prototypically, and appears to scale much better than a previous analysis based on an encoding of programs as timed automata. The results have been published at AVoCS 2014 [15].

During his internship within the first year of the Erasmus Mundus master program on Dependable Software Systems, Jorge Ibarra Delgado investigated the possibility of adapting the JOP toolset for Safety-Critical Java, and in particular its Worst-Case Execution Time (WCET) analyzer, for obtaining suitable annotations for basic blocks.

6.3.4. Bounding Message Length in Attacks Against Security Protocols

Joint work with Myrto Arapinis from the University of Glasgow, UK.

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. We have shown that, under a syntactic and reasonable condition of "well-formedness" on the protocol, we can get rid of the infinitely branching part. A journal version of this result, extending the set of security properties to which it is applicable and that particular includes authentication properties, has been published in Information and Computation [13].

6.3.5. Evaluating and Verifying Probabilistic Systems

Joint work with colleagues at ENS Cachan and University Paris Est Créteil.

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 fulfills 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 fall outside the scope of model checking. In particular, it is often of interest not to tell wether a property is satisfied but how well the system performs with respect to a certain measure. We have designed

a statistical tool for tackling both performance and verification issues. Following several conference talks, two journal papers have been submitted. The first one presents the approach in details with a few illustrative applications. The second one focuses on biological applications, and more precisely the use of statistical model checking to detect and measure several indicators of oscillating biological systems.

CARTE Project-Team

6. New Results

6.1. Highlights of the Year

Our team made remarkable progress into the understanding of complexity of higher-order functionals. While a robust class of computable functionals exists at any finite type built from $\mathbb{N}$ and $\rightarrow$ (the Kleene-Kreisel functionals), no satisfying complexity classes had been defined so far, except the class BFF of Basic Feasible Functionals. However that class is not a complexity class in the usual sense and does not offer the possibility to define space complexity or non-deterministic time complexity. In his PhD Hugo Férée has developed a non-trivial notion of size for higher-order functionals using game semantics and he has defined a notion of polynomial-time computable functional including BFF but behaving more satisfactorily in several ways. A paper in preparation will gather these results.

6.2. Malware Detection and Program Analysis

- **Complexity Information Flow in a Multi-threaded Imperative Language.** Program resource analysis using tiering based type system has been extended to analyze the time consumed by multi-threaded imperative programs with a shared global memory, which delineates a class of safe multi-threaded programs. In this work presented at TAMC'14 (Theory and Applications of Models of Computation) [22] Jean-Yves Marion and Romain Péchoux have demonstrated that a safe multi-threaded program runs in polynomial time if (i) it is strongly terminating w.r.t. a non-deterministic scheduling policy or (ii) it terminates w.r.t. a deterministic and quiet scheduling policy. As a consequence, we obtain a characterization of the set of polynomial time functions. As far as we know,this is the first characterization by a type system of polynomial time multi-threaded programs
- **A Categorical Treatment of Malicious Behavioral Obfuscation.** In this work presented at TAMC'14 (Theory and Applications of Models of Computation) [23] Romain Péchoux and Thanh Dinh Ta consider malicious behavioral obfuscation through the use of a new abstract model for process and kernel interactions based on monoidal categories. In this model, program observations are considered to be finite lists of system call invocations. In a first step, the authors have shown how malicious behaviors can be obfuscated by simulating the observations of benign programs. In a second step, they have shown how to generate such malicious behaviors through a technique called path replaying and they have extended the class of captured malwares by using some algorithmic transformations on morphisms graphical representation.
- **Malware Message Classification by Dynamic Analysis.** Guillaume Bonfante, Jean-Yves Marion and Thanh Dinh Ta presented to FPS in 2014 a new approach in malware retro-engineering. Usually, either communications, or code is analyzed. Here, the authors take a hybrid perspective. They showed how malware communication can be seen under a language perspective. They tested their idea on real malware and, for instance, showed that the botnet Zeus uses FTP as an underlying network support.
- **Supertagging with Constraints.** The parsing in Natural Language Processing is usually done by statistical analysis. Formal approaches are much more challenging, usually involving hard problems. Guillaume Bonfante, Bruno Guillaume, Mathieu Morey, and Guy Perrier [24] propose a new stream algorithm which discriminates tags in sentences.

6.3. Computability and Complexity

- **Genericity of semi-computable objects.** One of the main goals of computability theory is to understand and classify the algorithmic content of infinite objects, which can be expressed as the difficulty of computing them or as their ability to help solving problems. In establishing this classification one is often led to separate classes of algorithmic complexity and the construction of counter-examples is usually a hard task that requires the use of advanced technics (among which the so-called priority method with finite injury). The difficulty in such a construction is that the constructed object should satisfy two types of requirements going in opposite directions: it should lack algorithmic content but at the same time should be constructible in some way. In other words, these objects live somewhere between *generic* objects (objects with no structure) and *computable* objects (the most constructible objects). While computability theory provides formal notions of genericity, these ones are always incompatible with computability.

  We introduce a new notion of genericity which has two advantages: it is close to plain genericity, and we prove that it is compatible with semi-computability (for a property, being semi-decidable is a semi-computability notion while being decidable is a plain computability notion). The latter result has important consequences: many ad hoc existing constructions are subsumed by this result and then unified, new results can be obtained whenever the new notion of genericity captures the sought properties, and the result clarifies the role of topology in computability theory.

  This work is the sequel of the STACS 2013 paper [19] and is currently submitted [26].
- **Analytical properties of resource-bounded real functionals.** In [14] Hugo Férée, Walid Gomaa and Mathieu Hoyrup extend the results of [52] to non-deterministic complexity. More precisely, we introduce the analytical concepts of essential point and sufficient set for norms over continuous functions and use them to characterize the class of norms that are computable in non-deterministic polynomial time.
- **Call-by-value, call-by-name and the vectorial behaviour of the algebraic λ-calculus.** In this article published in LMCS (Logical Methods in Computer Science) [12], Ali Assaf, Alejandro Díaz-Caro, Simon Perdrix, Christine Tasson and Benoît Valiron examine the relationship between the algebraic lambda-calculus, a fragment of the differential lambda-calculus and the linear-algebraic lambda-calculus, a candidate lambda-calculus for quantum computation. Both calculi are algebraic: each one is equipped with an additive and a scalar-multiplicative structure, and their set of terms is closed under linear combinations. However, the two languages were built using different approaches: the former is a call-by-name language whereas the latter is call-by-value; the former considers algebraic equalities whereas the latter approaches them through rewrite rules. In this paper, they analyse how these different approaches relate to one another. To this end, four canonical languages based on each of the possible choices are proposed: call-by-name versus call-by-value, algebraic equality versus algebraic rewriting. The various languages are simulating each other. Due to subtle interaction between beta-reduction and algebraic rewriting, to make the languages consistent some additional hypotheses such as confluence or normalisation might be required.
- **Real or Natural numbers interpretations and their effect on complexity.** Guillaume Bonfante, Florian Deloup and Antoine Henrot [13] have shown how deep results in algebraic geometry may be read in a complexity perspective. They show that real numbers though they are not well founded can be used as natural numbers are for program interpretations. The argument is based on Positivstellensatz, a major result proved by Stengle.
- **Information carried by programs about the objects they compute.** In computability theory and computable analysis, finite programs can compute infinite objects. Presenting a computable object via any program for it, provides at least as much information as presenting the object itself, written on an infinite tape. What additional information do programs provide? We characterize this additional information to be any upper bound on the Kolmogorov complexity of the object, i.e., it gives an upper bound on size of a shortest program computing the object.

  This problem can be formalized using the two classical models of computation of Markov-computability [61] and Type-2 computability [74], which are the most famous and studied ways of

computing with infinite objects. Many celebrated results comparing these models have been developed in the 50's (theorems by Rice, Rice-Shapiro, Kreisel-Lacombe-Schoenfiled/Ceitin, Friedberg) but a complete understanding of their precise relationship has never been obtained. Our results fill this void, identifying the exact relationship between the two models. In particular this relationship enables us to obtain several results characterizing the computational and topological structure of Markov-semidecidable properties.

This work, made in collaboration with Cristóbal Rojas (Santiago) during his visit as an Inria "Chercheur Invité", has been accepted in STACS 2015 [20].

- **Causal Graph Dynamics.** Causal Graph Dynamics extend Cellular Automata to arbitrary, bounded-degree, time-varying graphs. The whole graph evolves in discrete time steps, and this global evolution is required to have a number of physics-like symmetries: shift-invariance (it acts everywhere the same) and causality (information has a bounded speed of propagation). Pablo Arrighi, Emmanuel Jeandel, Simon Martiel (I3S, Univ. Nice-Sophia Antipolis), and Simon Perdrix are investigating the properties of this model. In particular a work on the reversibility of causal graph dynamics has just been submitted in January 2015.
- **The Parameterized Complexity of Domination-type Problems and Application to Linear Codes.** In this article presented at TAMC'14 (Theory and Applications of Models of Computation) [17], David Cattanéo and Simon Perdrix study the parameterized complexity of domination-type problems. (σ, ρ)-domination is a general and unifying framework introduced by Telle: given $\sigma, \rho \subseteq \mathbb{N}$, a set D of vertices of a graph G is (σ, ρ)-dominating if for any $v \in D$, $|N(v) \cap D| \in \sigma$ and for any $v \notin D, |N(v) \cap D| \in \rho$. The main result is that for any σ and ρ recursive sets, deciding whether there exists a (σ, ρ)-dominating set of size k, or of size at most k, are both in W[2]. This general statement is optimal in the sense that several particular instances of (σ, ρ)-domination are W[2]-complete (e.g., DOMINATING SET). This result is also extended to a class of domination-type problems which do not fall into the (σ, ρ)-domination framework, including CONNECTED DOMINATING SET and the problem of the minimal distance of a linear code over a finite field.

  To prove the W[2]-membership of the domination-type problems the authors extend the Turing-way to parameterized complexity by introducing a new kind of non-deterministic Turing machine with the ability to perform 'blind' transitions, i.e., transitions which do not depend on the content of the tapes.
- **Quantum Circuits for the Unitary Permutation Problem**. In this paper [18] presented at DCM'14 (New Development in Computational models) and at the Workshop on Quantum Metrology, Interaction, and Causal Structure 2014 (invited talk), Stefano Facchni and Simon Perdrix consider the *Unitary Permutation* problem which consists, given n quantum gates $U_1, ..., U_n$ and a permutation σ of $\{1, ..., n\}$, in applying the quantum gates in the order specified by σ, i.e., in performing $U_{\sigma(n)} \circ ... \circ U_{\sigma(1)}$.

  This problem has been introduced and investigated in [40] where two models of computations are considered. The first is the (standard) model of query complexity: the complexity measure is the number of calls to any of the quantum gates U_i in a quantum circuit which solves the problem. The second model is roughly speaking a model for higher order quantum computation, where quantum gates can be treated as objects of second order. In both model the existing bounds are improved, in particular the upper and lower bounds for the standard quantum circuit model are established by pointing out connections with the *permutation as substring* problem introduced by Karp.

CASSIS Project-Team

6. New Results

6.1. Highlights of the Year

Véronique Cortier was one of the two FLoC plenary speakers during the Vienna Summer of Logic [31].

Steve Kremer and Robert Künnemann got a paper accepted at the 35th IEEE symposium on Security and Privacy [45].

The ANR project SEQUOIA has been accepted.

BEST PAPERS AWARDS :
[43] **Software Security and Reliability (SERE)**. E. FOURNERET, J. CANTENOT, F. BOUQUET, B. LEGEARD, J. BOTELLA.
[47] **The 7th International Symposium on Foundations & Practice of Security FPS'2014**. H. H. NGUYEN, A. IMINE, M. RUSINOWITCH.

6.2. 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.

6.2.1. Combination of Satisfiability Procedures

Participant: Christophe Ringeissen.

A satisfiability problem is often expressed in a combination of theories, and a natural approach consists in solving the problem by combining the satisfiability procedures available for the component theories. This is the purpose of the combination method introduced by Nelson and Oppen. However, in its initial presentation, the Nelson-Oppen combination method requires the theories to be signature-disjoint and stably infinite (to guarantee the existence of an infinite model). The design of a combination method for non-disjoint unions of theories is clearly a hard task but it is worth exploring simple non-disjoint combinations that appear frequently in verification. An example is the case of shared sets, where sets are represented by unary predicates. Another example is the case of bridging functions between data structures and a target theory (e.g., a fragment of arithmetic). In collaboration with Paula Chocron (U. Buenos Aires, former intern in Cassis) and Pascal Fontaine (project-team Veridis), we have investigated both cases.

The notion of gentle theory has been introduced in the last few years as one solution to go beyond the restriction of stable infiniteness, but in the case of disjoint theories. In [36], [59], we adapt the notion of gentle theory to the non-disjoint combination of theories sharing only unary predicates (plus constants and the equality). Like in the disjoint case, combining two theories, one of them being gentle, requires some minor assumptions on the other one. We show that major classes of theories, i.e., Loewenheim and Bernays-Schoenfinkel-Ramsey, satisfy the appropriate notion of gentleness introduced for this particular non-disjoint combination framework.

We have also considered particular non-disjoint unions of theories connected via bridging functions [37]. We present a combination procedure which is proved correct for the theory of absolutely free data structures. We consider the problem of adapting the combination procedure to get a satisfiability procedure for the standard interpretations of the data structure.

6.2.2. Unification Modulo Equational Theories of Cryptographic Primitives

Participants: Christophe Ringeissen, Michaël Rusinowitch.

Asymmetric unification is a new paradigm for unification modulo theories that introduces irreducibility constraints on one side of a unification problem. It has important applications in symbolic cryptographic protocol analysis, for which it is often necessary to put irreducibility constraints on portions of a state. However many facets of asymmetric unification that are of particular interest, including its behavior under combinations of disjoint theories, remain poorly understood. In [42], [63] we give a new formulation of the method for unification in the combination of disjoint equational theories developed by Baader and Schulz that both gives additional insights into the disjoint combination problem in general, and furthermore allows us to extend the method to asymmetric unification, giving the first unification method for asymmetric unification in the combination of disjoint theories.

Some attacks exploit in a clever way the interaction between protocol rules and algebraic properties of cryptographic operators. In [79], we provide a list of such properties and attacks as well as existing formal approaches for analyzing cryptographic protocols under algebraic properties.

We have further investigated unification problems related to the Cipher Block Chaining (CBC) mode of encryption. We first model chaining in terms of a simple, convergent, rewrite system over a signature with two disjoint sorts: list and element. The 2-sorted convergent rewrite system is then extended into one that captures a block chaining encryption-decryption mode at an abstract level, (using no AC-symbols); unification modulo this extended system is shown to be decidable [13].

6.2.3. Enumeration of Planar Proof Terms

Participant: Alain Giorgetti.

By the Curry-Howard isomorphism, simply typed lambda terms correspond to natural deduction proofs in minimal logic. Abramsky has introduced a notion of planarity for proof terms, from a graphical representation of proofs. Noam Zeilberger and Alain Giorgetti have initiated an enumerative study of normal planar lambda terms. At the MAP 2014 workshop in Paris, Noam Zeilberger conjectured that the sequence counting the number of closed normal planar lambda terms by increasing size may coincide with the one counting the number of rooted planar maps by number of edges. Zeilberger and Giorgetti started discussing this curious coincidence at the workshop and found a proof that both families are in size-preserving bijection [70]. Although the formal aspect is not emphasized in the paper, the use of formal representations of both normal planar lambda terms and rooted planar maps, of logic programming and a proof assistant software helped much in more quickly finding the bijection. Moreover the result puts a new light on the structure of proofs in minimal logic.

6.3. 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 [76]. We have edited a book [71] where each chapter presents an important and now standard analysis technique. This year, we have written a tutorial that may serve when teaching formal analysis of security protocols [26]. We develop new techniques for richer primitives, wider classes of protocols and higher security guarantees. In Section 6.5.3 we consider derived testing techniques for verifying protocol implementations.

6.3.1. Voting Protocols

Participants: Véronique Cortier, David Galindo-Chacon, Stéphane Glondu, Steve Kremer.

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.

One famous e-voting protocol is Helios, an open-source web-based end-to-end verifiable electronic voting system, used e.g., by UCL and the IACR association in real elections. One main advantage of Helios is its verifiability, up-to the ballot box (a dishonest ballot box may add ballots). We have defined a variant of Helios, named Belenios, that prevents from ballot stuffing, even against a dishonest ballot box. Our approach consists in introducing an additional authority that provides credentials that the ballot box can verify but not forge. Ballot privacy of Belenios then follows from ballot privacy of Helios. For full verifiability, we had first to adapt existing definitions of verifiability in the case of a corrupted ballot box and then prove verifiability of Helios [40], [61].

This new version has been implemented by Stéphane Glondu and has been tested in an election that involved the members of the Inria Nancy-Grand Est center and the LORIA lab (about 500 people that had to chose the best LORIA pictures).

Even a basic property like ballot secrecy is difficult to define formally and several definitions co-exist. We studied all game-based privacy definitions of the literature and discovered that none of them was satisfactory: they were either limited (not fully modeling e-voting protocols), or too strong (incompatible with verifiability), or even flawed for a few of them. Based on our findings, we have proposed a new game-based privacy definition BPRIV, proved that it implies simulation-based privacy and showed that it is realized by the Helios protocol.

Existing automated analysis techniques are inadequate to deal with commonly used cryptographic primitives, such as homomorphic encryption and mix-nets, as well as some fundamental security properties, such as verifiability. In collaboration with Matteo Maffei and Fabienne Eigner (Saarland University) we propose a novel approach based on refinement type systems for the automated analysis of two fundamental properties of e-voting protocols, namely, vote privacy and verifiability. We demonstrate the effectiveness of our approach by developing the first automated analysis of Helios using an off-the-shelf type-checker.

We have presented some of our results on e-voting as plenary speaker of FLOC 2014 [31].

6.3.2. Other Families of Protocols

Participants: Véronique Cortier, Steve Kremer, 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. We have proposed a new model and an associated decision procedure to check whether a routing protocol can ensure that honest nodes only accept valid routes, even if one of the nodes of the network is compromised. This result has been obtained for a bounded number of sessions, adapting constraint solving techniques to node topologies as well as some families of recursive tests, used in routing protocols [15].

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 designed a generic API for key-management based on key hierarchy [20], that can self-recover from corruption of arbitrary keys, provided the number of corrupted, active keys is smaller than some threshold.

Security APIs, key servers and protocols that need to keep the status of transactions, require to maintain a global, non-monotonic state, e.g., in the form of a database or register. However, most existing automated verification tools do not support the analysis of such stateful security protocols - sometimes because of fundamental reasons, such as the encoding of the protocol as Horn clauses, which are inherently monotonic. A notable exception is the recent tamarin prover which allows specifying protocols as multiset rewrite (MSR) rules, a formalism expressive enough to encode states. As multiset rewriting is a “low-level” specification language with no direct support for concurrent message passing, encoding protocols correctly is a difficult and error-prone process. In [45] we propose a process calculus with constructs for manipulation of a global state by processes running in parallel. We show that this language can be translated to MSR rules whilst preserving

all security properties expressible in a dedicated first-order logic for security properties. The translation has been implemented in a prototype tool which uses the tamarin prover as a backend. We apply the tool to several case studies among which a simplified fragment of PKCS#11, the Yubikey security token, and an optimistic contract signing protocol.

6.3.3. Automated Verification of Indistinguishability Properties

Participants: Vincent Cheval, 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.

Active case, unbounded number of sessions. We have studied how to reduce the search space for attacks on equivalence-based properties, for an unbounded number of sessions. Specifically, we have shown [38], [60] that if there is an attack then there is one that is well-typed. Our result holds for a large class of typing systems and a large class of *determinate* security protocols. Assuming finitely many nonces and keys, we can derive from this result that trace equivalence is decidable for an unbounded number of sessions for a class of tagged protocols, yielding one of the first decidability results for the unbounded case. As an intermediate result, we also provide a novel decision procedure in the case of a bounded number of sessions.

Active case, bounded number of sessions. We previously proposed a procedure for approximating trace equivalence in the case of a bounded number of sessions, i.e., for a replication free fragment of a cryptographic process calculus. The procedure is implemented in the *Akiss* tool. While we proved soundness and correctness for any convergent rewrite system that has the finite variant property, termination of the procedure was still an open question. We have recently shown that the procedure indeed terminates for the class of subterm convergent rewrite systems. The submission of this result is in preparation.

6.3.4. Securely Composing Protocols

Participants: Véronique Cortier, Steve Kremer, Éric Le Morvan.

Protocols may interact with an arbitrary attacker which yields a verification problem that has several sources of unboundedness (size of messages, number of sessions, etc.). In [14], we characterise a class of protocols for which deciding security for an unbounded number of sessions is decidable, by the means of a composition result. More precisely, we present a simple transformation which maps a protocol that is secure for a bounded number of protocol sessions (a decidable problem) to a protocol that is secure for an unbounded number of sessions. The precise number of sessions that need to be considered is a function of the security property and we show that for several classical security properties a single session is sufficient. Therefore, in many cases our result yields a design strategy for security protocols: (i) design a protocol intended to be secure for a single session; and (ii) apply our transformation to obtain a protocol which is secure for an unbounded number of sessions.

Protocols are often built in a modular way. For example, authentication protocols may assume pre-distributed keys or may assume secure channels. However, when an authentication protocol has been proved secure assuming pre-distributed keys, there is absolutely no guarantee that it remains secure when executing a real protocol for distributing the keys. How the security of these protocols can be combined is an important issue that is studied in the PhD thesis started by Éric Le Morvan.

6.3.5. 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 somewhat 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.

Gergei Bana and Hubert Comon have proposed a new framework [73] where the symbolic model now specifies what an attacker *cannot* do instead of specifying what it can do. Checking protocols security can then be reduced to checking inconsistency of some set of first order formula. During his PhD, Guillaume Scerri studies how to develop a (polynomial) decision procedure for deciding consistency of sets of formulas, for some class of formulas corresponding to security protocols. This procedure has been extended and implemented, yielding the tool SCARY that can successfully analyse several protocols of the literature [52].

6.3.6. *Advanced Cryptographic Models*

Participant: David Galindo-Chacon.

A classical approach in cryptographic research consists in weakening the assumptions cryptographic primitives are built upon. The following works belong to this research line.

We generalize the decisional problem that was used to prove the security of a well-known hierarchical identity-based encryption scheme by Boneh, Boyen and Goh. We argue that our new problem is strictly harder than the original problem, and thus the security of the aforementioned cryptographic primitive is laid on even stronger foundations [24].

It is known how to transform certain canonical three-pass identification schemes into signature schemes via the Fiat-Shamir transform. Pointcheval and Stern showed that those schemes are existentially unforgeable in the random-oracle model leveraging the, at that time, novel forking lemma. Recently, a number of 5-pass identification protocols have been proposed. Extending the above technique to capture 5-pass identification schemes would allow to obtain novel unforgeable signature schemes. In this paper, we provide an extension of the forking lemma (and the Fiat-Shamir transform) in order to assess the security of what we call n-generic signature schemes. These include signature schemes that are derived from certain $(2n + 1)$-pass identification schemes. In doing so, we put forward a generic methodology for proving the security of a number of signature schemes derived from $(2n + 1)$-pass identification schemes for $n \geq 2$. As an application of this methodology, we obtain two new code-based existentially-unforgeable signature schemes, along with a security reduction. In particular, we solve an open problem in multivariate cryptography posed by Sakumoto, Shirai and Hiwatari at CRYPTO 2011 [22].

Traditionally, symbolic and computational models for cryptographic protocols do not take into account the data leaked due to the physical nature of the cryptographic computations. Recently, the research area of leakage-resilient cryptography has emerged in order to cope with this source of attacks in the computational model. We have studied a conjecture that states that an ElGamal-based public-key encryption scheme with stateful decryption resists lunch-time chosen ciphertext and leakage attacks in the only computation leaks information model. We have given a non-trivial upper bound on the amount of leakage tolerated by this conjecture. More precisely, we prove that the conjecture does not hold if more than a $(\frac{3}{8} + o(1))$ fraction of the bits are leaked at every decryption step, by showing a lunch-time attack that recovers the full secret key. The attack uses a new variant of the Hidden Number Problem, that we call Hidden Shares - Hidden Number Problem, which is of independent interest [25].

6.4. Model-based Verification

We have investigated extensions of regular model-checking to new classes of rewrite relations on terms. We have studied specification and proof of modular imperative programs, as well as of modal workflows.

6.4.1. *Tree Automata with Constraints*

Participants: Pierre-Cyrille Héam, Olga Kouchnarenko.

Tree automata with constraints are widely used to tackle data base algorithmic problems, particularly to analyse queries over XML documents. The model of Tree Automata with Global Constraints (TAGED) is a model introduced in 2009 for these purposes. The membership problem for TAGED is known to be NP-complete. The emptiness problem for TAGED is known to be decidable and the best known algorithm in the general case is non elementary. In collaboration with Vincent Hugot, we show that if there is at least one

negative constraint, the problem is already NP-hard [64]. In the future, we plan to investigate upper bounds for the emptiness problem with a unique negative constraint. We also plan to study the complexity of the universality problem with a single constraint.

6.4.2. Random Generation of Finite Automata

Participant: Pierre-Cyrille Héam.

Developing new algorithms and heuristics raises crucial evaluation issues, as improved worst-case complexity upper-bounds do not always transcribe into clear practical gains. A suite for software performance evaluation can usually gather three types of entries: benchmarks, hard instance and random inputs, that deliver average complexity estimations, for which the catch resides in obtaining a meaningful random distribution (for instance a uniform random distribution).

In collaboration with Jean-Luc Joly, we investigate the problem of randomly and uniformly generating deterministic pushdown automata [65]. Based on a recursive counting approach, we propose a polynomial time algorithm for this purpose. The influence of the accepting condition on the generated automata is also experimentally studied.

Partially ordered automata are finite automata where simple loops have length one. They appear in several verification techniques, such as computing closures under semi-commutation relations or studying FIFO systems. In [68], we use a Markov chain based approach to randomly - and uniformly - generate deterministic partially ordered automata. The advantage of such a technique is its flexibility, allowing for instance to easily bound the number of loops. Experiments show that the mixing time seems to be polynomial, providing a tractable approach.

6.4.3. Verification of Linear Temporal Patterns over Finite and Infinite Traces

Participants: Pierre-Cyrille Héam, 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 collaboration with Vincent Hugot, 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.

6.4.4. Machine-Learning Techniques for Regular Model-Checking

Participants: Maxime Bride, Pierre-Cyrille Héam.

Using a machine-learning approach, we address the general problem of regular model-checking of computing $R^*(L)$, when L is a regular language and R a relation. Rather than developing specific algorithms to compute $R^*(L)$, it consists in using Angluin style's algorithms. In [58], we focus on the generation of examples, counter-examples and on the design of an oracle for the specific case of semi-commutation relations. Experiments are promising, particularly for the sizes of the obtained automata, which are quite smaller than with dedicated algorithms.

6.4.5. Constraint Solving for Verifying Modal Workflow Specifications

Participants: Hadrien Bride, Olga Kouchnarenko.

Workflow Petri nets are well suited for modelling and analysing discrete event systems exhibiting behaviours such as concurrency, conflict, and causal dependency between events. They represent finite or infinite-state processes, and several important verification problems, like reachability or soundness, are known to be decidable. Modal specifications introduced in [84] allow loose or partial specifications in a framework based on process algebras.

Our work in [34] focuses on the verification of modal workflow specifications using constraint solving as a computational tool. Its main contribution consists of a formal framework based on constraint systems to model executions of workflow Petri nets and their structural properties, as well as to verify their modal specifications. An implementation and promising experimental results obtained within the proposed approach constitute a practical contribution. In particular, a business process example from the IT domain enables to successfully assess the reliability of our contributions.

6.4.6. Rewriting-based Mathematical Model Transformations

Participants: Walid Belkhir, Alain Giorgetti.

Since 2011 we collaborate 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). The goal is to design a software, called *MEMSALab*, for the automatic derivation of multiscale models of arrays of micro- and nanosystems. In this domain a model is a partial differential equation. Multiscale methods approximate it by another partial differential equation which can be numerically simulated in a reasonable time. The challenge consists in taking into account a wide range of geometries combining thin and periodic structures with the possibility of multiple nested scales. We have designed a transformation language facilitating the design of MEMSALab [17]. It is proposed as a Maple™ package for rule-based programming, rewriting strategies and their combination with standard Maple™ code. We illustrate the practical interest of this language by using it to encode two examples of multiscale derivations, namely the two-scale limit of the derivative operator and the two-scale model of the stationary heat equation. A more general framework for the derivation of the multi-scale models was established in [29].

6.5. 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 [55]. The test generation uses various underlying techniques such as symbolic animation of models [80], or symbolic execution of programs by means of dedicated constraints, SMT solvers, or model-checkers.

6.5.1. Automated Test Generation from Behavioral Models

Participants: Fabrice Bouquet, Kalou Cabrera, Jérome Cantenot, Frédéric Dadeau, Jean-Marie Gauthier, Julien Lorrain, Alexandre Vernotte.

We have developed an original model-based testing approach that takes a behavioral view (modelled in UML) of the system under test and automatically generates test cases and executable test scripts according to model coverage criteria [18]. We continue to extend this result to SysML specifications for validating embedded systems. We apply this method on smartSurface [44].

In the context of the FSN DAST project on Dynamic Application Security Testing, we investigated the use of a model-based testing approach for vulnerability testing in web applications. We designed a process based on two artefacts. First, a generic UML model, that is used to represent the web application entities (pages, forms, etc.), coupled with OCL constraints that describe the business logics of the application. Second, a set of test purposes, that will look for specific vulnerabilities (cross-site scripting, SQL injections, etc.). We have implemented a research prototype and applied it on several case studies. It has shown its effectiveness to detect vulnerabilities on already deployed web applications [50].

6.5.2. Scenario-Based Verification and Validation

Participants: Fabrice Bouquet, Kalou Cabrera, Frédéric Dadeau.

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 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 also 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 [41]. This process has been fully tool-supported into an integrated software prototype[0]. This process has been designed during the ANR TASCCC project (2009-2012) and was continued during the ANR ASTRID OSEP project (2012-2013). The industrialization of this approach, and its integration within commercial test generation tools has started with the ANR ASTRID Maturation MBT_Sec project (2014-2015).

In the context of the SecureChange project, we have also investigated 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 [43].

6.5.3. Mutation-based Testing of Security Protocols

Participants: Frédéric Dadeau, Pierre-Cyrille Héam, Ghazi Maatoug, Michaël Rusinowitch.

We have proposed a model-based penetration testing approach for security protocols [41]. 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 the protocol mutation tool jMuHLPSL that performs the mutations. The mutants are then analyzed by *CL-AtSe*. We have experimented our approach on a set of protocols, and we have shown the relevance of the proposed mutation operators and the efficiency of the *CL-AtSe* to conclude on the vulnerability of a protocol and produce an attack trace that can be used as a test case for implementations. We applied our approach on the Paypal Express protocol, and we were able to retrieve an existing attack trace on this protocol[0]. We also investigated the transformation of an attack trace into executable tests scripts. To achieve that, we have proposed to automatically generate skeletons of Java test programs that the validation engineer only has to fill in order to concretize the steps of the test. Experimentations on these principles have been described in [53].

6.5.4. Code and Contract-based Test Generation and Static Analysis

Participants: Fabrice Bouquet, Frédéric Dadeau, Ivan Enderlin, Alain Giorgetti.

With the CEA we have developed a test generation technique based on C code and formal specifications, to facilitate deductive verification, in a new tool named StaDy [67], [49], [51]. The tool integrates the concolic test generator PathCrawler within the static analysis platform Frama-C. StaDy is able to handle the ANSI C Specification Language (ACSL) of the framework and other Frama-C plug-ins are able to reuse results from the test generator. This tool is designed to be the foundation stone of modular static and dynamic analysis combinations in the Frama-C platform.

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 based

[0] A video of the prototype is available at: http://vimeo.com/53210102

[0] http://www.nbs-system.com/blog/faille-securite-magento-paypal.html

on various strategies, namely uniform random generation, bounded exhaustive generation and rule-coverage-based test generation. In a recent work, we have proposed a dedicated constraint solver for PHP arrays aiming to avoid rejection during the generation of array structures. Finally, we have proposed dedicated specification coverage criteria to drive the test generation process. These coverage criteria focus on the selection of a subset of a method's contract, or the selection of specific predicates or realistic domains inside the contract. The whole approach has been implemented into a dedicated framework [62] integrated with state-of-the-practice test execution environments, such as atoum.

6.5.5. Random Testing

Participants: Aloïs Dreyfus, Pierre-Cyrille Héam, Olga Kouchnarenko.

The random testing paradigm represents a quite simple and tractable software assessment method for various testing approaches. When performing random testing, the random sampler is supposed to be independent of tester choices or convictions: a solution is to exploit uniform random generators.

In [82] a method is proposed for drawing paths in finite graphs uniformly, and it is explained how to use these techniques for testing C programs within a control flow graph based approach. Nevertheless, as finite graphs often provide strong abstractions of the systems under test, many abstract tests generated by the approach cannot be played on the implementation. In [83], we have proposed a new approach, extending [82], to manage stack-call during the random test generation while preserving uniformity. In [23], we go further by investigating a way to biase the random testing, in order to optimize the probability to fulfil a coverage criterion. The new approaches have been implemented in a prototype and experimented on several examples.

6.6. 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.

6.6.1. Automatic Analysis of Web Services Security

Participants: Walid Belkhir, 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. This orchestration specification is expressed in ASLan language, a formal language designed for modeling Web Services tied with security policies that was developed in AVANTSSAR project. The AVANTSSAR Orchestrator (presented in [56]) generates an attack trace describing the execution of the mediator and translates it into ASLan. 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 can compile the ASLan specification into a Java servlet that can be used to execute the orchestration.

In [16] we develop our alternative approach based on *parametrized automata*, a natural extension of finite-state automata over infinite alphabet. In this model the transitions are labeled with constants or variables that can be refreshed in some specified states. We prove several closure properties for this class of automata and study their decision problems. We show the applicability of our model to Web services handling data from an infinite domain. We introduce a notion of simulation that enables us to reduce the Web service composition problem to the construction of a simulation of a target service by the asynchronous product of existing services, and prove that this construction is computable. The existence of a service orchestrator solving a service composition problem can alternatively be reduced to the satisfiability of formula in parametrized propositional dynamic logic, and the latter was shown decidable in [33].

We now work on synthesizing composed services that satisfy required security policies.

6.6.2. Secure Querying and Updating of XML Data

Participants: Abdessamad Imine, Houari Mahfoud, Michaël Rusinowitch.

It is increasingly common to find XML views used to enforce access control as found in many applications and commercial database systems. To overcome the overhead of view materialization and maintenance, XML views are necessarily virtual. With this comes the need for answering XML queries posed over virtual views, by rewriting them into equivalent queries on the underlying documents. A major concern here is that query rewriting for recursive XML views is still an open problem, and proposed approaches deal only with non-recursive XML views. Moreover, a small number of works have studied the access rights for updates. In [11], we present SVMAX (Secure and Valid MAnipulation of XML), the first system that supports specification and enforcement of both read and update access policies over arbitrary XML views (recursive or non). SVMAX defines general and expressive models for controlling access to XML data using significant class of XPath queries and in the presence of the update primitives of W3C XQuery Update Facility. Furthermore, SVMAX features an additional module enabling efficient validation of XML documents after primitive updates of XQuery. The wide use of W3C standards makes of SVMAX a useful system that can be easily integrated within commercial database systems. We give extensive experimental results, based on real-life DTDs, that show the efficiency and scalability of our system.

6.6.3. Secure Computation in Social Networks

Participants: Bao Thien Hoang, Abdessamad Imine, Huu Hiep Nguyen, Michaël Rusinowitch.

Online social networks are currently experiencing a peak and they resemble real platforms of social conversion and content delivery. Indeed, they are exploited in many ways: from conducting public opinion polls about any political issue to publish social graph data for achieving in-depth studies. To securely perform these large-scale computations, we need the design of reliable protocols to ensure the data privacy. To address the polling problem in social networks (where the privacy of exchanged information and user reputation are very critical), we provide a simple decentralized polling protocol that relies on the current state of social graphs. More explicitly, we define one family of social graphs that satisfy what we call the m-broadcasting property (where m is less than or equal to a minimum node degree). We show their structures enable low communication cost and constitute necessary and sufficient condition to ensure vote privacy and limit the impact of dishonest users on the accuracy of the polling output. To securely publish social graph data, we focus on the problem of anonymizing a deterministic graph by converting it into an uncertain form [48], [47]. We first analyze drawbacks in a recent uncertainty-based anonymization scheme and then propose Maximum Variance, a novel approach that gains better tradeoff between privacy and utility. Towards a fair comparison between the anonymization schemes on graphs, the second contribution of our work is to describe a quantifying framework for graph anonymization by assessing privacy and utility scores of typical schemes in a unified space.

6.6.4. Safe and Secure Protocols for Collaborative Applications

Participants: Abdessamad Imine, Michaël Rusinowitch.

The Operational Transformation (OT) approach, used in many collaborative editors, allows a group of users to concurrently update replicas of a shared object and exchange their updates in any order. The basic idea is to transform any received update operation before its execution on a replica of the object. Designing transformation functions for achieving convergence of object replicas is a critical and challenging issue. In this work, we investigate the existence of transformation functions [27]. From the theoretical point of view, two properties, named TP1 and TP2, are necessary and sufficient to ensure convergence. Using controller synthesis technique, we show that there are some transformation functions, which satisfy TP1 for the basic signatures of insert and delete operations. But, there is no transformation function, which satisfies both TP1 and TP2. Consequently, a transformation function which satisfies both TP1 and TP2 must necessarily have additional parameters in the signatures of some update operations. Accordingly, we provide a new transformation function and show formally that it ensures convergence.

In [19], 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 use 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. To validate our approach, we implement an optimistic access control on the top of a collaboration prototype and measure its performance in the distributed grid GRID'5000 to highlight the scalability of our solution.

However, verifying whether the combination of access control and coordination protocols preserves the data consistency is a hard task since it requires examining a large number of situations. In [30], we specify this access control protocol in the first-order relational logic with Alloy, and we verify that it preserves the correctness of the system on which it is deployed, namely that the access control policy is enforced identically at all participating user sites and, accordingly, the data consistency remains still maintained.

COMETE Project-Team

6. New Results

6.1. Highlights of the Year

- Prix de thèse de l'Ecole Polytechnique 2014 for the thesis "The Epistemic View of Concurrency Theory" by Sophia Knight (Defended 20 September, 2013).
- Catuscia Palamidessi has been invited keynote speaker at the joint conferences CONCUR 2014 and TGC 2014. Rome, September 2014.

6.2. 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.

6.2.1. Additive and multiplicative notions of leakage, and their capacities

Protecting sensitive information from improper disclosure is a fundamental security goal. It is complicated, and difficult to achieve, often because of unavoidable or even unpredictable operating conditions that can lead to breaches in planned security defences. An attractive approach is to frame the goal as a quantitative problem, and then to design methods that measure system vulnerabilities in terms of the amount of information they leak. A consequence is that the precise operating conditions, and assumptions about prior knowledge, can play a crucial role in assessing the severity of any measured vunerability.

In [20] we developed this theme by concentrating on vulnerability measures that are *robust* in the sense of allowing general leakage bounds to be placed on a program, bounds that apply whatever its operating conditions and whatever the prior knowledge might be. In particular we proposed a theory of channel capacity, generalising the Shannon capacity of information theory, that can apply both to additive and to multiplicative forms of a recently-proposed measure known as g-leakage. Further, we explored the computational aspects of calculating these (new) capacities: one of these scenarios can be solved efficiently by expressing it as a Kantorovich distance, but another turns out to be NP-complete.

We also found capacity bounds for arbitrary correlations with data not directly accessed by the channel, as in the scenario of Dalenius's Desideratum.

6.2.2. Compositionality Results for Quantitative Information Flow

In the min-entropy approach to quantitative information flow, the leakage is defined in terms of a minimization problem, which, in case of large systems, can be computationally rather heavy. The same happens for the recently proposed generalization called g-vulnerability. In [28] we studied the case in which the channel associated to the system can be decomposed into simpler channels, which typically happens when the observables consist of several components. Our main contribution was the derivation of bounds on the g-leakage of the whole system in terms of the g-leakages of its components.

6.2.3. LeakWatch: Estimating Information Leakage from Java Programs

Programs that process secret data may inadvertently reveal information about those secrets in their publicly-observable output. In [23] we presented LeakWatch, a quantitative information leakage analysis tool for the Java programming language; it is based on a flexible "point-to-point" information leakage model, where secret and publicly-observable data may occur at any time during a program's execution. LeakWatch repeatedly executes a Java program containing both secret and publicly-observable data and uses robust statistical techniques to provide estimates, with confidence intervals, for min-entropy leakage (using a new theoretical result presented in this paper) and mutual information. We demonstrated how LeakWatch can be used to estimate the size of information leaks in a range of real-world Java programs.

6.2.4. On the information leakage of differentially-private mechanisms

Differential privacy aims at protecting the privacy of participants in statistical databases. Roughly, a mechanism satisfies differential privacy if the presence or value of a single individual in the database does not significantly change the likelihood of obtaining a certain answer to any statistical query posed by a data analyst. Differentially-private mechanisms are often oblivious: first the query is processed on the database to produce a true answer, and then this answer is adequately randomized before being reported to the data analyst. Ideally, a mechanism should minimize leakage—i.e., obfuscate as much as possible the link between reported answers and individuals' data—while maximizing utility—i.e., report answers as similar as possible to the true ones. These two goals are, however, conflicting, and a trade-off between privacy and utility is imposed.

In [13] we used quantitative information flow principles to analyze leakage and utility in oblivious differentially-private mechanisms. We introduced a technique that exploits graph-symmetries of the adjacency relation on databases to derive bounds on the min-entropy leakage of the mechanism. We evaluated utility using identity gain functions, which are closely related to min-entropy leakage, and we derived bounds for it. Finally, given some graph-symmetries, we provided a mechanism that maximizes utility while preserving the required level of differential privacy.

6.2.5. Metric-based approaches for privacy in concurrent systems

In a series of two papers we investigated metric-based techniques for varifying differential privacy in the context of concurrent systems.

The first work [30] was motivated from the one of Tschantz et al., who proposed a verification method based on proving the existence of a stratified family of bijections between states, that can track the privacy leakage, ensuring that it does not exceed a given leakage budget. We improved this technique by investigating state properties which are more permissive and still imply differential privacy. We introduced a new pseudometric, still based on the existence of a family of bijections, but relaxing the relation between them by integrating the notion of amortization, and showed that this results to a more parsimonious use of the privacy budget. We also showed that for the new pseudometric the level of differential privacy is continuous on the distance between the starting states, which makes it suitable for verification.

Continuing this line of work, we studied the pseudometric based on the Kantorovich lifting, which is one of the most popular notions of distance between probabilistic processes proposed in the literature. However, its application in verification is limited to linear properties. In [19], we proposed a generalization which allows to deal with a wider class of properties, such as those used in security and privacy. More precisely, we proposed a family of pseudometrics, parametrized on a notion of distance which depends on the property we want to verify. Furthermore, we showed that the members of this family still characterize bisimilarity in terms of their kernel, and provided a bound on the corresponding distance between trace distributions. Finally, we studied the instance corresponding to differential privacy, and we showed that it has a dual form, easier to compute. We also proved that the typical process-algebra constructs are non-expansive, thus paving the way to a modular approach to verification.

6.2.6. Optimal Geo-Indistinguishable Mechanisms for Location Privacy

With location-based services becoming increasingly more popular, serious concerns are being raised about the potential privacy breaches that the disclosure of location information may induce. In [21] we considered two approaches that have been proposed to limit and control the privacy loss: one is the *geo-indistinguishability* notion developed within Comète, which is inspired by differential privacy, and like the latter it is independent from the side knowledge of the adversary and robust with respect to composition of attacks. The other one is the mechanism of Shokri et al., which offers an optimal trade-off between the loss of quality of service and the privacy protection with respect to a given Bayesian adversary.

We showed that it is possible to combine the advantages of the two approaches: given a minimum threshold for the degree of geo-indistinguishability, we construct a mechanism that offers the maximal utility, as the solution of a linear program. Thanks to the fact that geo-indistinguishability is insensitive to the remapping of a Bayesian adversary, the mechanism so constructed is optimal also in the sense of Shokri et al. Furthermore

we proposed a method to reduce the number of constraints of the linear program from cubic to quadratic (with respect to the number of locations), maintaining the privacy guarantees without affecting significantly the utility of the generated mechanism. This lowers considerably the time required to solve the linear program, thus enlarging significantly the size of location sets for which the optimal trade-off mechanisms can still be computed.

6.2.7. A Predictive Differentially-Private Mechanism for Mobility Traces

With the increasing popularity of GPS-enabled handheld devices, location based applications and services have access to accurate and real-time location information, raising serious privacy concerns for their millions of users. Trying to address these issues, the notion of *geo-indistinguishability* was recently introduced, adapting the well-known concept of Differential Privacy to the area of location-based systems. A Laplace-based obfuscation mechanism satisfying this privacy notion works well in the case of a *sporadic* use; Under repeated use, however, *independently* applying noise leads to a quick loss of privacy due to the correlation between the location in the trace.

In [22] we showed that correlations in the trace can be in fact exploited in terms of a *prediction function* that tries to guess the new location based on the previously reported locations. The proposed mechanism tests the quality of the predicted location using a private test; in case of success the prediction is reported otherwise the location is sanitized with new noise. If there is considerable correlation in the input trace, the extra cost of the test is small compared to the savings in budget, leading to a more efficient mechanism.

We evaluated the mechanism in the case of a user accessing a location-based service while moving around in a city. Using a simple prediction function and two budget spending strategies, optimizing either the utility or the budget consumption rate, we showed that the predictive mechanism can offer substantial improvements over the independently applied noise.

6.2.8. A differentially private mechanism of optimal utility for a region of priors

Differential privacy is a notion of privacy that was initially designed for statistical databases, and has been recently extended to a more general class of domains. Both differential privacy and its generalized version can be achieved by adding random noise to the reported data. Thus, privacy is obtained at the cost of reducing the data's accuracy, and therefore their *utility*.

In [31] we considered the problem of identifying *optimal* mechanisms for generalized differential privacy, i.e. mechanisms that maximize the utility for a given level of privacy. The utility usually depends on a prior distribution of the data, and naturally it would be desirable to design mechanisms that are *universally optimal*, i.e., optimal for all priors. However it is already known that such mechanisms do not exist in general. We then characterized maximal *classes of priors* for which a mechanism which is optimal for all the priors of the class *does exist*. We showed that such classes can be defined as convex polytopes in the priors space.

As an application, we considered the problem of privacy that arises when using, for instance, location-based services, and we showed how to define mechanisms that maximize the quality of service while preserving the desired level of geo-indistinguishability.

6.2.9. 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 [14] 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.

6.3. 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.

6.3.1. A Concurrent Pattern Calculus

In [16] we detailed how Concurrent pattern calculus (CPC) drives interaction between processes by comparing data structures, just as sequential pattern calculus drives computation. By generalising from pattern matching to pattern unification, interaction becomes symmetrical, with information flowing in both directions. CPC provides a natural language to express trade where information exchange is pivotal to interaction. The unification allows some patterns to be more discriminating than others; hence, the behavioural theory must take this aspect into account, so that bisimulation becomes subject to compatibility of patterns. Many popular process calculi can be encoded in CPC; this allows for a gain in expressiveness, formalised through encodings.

6.3.2. An Intensional Concurrent Faithful Encoding of Turing Machines

The benchmark for computation is typically given as Turing computability; the ability for a computation to be performed by a Turing Machine. Many languages exploit (indirect) encodings of Turing Machines to demonstrate their ability to support arbitrary computation. However, these encodings are usually by simulating the entire Turing Machine within the language, or by encoding a language that does an encoding or simulation itself. This second category is typical for process calculi that show an encoding of lambda-calculus (often with restrictions) that in turn simulates a Turing Machine. Such approaches lead to indirect encodings of Turing Machines that are complex, unclear, and only weakly equivalent after computation. In [25] we developed an approach to encoding Turing Machines into intensional process calculi that is faithful, reduction preserving, and structurally equivalent. The encoding is demonstrated in a simple asymmetric concurrent pattern calculus before generalised to simplify infinite terms, and to show encodings into Concurrent Pattern Calculus and Psi Calculi.

6.3.3. Expressiveness via Intensionality and Concurrency

Computation can be considered by taking into account two dimensions: extensional versus intensional, and sequential versus concurrent. Traditionally sequential extensional computation can be captured by the lambda-calculus. However, recent work shows that there are more expressive intensional calculi such as SF-calculus. Traditionally process calculi capture computation by encoding the lambda-calculus, such as in the pi-calculus. Following this increased expressiveness via intensionality, other recent work has shown that concurrent pattern calculus is more expressive than pi-calculus. In [26] we formalised the relative expressiveness of all four of these calculi by placing them on a square whose edges are irreversible encodings. This square is representative of a more general result: that expressiveness increases with both intensionality and concurrency.

6.3.4. On the Expressiveness of Intensional Communication

The expressiveness of communication primitives has been explored in a common framework based on the pi-calculus by considering four features: synchronism (asynchronous vs synchronous), arity (monadic vs polyadic data), communication medium (shared dataspaces vs channel-based), and pattern-matching (binding to a name vs testing name equality). In [27] pattern-matching is generalised to account for terms with internal structure such as in recent calculi like Spi calculi, Concurrent Pattern Calculus and Psi calculi. This exploreD intensionality upon terms, in particular communication primitives that can match upon both names and structures. By means of possibility/impossibility of encodings, we showed that intensionality alone can encode synchronism, arity, communication-medium, and pattern-matching, yet no combination of these without intensionality can encode any intensional language.

6.3.5. Weak CCP Bisimilarity with Strong Procedures

Concurrent constraint programming (CCP) is a well-established model for concurrency that singles out the fundamental aspects of asynchronous systems whose agents (or processes) evolve by posting and querying (partial) information in a global medium. Bisimilarity is a standard behavioral equivalence in concurrency theory. However, only recently a well-behaved notion of bisimilarity for CCP, and a CCP partition refinement algorithm for deciding the strong version of this equivalence have been proposed. Weak bisimilarity is a central behavioral equivalence in process calculi and it 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 [17] we demonstrated that, because of its involved labeled transitions, the above-mentioned saturation technique does not work for CCP. We 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.

6.3.6. Efficient Algorithms for Program Equivalence for Confluent Concurrent Constraint Programming

While the foundations and principles of CCP e.g., semantics, proof systems, axiomatizations, have been thoroughly studied for over the last two decades. In contrast, the development of algorithms and automatic verification procedures for CCP have hitherto been far too little considered. To the best of our knowledge there is only one existing verification algorithm for the standard notion of CCP program (observational) equivalence. In [18] we first showed that this verification algorithm has an exponential-time complexity even for programs from a representative sub-language of CCP; the summation-free fragment (CCP+). We then significantly improved on the complexity of this algorithm by providing two alternative polynomial-time decision procedures for CCP+ program equivalence. Each of these two procedures has an advantage over the other. One has a better time complexity. The other can be easily adapted for the full language of CCP to produce significant state space reductions. The relevance of both procedures derives from the importance of CCP+. This fragment, which has been the subject of many theoretical studies, has strong ties to first-order logic and an elegant denotational semantics, and it can be used to model real-world situations. Its most distinctive feature is that of confluence, a property we exploited to obtain our polynomial procedures.

6.3.7. A Behavioral Congruence for Concurrent Constraint Programming with Nondeterministic Choice

Weak bisimilarity is one of the most representative notions of behavioral equivalence for models of concurrency. As we mentioned earlier, a notion of weak bisimilarity, called weak saturated barbed bisimilarity (wsbb), was recently proposed for CCP. This equivalence improves on previous bisimilarity notions for CCP that were too discriminating and it is a congruence for the choice-free fragment of CCP. In [29], however, we showed that wsbb is not a congruence for CCP with nondeterministic choice. We then introduced a new notion of bisimilarity, called weak full bisimilarity (wfb), and showed that it is a congruence for the full language of CCP. We also showed the adequacy of wfb by establishing that it coincides with the congruence induced by closing wsbb under all contexts. The advantage of the new definition is that, unlike the congruence induced by wsbb, it does not require quantifying over infinitely many contexts.

6.3.8. Abstract Interpretation of Temporal Concurrent Constraint Programs

Timed Concurrent Constraint Programming (tcc) is a declarative model for concurrency offering a logic for specifying reactive systems, i.e. systems that continuously interact with the environment. The universal tcc formalism (utcc) is an extension of tcc with the ability to express mobility. Here mobility is understood as communication of private names as typically done for mobile systems and security protocols. In [15] we considered the denotational semantics for tcc, and we extended it to a "collecting" semantics for utcc based on closure operators over sequences of constraints. Relying on this semantics, we formalized a general framework for data flow analyses of tcc and utcc programs by abstract interpretation techniques. The concrete and abstract semantics we proposed are compositional, thus allowing us to reduce the complexity of data flow analyses. We

showed that our method is sound and parametric with respect to the abstract domain. Thus, different analyses can be performed by instantiating the framework. We illustrated how it is possible to reuse abstract domains previously defined for logic programming to perform, for instance, a groundness analysis for tcc programs. We showed the applicability of this analysis in the context of reactive systems. Furthermore, we made use of the abstract semantics to exhibit a secrecy flaw in a security protocol. We also showed how it is possible to make an analysis which may show that tcc programs are suspension free. This can be useful for several purposes, such as for optimizing compilation or for debugging.

6.3.9. Bisimulation for Markov Decision Processes through Families of Functional Expressions

In [24], we transfered a notion of quantitative bisimilarity for labelled Markov processes to Markov decision processes with continuous state spaces. This notion takes the form of a pseudometric on the system states, cast in terms of the equivalence of a family of functional expressions evaluated on those states and interpreted as a real-valued modal logic. Our proof amounted to a slight modification of previous techniques used to prove equivalence with a fixed-point pseudometric on the state-space of a labelled Markov process and making heavy use of the Kantorovich probability metric. Indeed, we again demonstrated equivalence with a fixed-point pseudometric defined on Markov decision processes; what is novel is that we recasted this proof in terms of integral probability metrics defined through the family of functional expressions, shifting emphasis back to properties of such families. The hope is that a judicious choice of family might lead to something more computationally tractable than bisimilarity whilst maintaining its pleasing theoretical guarantees. Moreover, we used a trick from descriptive set theory to extend our results to MDPs with bounded measurable reward functions, dropping a previous continuity constraint on rewards and Markov kernels.

DICE Team

5. New Results

5.1. The economy of intermediation

We have presented in [6] an introductory panorama on the disruption of the intermediation revolution. Our efforts to measure data flows in the world, have been pursued [2] to estimate the concentration of the data industry. It is well known that the main platforms of the Web are concentrated in a few countries, mostly in the USA. Some countries, mostly in Asia, such as China, Russia, Korea or Japan have successfully developed their own Web 2.0 industry, while others, such as European countries, have failed to do so. We have explored in [7] the strategy of China, which has the largest Web industry behind the US and has made a priority of keeping its data at home, with systems in all activity sectors developed in general only one or two years after their main American counterparts. The innovation strategy of China aims in all fields to achieve technological independence, with at most 30% of foreign IP.

The rise of the economy of data disrupts values, such as privacy, and the way we think about our visibility. In [9], we investigate the digital world from an ethical perspective and a computer science viewpoint. We assess the structure and the dynamic of digital visibility and propose a model-driven approach to handle visibility in service compositions.

5.2. Architecture design for intermediation platforms

During our joint work with Worldline we built a JavaScript compiler for generating dataflow program from plain standard JavaScript sources. In an ACM Middleware conference poster session we raised the question of extracting a dataflow design from JavaScript callback hell. The compiler https://github.com/etnbrd/due-compiler is used to help JavaScript standard developers generate their equivalent dataflow scheme without the need of external libraries such as Promises, Async or Q. With this tool, developer may migrate their javascript legacy code towards a new flow based design. Our due npm module https://github.com/etnbrd/due is a first step towards a dynamic flow based architecture studied in Etienne's project.

The C3PO project provides a browser based application for interacting with other nearby participants in chat mode. The client architecture runs exclusively in the browser over a DTN layer and listens to posts send through a dedicated spontaneous and ephemeral social network (SESN) [5]. The client is organized around a display canvas hosting plugins. Each plugin registers for some tags it wishes to handle. The local DTN manager receives posts and propagates them to the plugins.

We have used intermediation technologies for voting systems. A brief presentation of our motivations has been made in [4]. A patent on the BitBallot protocol is on its way.

PRIVATICS Project-Team

5. New Results

5.1. Highlights of the Year

Vincent Roca was awarded the 3rd Applied Research price of the Fédération des Industries Electriques, Electroniques et Communications (FIEEC), for his transfer activities to the Expway French SME, Lyon, October 8th, 2014.

The team got two major contributions:

- *A Case Study: Privacy Preserving Release of Spatio-temporal Density in Paris* was published by Gergely Acs and Castelluccia at KDD 2014.
- *Censorship in the Wild: Analyzing Internet Filtering in Syria* was published by Chaabane Abdelberi, Mathieu Cunche,and Mohamed Ali Kaafar at IMC 2014.

5.2. Filtering and blocking the Internet

Participants: Mohamed Ali Kaafar, Abdelberi Chaabane, Mathieu Cunche, Cédric Lauradoux, Amrit Kumar.

- **Censorship**

  Based on 600GB leaked logs from appliances used to filter Internet traffic in Syria, we performed an analysis of the Syrian censorship apparatus. This study have been published in ACM Internet Measurement Conference [7].

  We found that the Internet traffic in Syria was filtered in several ways using IP addresses, domain names and keywords. Content sharing, instant messaging and proxy technologies were heavily censored. Some social media such as badoo.com were fully censored, but others such as Facebook are only censored for specific political and religious pages. We also found evidences of successful usage of censorship-circumvention techniques such as Tor and VPN. We also found that P2P file-sharing and Google cache were used to escape censorship blockage.

  While our work might help organizations on both sides of the censorship line, we believe the presented results can help understand the underlying technologies, policies and can inform the design of tools designed to evade the censorship.

- **Attacking filters** Many major Internet companies use probabilistic techniques to filter the users requests or to prevent malicious attacks. In our work [35], [34], we show how they can be polluted/saturated using pre-image attacks and how it increases the false-positive probability. Then, we show how to forge false-positives to mount attacks. In the adversarial settings, we have the liberty to assume that the inputs to the filter are non uniformly distributed. This observation leads to our second contribution: we compute the worst case false-positive probability and obtain new equations for Bloom filter parameters. To support our contributions, we provide four attacks on software applications based on Bloom filter: Bloom-enabled SCRAPY web spider, BITLYDABLOOMS spam filter, SQUID web cache and GOOGLE Safe Browsing. Our attacks retain some form of DoS. They are all based on the forgery of Uniform Resource Locators (URLs) matching certain pre-image or second pre-image property. The impact of our attack ranges from denial-of-service to massively distributed denial-of-service with reflection.

5.3. Selling Off Privacy at Auction

Participants: Claude Castelluccia, Lukasz Olejnik, Minh-Dung Tran.

The first one is a privacy analysis of Real-Time Bidding (RTB) and Cookie Matching (CM). RTB is a technology that allows ad buyers (advertisers) and ad sellers (publishers) to buy and sell ad spaces at real-time auctions through ad exchanges. In RTB, when user visits a publisher page, the ad impression (i.e. one ad display in an ad space) and the user information are immediately broadcast by the ad exchange to a number of bidders (i.e. advertisers or their representatives) for them to bid for the chance to serve ads to this user. CM protocol allows the ad exchange and the bidder to synchronize their cookies of the same user, thus facilitating their exchange of user data.

In [13], we characterize and quantify the potential user web history leakage from ad exchanges to bidders in RTB as a result of exchanging user data. We also discuss and quantify the extent to which companies can potentially collude to increase their tracked user profiles using CM. In addition, we leverage a design characteristic of RTB to observe the winning price of each RTB auction. By analyzing these prices, we show how advertisers evaluate the value of user privacy. This work (titled Selling Off Privacy at Auction) will be presented in NDSS 2014, San Diego, USA in February, 2014.

5.4. Data anonymization

Participants: Claude Castelluccia, Gergely Acs.

With billions of handsets in use worldwide, the quantity of mobility data is gigantic. When aggregated they can help understand complex processes, such as the spread viruses, and built better transportation systems, prevent traffic congestion. While the benefits provided by these datasets are indisputable, they unfortunately pose a considerable threat to location privacy. At KDD 2014 [9], we present a new anonymization scheme to release the spatio-temporal density of Paris, in France, i.e., the number of individuals in 989 different areas of the city released every hour over a whole week. The density is computed from a call-data-record (CDR) dataset, provided by the French Telecom operator Orange, containing the CDR of roughly 2 million users over one week. Our scheme is differential private, and hence, provides provable privacy guarantee to each individual in the dataset. Our main goal with this case study is to show that, even with large dimensional sensitive data, differential privacy can provide practical utility with meaningful privacy guarantee, if the anonymization scheme is carefully designed. This work is part of the national project XData (http://xdata.fr) that aims at combining large (anonymized) datasets provided by different service providers (telecom, electricity, water management, postal service, etc.).

5.5. Wi-Fi and privacy

Participants: Jagdish Achara, Mathieu Cunche, Vincent Roca.

In Android, installing an application implies accepting the permissions it requests, and these permissions are then enforced at runtime. In our WISEC 2014 paper [29], we focus on the privacy implications of the `ACCESS_WIFI_STATE` permission. For this purpose, we analyzed permissions of the 2700 most popular applications on Google Play and found that the `ACCESS_WIFI_STATE` permission is used by 41% of them. We then performed a static analysis of 998 applications requesting this permission and based on the results, chose 88 applications for dynamic analysis. Our analyses reveal that this permission is already used by some companies to collect user Personally Identifiable Information (PII). We also conducted an online survey to study users' perception of the privacy risks associated with this permission. This survey shows that users largely underestimate the privacy implications of this permission. As this permission is very common, most users are therefore potentially at risk.

5.6. Sensor security and privacy

Participant: Marine Minier.

Wireless sensor networks (WSNs) are composed of a large number of low-cost, low-power, and multi-functional sensor nodes that communicate at short distance through wireless links. They are usually deployed in an open and uncontrolled environment where attackers may be present. Due to the use of low-cost materials, hardware components are not tamper-resistant and an adversary could access to a sensor's internal state. With Ochirkhand Erdene-Ochir and Pierre Brunisholz, we continue to work on the notion of resiliency in WSNs [17], [31].

5.7. Buidling blocks

Participant: Marine Minier.

In the context of the BLOC project funded by the ANR, we continue to work on Extended Generalized Feistel Network and on new ligthweight block cipher design (see [30]). We hope to obtain results in this area at the beginning of 2015. With Christine Solnon and Julia Reboul, we work on the formalism of related-key and chosen-key attacks against symmetric key primitives using constraint programming (CP). This preliminary work was presented at the CP 2014 workshop ModRef 2014 in [42].

5.8. Formal and legal issues of privacy

Participants: Thibaud Antignac, Denis Butin, Daniel Le Métayer.

- **Privacy Architectures: Reasoning About Data Minimization and Integrity** Privacy by design will become a legal obligation in the European Community if the Data Protection Regulation eventually gets adopted. However, taking into account privacy requirements in the design of a system is a challenging task. We present an approach based on the specification of privacy architectures at FM 2014 [12] and focus on a key aspect of privacy, data minimisation, and its tension with integrity requirements. We illustrate our formal framework through a smart metering case study.
- **Log Analysis for Data Protection Accountability**

  Accountability is increasingly recognized as a cornerstone of data protection, notably in European regulation, but the term is frequently used in a vague sense. For accountability to bring tangible benefits, the expected properties of personal data handling logs and the assumptions regarding the logging process must be defined with accuracy. At STM 2014 [10], we provide a formal framework for accountability and show the correctness of the log analysis with respect to abstract traces used to specify privacy policies. We also show that compliance with respect to data protection policies can be checked based on logs free of personal data, and describe the integration of our formal framework in a global accountability process.

PROSECCO Project-Team

6. New Results

6.1. Highlights of the Year

This year, we published 17 articles in international peer-reviewed journals and conferences, including papers in prestigious conferences such as POPL (2 papers) and all the top conferences in computer security: IEEE S&P Oakland (2 papers), CRYPTO, ACM CCS, NDSS, and Financial Cryptography. Our papers in these top venues (discussed later in New Results) serve as highlights of our research during the year. In addition to these papers, we published 1 PhD thesis and several technical reports.

We released updates to miTLS, ProVerif, CryptoVerif, and started working on a brand-new version of F*. We discovered serious vulnerabilities in a number of TLS libraries, web browsers, and web servers, resulting in 6 published CVEs, and over a dozen software updates based on our recommendations in widely used software such as Firefox, Chrome, Internet Explorer, Safari, OpenSSL, Java, and Mono.

We organized a winter school "The Joint EasyCrypt-F*-CryptoVerif School 2014" which attracted industrial researchers, academics, and students from around the world. Over 75 students learned to use cryptographic verification tools from instructors at Inria, IMDEA, and Microsoft Research. Two of the tools: CryptoVerif and F* are being developed in collaboration with Inria.

If we were to choose one research theme as our highlight of the year, it would be our activities surrounding Transport Layer Security (TLS):

- At CRYPTO 2014, we published a detailed cryptographic proof of the TLS handshake as implemented in miTLS
- At NDSS 2014, we published a study in the use of X.509 certificates in TLS servers on the web
- At IEEE S&P (Oakland), we published a new attack on the TLS protocol called the *triple handshake*, which affected all TLS libraries and mainstream TLS applications such as web browsers.
- To prevent our attack, we proposed patches to major software libraries as part of responsible disclosure. Our research directly led to security updates for all major web browsers and TLS implementations.
- We also proposed a long-term countermeasure for our attack, the TLS Session Hash extension, which we published as an internet draft and presented at the IETF. This draft is on its way to being a published standard and is already implemented in all major TLS libraries.
- We participated in the design of next version (1.3) of the TLS protocol. We hosted an interim TLS working group meeting in Paris. Our proposals such as the session hash construction are now an integral part of the new design, and we continue consulting on the design and implementation of TLS.

6.2. Verification of Security Protocols in the Symbolic Model

Participants: Bruno Blanchet, Miriam Paiola, Robert Künnemann.

Miriam Paiola wrote and defended her PhD thesis on the verification of security protocols with lists [45].

Robert Künnemann published a paper at the IEEE S&P conference on how to extend symbolic cryptographic protocol verifiers to account for global state [60].

Bruno Blanchet published a tutorial on the protocol verifier PROVERIF [66], as a follow-up to his teaching in the FOSAD'13 summer school last year.

The applied pi calculus is a widely used language for modeling security protocols, including as a theoretical basis of ProVerif. However, the seminal paper that describes this language (Abadi and Fournet, POPL'01) does not come with proofs, and detailed proofs for the results in this paper were never published. This year, Martin Abadi, Bruno Blanchet, and Cedric Fournet wrote detailed proofs for the main theorems of this paper. This work was also an opportunity to fix a few minor details in the results and to tune the calculus to improve it and make it closer to the input calculus of ProVerif. We plan to submit this work as a journal paper.

6.3. Verification of Security Protocols in the Computational model

Participants: Bruno Blanchet, David Cadé.

We worked on our computationally-sound protocol verifier CryptoVerif in two directions.

First, this verifier includes a specialized compiler that generates secure implementations of protocols from CryptoVerif specifications. We completed a journal version of the proof that this compiler preserves security, which is to appear in the Journal of Computer Security [48]

Second, Bruno Blanchet extended CryptoVerif with support for equational theories: associativity, commutativity, non-commutative and commutative groups, exclusive or. The goal is to be able to verify protocols that rely on the algebraic properties of groups and exclusive or. The extended tool is available at http://cryptoverif.inria.fr.

6.4. Computationally Complete Symbolic Attacker Models

Participants: Gergei Bana, Hubert Comon-Lundh.

A new approach to computational verification is to define a *computationally complete* symbolic attacker, so that a symbolic proof against this attacker can be shown to imply a computational proof of security. Following this line of inquiry, Gergei Bana and Hubert Comon-Lundh recently published work on proving computational reachability properties using symbolic techniques.

Gergei Bana (along with Hubert Comon-Lundh) published a paper on how to extend this work to prove stronger security properties expressed as equivalences [50]. Hence, the proof techniq can now be used also for properties like anonymity, strong secrecy etc. Besides being able to prove such properties, another advantage of this extension is that modern security properties of cryptographic primitives are also formulated in terms of indistinguishability, which makes it easier to translate the security properties cryptographers define to our language than before.

Using the computationally complete symbolic attacker, writing up a full, computationally sound proof (and identifying new attacks) for the NSL protocol when agents can run both roles, including running sessions with themselves. The proof is first attempted without any assumption other than that the encryption is CCA2 and that honest names are assigned at the beginning (that is, absolutely nothing about parsing: triples may be independent from pairs, pairing the projection of pairs may not give back the original item etc.). Along the way, we identified new attacks absent of some necessary parsing properties that implementations may not satisfy in general. Then with these additional parsing properties added to the properties satisfied by the implementation, we verified the protocol, namely secrecy, authentication and agreement. The project included graphical representation of the proof steps and the attacks. Type-flaw attacks that can be found in the literature have been reproduced this way, but a number of other attacks have also be revealed that cannot be found with the Dolev-Yao technique, and have not been found by other computational techniques either, although they are realistic. This is joint work with Pedro Adao of IST Lisbon. We hope to publish parts of this work to illustrate proving strategies. The current state of the writeup is available at http://prosecco.gforge.inria.fr/personal/gebana/nsl-long-both-roles.pdf

6.5. Authentication Attacks against Transport Layer Security

Participants: Karthikeyan Bhargavan [correspondant], Antoine Delignat-Lavaud, Cedric Fournet [Microsoft Research], Markulf Kohlweiss [Microsoft Research], Alfredo Pironti, Pierre-Yves Strub [IMDEA].

We discovered an important client impersonation attack on the Transport Layer Security protocol called the *triple handshake attack.* The attack is on the standard protocol and hence all compliant implementations were potentially at risk. Hence, we systematically followed responsible disclosure by notifying all major web browsers and TLS implementors, and then working with the TLS working group to design a countermeasure. The research results of this work were published at IEEE S&P [53].

To TLS implementors, we proposed short-term countermeasures that mitigated our attack, leading to security updates to all major web browsers: Google Chrome (CVE-2013-6628), Mozilla Firefox (CVE-2014-1491), Internet Explorer (CVE-2014-1771), Apple Safari (CVE-2014-1295), as well as to non-browser TLS libraries such as Oracle JSSE (CVE-2014-6457) and RSA BSAFE (CVE-2014-4630). For more details, see http://secure-resumption.com

To the TLS working group, we proposed a new cryptographic construction called the *session hash* that fundamentally alters the cryptographic core of TLS. This construction has now been adopted as a protocol extension to TLS 1.2 and has been integrated into the upcoming TLS 1.3. We expect an IETF standard for this construction to be published in early 2015.

While the triple handshake attacks primarily affect client-authentication, server authentication in HTTPS (HTTP over TLS) primarily relies on X.509 public key certificates. Antoine Delignat-Lavaud along with co-authors at Microsoft research published a paper at NDSS 2015 on a large-scale study of the Web PKI: how certificates are issued and used on the web [56]. Our work uncovered many unsafe practices and suggested best practices and new security policies.

Antoine Delignat-Lavaud also showed how the unsafe sharing of certificates across multiple websites could be exploited to fully compromise the same origin policty for websites, using an vulnerability called virtual host confusion. These results were discussed in a talk at BlackHat USA: for details see http://bh.ht.vc. A research paper on these attacks is forthcoming at WWW'2015.

6.6. A Verified Reference Implementation of Transport Layer Security

Participants: Benjamin Beurdouche [correspondant], Karthikeyan Bhargavan [correspondant], Antoine Delignat-Lavaud, Cedric Fournet [Microsoft Research], Markulf Kohlweiss [Microsoft Research], Alfredo Pironti, Pierre-Yves Strub [IMDEA], Santiago Zanella-Béguelin [Microsoft Research], Jean Karim Zinzindohoue.

Following on from previous work in the miTLS project, we published new versions of miTLS (http://mitls.org) that implemented various protocol extensions including the new session hash extension.

At CRYPTO 2014 [55], we published the first detailed cryptographic proof of an implementation of the TLS Handshake. The implementation consists of about 5000 lines of code and is equipped with about 2500 lines of security annotations written in F7, and a 3000 line EasyCrypt proof.

Currently, we are extending and improving this verified implementation to cover commonly used TLS extensions as well as TLS 1.3, the new version of TLS that we are actively involved in designing. We recently hosted a meeting of the TLS working group at Inria in Paris and are active members of the core working group.

In parallel, we have been analyzing other implementations of TLS and testing them against our implementation, both to ensure interoperability and to uncover bugs. Our analyses have led to the discovery of serious state machine vulnerabilities in many TLS implementations including Oracle JSSE, NSS, OpenSSL, SecureTransport, CyaSSL, Mono, and RSA BSAFE. On our recommendations, all these TLS libraries have issued important security updates in 2014.

6.7. Verified implementations of cryptographic primitives

Participants: Evmorfia-Iro Bartzia, Jean Karim Zinzindohoue, Pierre-Yves Strub, Karthikeyan Bhargavan.

Cryptographic libraries underpin the security of all security protocol implementations. A bug in the implementation of one primitive could enable an attacker to break the security of the full protocol. Hence, establishing the formal correctness of an efficient cryptographic mechanism is a much-desired but still open goal. We are investigating two directions of research towards this goal, specifically in the context of elliptic curve libraries.

Evmorfia-Iro Bartzia and Pierre-Yves Strub are building a Coq library that enables the precise proof of elliptic curve algorithms, and the automatic extraction of verified OCaml code that implements these algorithms. Their most recent result is the formal proof of a non-trivial theorem by Picard: the existence of an isomorphism between an elliptic curve and its Picard group of divisors. This work led to the publication "A formal library for Elliptic Curves in the Coq proof Assistant" and was presented at the ITP 2014 conference [51]. We have also been working on a formal proof of correctness of the GLV algorithm for scalar multiplication in Coq, using the above development and the CoqEal methodology. At present, we have an implementation of the algorithm in the OCaml language and a formal development regarding multiexponentiation, endomorphisms, scalar decomposition and coordinates in both affine and projective spaces. This work is still in progress.

Jean Karim Zinzindohoue and Karthikeyan Bhargavan are investigating the direct verification of implementations of the Curve25519 elliptic curve that is emerging as the preferred new curve for a variety of cryptographic standards, including TLS and the W3C web cryptography API. We use standard program verification tools such as the Frama-C/Why3 verification toolkit for a C implementation of Curve25519 and the F* typechecker for an OCaml implementation of the curve. This work is still in progress.

6.8. Dynamic Security Verification and Testing

Participants: Catalin Hritcu, Arthur Azevedo de Amorim, Zoi Paraskevopoulou, Nikolaos Giannarakis.

We investigated two directions in the runtime security verification of software and hardware systems.

Catalin Hritcu, Arthur Azevedo de Amorim, Nick Giannarakis, and their collaborators at University of Pennsylvania and Portland State University published work on *micro-policies* a generic framework for defining tag-based reference monitors on a simple tagged RISC processor. The framework was formalized and verified in the Coq proof assistant and was used to define and verify micro-policies for dynamic sealing, control-flow integrity, compartmentalization, and memory safety. This work resulted in publications at POPL 2014 [63], ASPLOS 2015 [58], and another paper is in submission.

Catalin Hritcu along with his co-authors worked on a testing framework for security and functional correctness. We published a journal paper about testing noninterference [68] and submitted an ANR JCJC grant pre-proposal on the whole project. Catalin Hritcu also worked with an intern Zoe Paraskevopoulou on this topic, who successfully defended her thesis at NTU Athens. We plan to publish a polished version of that in the near future.

6.9. Verified Security for Web Applications

Participants: Karthikeyan Bhargavan [correspondant], Chetan Bansal [Microsoft], Antoine Delignat-Lavaud, Sergio Maffeis [Imperial College London].

Karthikeyan Bhargavan, Antoine Delignat-Lavaud, and co-authors published a tutorial on Defensive JavaScript, a typed subset of JavaScript that is designed to be used for security-critical components such as cryptographic libraries that may be deployed within untrusted web pages. This tutorial was published as a follow-up of Karthikeyan Bhargavan's lectures at the FOSAD'13 summer school [65].

Karthikeyan Bhargavan, Antoine Delignat-Lavaud, and co-authors also published a journal version of their work on the WebSpi web security modeling library [47], one of the few formal models that captures the detailed security assumptions of various web mechanisms.

Karthikeyan Bhargavan along with collaborators at Microsoft Research published a paper at POPL 2014 on TS*: a new gradual type system for a large subset of JavaScript [47]. We showed how to compile and safely deploy well-typed TS* programs as standard JavaScript in websites. Such programs preserve their types even if other code running on the website is malicious. Our work was used as a basis for further work on the TypeScript compiler and typechecker developed at Microsoft.

6.10. Electronic Voting and Auctions

Participants: Benjamin Smyth [correspondant], Elizabeth Quaglia, Adam Mccarthy, David Bernhard.

Benjamin Smyth continued his work on proving privacy properties of electronic voting protocols. Smyth and Bernhard worked on a new formal definition of ballot secrecy that works even if the bulletin board (used to publish votes) is malicious [69].

Benjamin Smyth, Elizabeth Quaglia, and Adam McCarthy observed that existing electronic voting schemes could be used as core building blocks for electronic auction protocols. Using this link, they build two new e-auction protocols Hawk and Aucitas by building on top of the e-voting protocols Helios and Civitas resp. They prove that their protocols enjoy many desired security properties. This result was published at Financial Cryptography 2014 [61].