RESEARCH CENTER

FIELD
Algorithmics, Programming, Software and Architecture

Activity Report 2016

Section New Results

Edition: 2017-08-25

ARIC Project-Team

6. New Results

6.1. Floating-point arithmetic

6.1.1. Parallel floating-point expansions for extended-precision GPU computations

GPUs are an important hardware development platform for problems where massive parallel computations are needed. Many of these problems require a higher precision than the standard double floating-point (FP) available. One common way of extending the precision is the multiple-component approach, in which real numbers are represented as the unevaluated sum of several standard machine precision FP numbers. This representation is called an FP expansion and it offers the simplicity of using directly available and highly optimized FP operations. In [30] we present new data-parallel algorithms for adding and multiplying FP expansions specially designed for extended precision computations on GPUs. These are generalized algorithms that can manipulate FP expansions of different sizes (from double-double up to a few tens of doubles) and ensure a certain worst case error bound on the results.

6.1.2. Error analysis of the Cornea-Harrison-Tang method

Assuming floating-point arithmetic with a fused multiply-add operation and rounding to nearest, the Cornea-Harrison-Tang method aims to evaluate expressions of the form $ab + cd$ with high relative accuracy. In [12] we provide a rounding error analysis of this method, which unlike previous studies is not restricted to binary floating-point arithmetic but holds for any radix β. We show first that an asymptotically optimal bound on the relative error of this method is $\frac{2\beta u+2u^2}{\beta-2u^2} = 2u + \frac{2}{\beta}u^2 + O(u^3)$, where $u = \frac{1}{2}\beta^{1-p}$ is the unit roundoff in radix β and precision p. Then we show that the possibility of removing the $O(u^2)$ term from this bound is governed by 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 $2u + \frac{2}{\beta}u^2 - 4u^3$. All these results hold provided underflows and overflows do not occur and under some mild assumptions on β and p satisfied by IEEE 754-2008 formats.

6.1.3. Sharp error bounds for complex floating-point inversion

In [14] we study the accuracy of the classic algorithm for inverting a complex number given by its real and imaginary parts as floating-point numbers. Our analyses are done in binary floating-point arithmetic, with an unbounded exponent range and in precision p; we also assume that the basic arithmetic operations ($+$, $-$, $\times$, $/$) are rounded to nearest, so that the unit roundoff is $u = 2^{-p}$. We bound the largest relative error in the computed inverse either in the componentwise or in the normwise sense. We prove the componentwise relative error bound $3u$ for the complex inversion algorithm (assuming $p \geq 4$), and we show that this bound is asymptotically optimal (as $p \to \infty$) when p is even, and sharp when using one of the basic IEEE 754 binary formats with an odd precision ($p = 53, 113$). This componentwise bound obviously leads to the same bound $3u$ for the normwise relative error. However, we prove that the smaller bound $2.707131u$ holds (assuming $p \geq 24$) for the normwise relative error, and we illustrate the sharpness of this bound for the basic IEEE 754 binary formats ($p = 24, 53, 113$) using numerical examples.

6.1.4. On relative errors of floating-point operations: optimal bounds and applications

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 fl and barring underflow and overflow, such models bound the relative errors $E_1(t) = |t - \mathrm{fl}(t)|/|t|$ and $E_(t) = |t - \mathrm{fl}(t)|/|\mathrm{fl}(t)|$ by the unit roundoff u. With S. M. Rump (Hamburg University of Technology), we investigate in [15] the possibility and the usefulness 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. We show that $E_1(t)$ and $E_2(t)$ are optimally bounded by $u/(1+u)$ and u, respectively, when t is real or, under mild assumptions on the base and the precision, when $t = x \pm y$ or $t = xy$ with x, y two floating-point numbers. We prove that while this remains true for division in base $\beta > 2$, smaller, attainable bounds can be derived for both division in base $\beta = 2$ and square root. This set of optimal bounds is then applied to the rounding error analysis of various numerical algorithms: in all cases, we obtain significantly shorter proofs of the best-known error bounds for such algorithms, and/or improvements on these bounds themselves.

6.1.5. Computing floating-point logarithms with fixed-point operations

Elementary functions from the mathematical library input and output floating-point numbers. However, it is possible to implement them purely using integer/fixed-point arithmetic. This option was not attractive between 1985 and 2005, because mainstream processor hardware supported 64-bit floating-point, but only 32-bit integers. Besides, conversions between floating-point and integer were costly. This has changed in recent years, in particular with the generalization of native 64-bit integer support. The purpose of this article is therefore to reevaluate the relevance of computing floating-point functions in fixed-point. For this, several variants of the double-precision logarithm function are implemented and evaluated. Formulating the problem as a fixed-point one is easy after the range has been (classically) reduced. Then, 64-bit integers provide slightly more accuracy than 53-bit mantissa, which helps speed up the evaluation. Finally, multi-word arithmetic, critical for accurate implementations, is much faster in fixed-point, and natively supported by recent compilers. Novel techniques of argument reduction and rounding test are introduced in this context. Thanks to all this, a purely integer implementation of the correctly rounded double-precision logarithm outperforms the previous state of the art, with the worst-case execution time reduced by a factor 5. This work also introduces variants of the logarithm that input a floating-point number and output the result in fixed-point. These are shown to be both more accurate and more efficient than the traditional floating-point functions for some applications [35].

6.1.6. A library for symbolic floating-point arithmetic

To analyze a priori the accuracy of an algorithm in floating-point arithmetic, one usually derives a uniform error bound on the output, valid for most inputs and parametrized by the precision p. To show further that this bound is sharp, a common way is to build an input example for which the error committed by the algorithm comes close to that bound, or even attains it. Such inputs may be given as floating-point numbers in one of the IEEE standard formats (say, for $p = 53$) or, more generally, as expressions parametrized by p, that can be viewed as symbolic floating-point numbers. With such inputs, a sharpness result can thus be established for virtually all reasonable formats instead of just one of them. This, however, requires the ability to run the algorithm on those inputs and, in particular, to compute the correctly-rounded sum, product, or ratio of two symbolic floating-point numbers. We show in [61] how these basic arithmetic operations can be performed automatically. We introduce a way to model symbolic floating-point data, and present algorithms for round-to-nearest addition, multiplication, fused multiply-add, and division. An implementation as a Maple library is also described, and experiments using examples from the literature are provided to illustrate its interest in practice.

6.1.7. On the robustness of the 2Sum and Fast2Sum algorithms

The 2Sum and Fast2Sum algorithms are important building blocks in numerical computing. They are used (implicitly or explicitly) in many *compensated* algorithms (such as compensated summation or compensated polynomial evaluation). They are also used for manipulating floating-point *expansions*. We show in [56] that these algorithms are much more robust than it is usually believed: the returned result makes sense even when the rounding function is not round-to-nearest, and they are almost immune to overflow.

6.1.8. Tight and rigourous error bounds for basic building blocks of double-word arithmetic

In [63] we analyze several classical basic building blocks of double-word arithmetic (frequently called "double-double arithmetic" in the literature): the addition of a double-word number and a floating-point number, the addition of two double-word numbers, the multiplication of a double-word number by a floating-point number, the multiplication of two double-word numbers, the division of a double-word number by a

floating-point number, and the division of two double-word numbers. For multiplication and division we get better relative error bounds than the ones previously published. For addition of two double-word numbers, we show that the previously published bound was wrong, and we provide a relative error bound. We introduce new algorithms for division. We also give examples that illustrate the tightness of our bounds.

6.1.9. A new multiplication algorithm for extended precision using floating-point expansions

Some important computational problems must use a floating-point (FP) precision several times higher than the hardware-implemented available one. These computations critically rely on software libraries for high-precision FP arithmetic. The representation of a high-precision data type crucially influences the corresponding arithmetic algorithms. Recent work showed that algorithms for FP expansions, that is, a representation based on unevaluated sum of standard FP types, benefit from various high-performance support for native FP, such as low latency, high throughput, vectorization, threading, etc. Bailey's QD library and its corresponding Graphics Processing Unit (GPU) version, GQD, are such examples. Despite using native FP arithmetic as the key operations, QD and GQD algorithms are focused on double-double or quad-double representations and do not generalize efficiently or naturally to a flexible number of components in the FP expansion. In [45] we introduce a new multiplication algorithm for FP expansion with flexible precision, up to the order of tens of FP elements in mind. The main feature consists in the partial products being accumulated in a special designed data structure that has the regularity of a fixed-point representation while allowing the computation to be naturally carried out using native FP types. This allows us to easily avoid unnecessary computation and to present rigorous accuracy analysis transparently. The algorithm, its correctness and accuracy proofs and some performance comparisons with existing libraries are all contributions of this paper.

6.1.10. CAMPARY: Cuda Multiple Precision Arithmetic Library and Applications

Many scientific computing applications demand massive numerical computations on parallel architectures such as Graphics Processing Units (GPUs). Usually, either floating-point single or double precision arithmetic is used. Higher precision is generally not available in hardware, and software extended precision libraries are much slower and rarely supported on GPUs. We develop CAMPARY: a multiple-precision arithmetic library, using the CUDA programming language for the NVidia GPU platform. In our approach, the precision is extended by representing real numbers as the unevaluated sum of several standard machine precision floating-point numbers. We make use of error-free transforms algorithms, which are based only on native precision operations, but keep track of all rounding errors that occur when performing a sequence of additions and multiplications. This offers the simplicity of using hardware highly optimized floating-point operations, while also allowing for rigorously proven rounding error bounds. This also allows for easy implementation of an interval arithmetic. Currently, all basic multiple-precision arithmetic operations are supported. Our target applications are in chaotic dynamical systems or automatic control [34].

6.1.11. Arithmetic algorithms for extended precision using floating-point expansions

Many numerical problems require a higher computing precision than the one offered by standard floating-point (FP) formats. One common way of extending the precision is to represent numbers in a *multiple component* format. By using the so-called *floating-point expansions*, real numbers are represented as the unevaluated sum of standard machine precision FP numbers. This representation offers the simplicity of using directly available, hardware implemented and highly optimized, FP operations. It is used by multiple-precision libraries such as Bailey's QD or the analogue Graphics Processing Units (GPU) tuned version, GQD. In this article we briefly revisit algorithms for adding and multiplying FP expansions, then we introduce and prove new algorithms for normalizing, dividing and square rooting of FP expansions. The new method used for computing the reciprocal a^{-1} and the square root $\sqrt{a}$ of an FP expansion a is based on an adapted Newton-Raphson iteration where the intermediate calculations are done using "truncated" operations (additions, multiplications) involving FP expansions. We give here a thorough error analysis showing that it allows very accurate computations. More precisely, after q iterations, the computed FP expansion $x = x_0 + \cdots + x_{2^q-1}$ satisfies, for the reciprocal algorithm, the relative error bound: $|(x - a^{-1})/a^{-1}| \leq 2^{-2^q(p-3)-1}$ and, respectively, for the square root one: $|x - 1/\sqrt{a}| \leq 2^{-2^q(p-3)-1}/\sqrt{a}$, where $p > 2$ is the precision of the FP representation used ($p = 24$ for single precision and $p = 53$ for double precision) [16].

6.1.12. Comparison between binary and decimal floating-point numbers

We 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. We provide an implementation of several variants of our algorithm, and compare them [8].

6.1.13. Automatic source-to-source error compensation of floating-point programs: code synthesis to optimize accuracy and time

Numerical programs with IEEE 754 floating-point computations may suffer from inaccuracies, since finite precision arithmetic is an approximation of real arithmetic. Solutions that reduce the loss of accuracy are available, such as compensated algorithms or double-double precision floating-point arithmetic. With Ph. Langlois and M. Martel (LIRMM and Université de Perpignan), we show in [21] how to automatically improve the numerical quality of a numerical program with the smallest impact on its performance. We define and implement source code transformations in order to derive automatically compensated programs. We present several experimental results to compare the transformed programs and existing solutions. The transformed programs are as accurate and efficient as the implementations of compensated algorithms when the latter exist. Furthermore, we propose some transformation strategies allowing us to improve partially the accuracy of programs and to tune the impact on execution time. Trade-offs between accuracy and performance are assured by code synthesis. Experimental results show that user-defined trade-offs are achievable in a reasonable amount of time, with the help of the tools we present here.

6.1.14. Correctly rounded arbitrary-precision floating-point summation

We have designed a fast, low-level algorithm to compute the correctly rounded summation of several floating-point numbers in arbitrary precision in radix 2, each number (each input and the output) having its own precision. We have implemented it in GNU MPFR; it will be part of the next MPFR major release (GNU MPFR 4.0). In addition to a pen-and-paper proof, various kinds of tests are provided. Timings show that this new algorithm/implementation is globally much faster and takes less memory than the previous one (from MPFR 3.1.5): the worst-case time and memory complexity was exponential and it is now polynomial. Timings on pseudo-random inputs with various sets of parameters also show that this new implementation is even much faster than the (inaccurate) basic sum implementation in some cases. [36], [65]

6.2. Lattices: algorithms and cryptology

6.2.1. Zero-Knowledge Arguments for Lattice-Based Accumulators: Logarithmic-Size Ring Signatures and Group Signatures Without Trapdoors

An accumulator is a function that hashes a set of inputs into a short, constant-size string while preserving the ability to efficiently prove the inclusion of a specific input element in the hashed set. It has proved useful in the design of numerous privacy-enhancing protocols, in order to handle revocation or simply prove set membership. In the lattice setting, currently known instantiations of the primitive are based on Merkle trees, which do not interact well with zero-knowledge proofs. In order to efficiently prove the membership of some element in a zero-knowledge manner, the prover has to demonstrate knowledge of a hash chain without revealing it, which is not known to be efficiently possible under well-studied hardness assumptions. In [39], we provide an efficient method of proving such statements using involved extensions of Stern's protocol. Under the Small Integer Solution assumption, we provide zero-knowledge arguments showing possession of a hash chain. As an application, [39] describes new lattice-based group and ring signatures in the random oracle model. In particular, the paper obtains: (i) The first lattice-based ring signatures with logarithmic size in the cardinality of the ring; (ii) The first lattice-based group signature that does not require any GPV trapdoor and thus allows for a much more efficient choice of parameters.

6.2.2. A Lattice-Based Group Signature Scheme with Message-Dependent Opening

Group signatures are an important anonymity primitive allowing users to sign messages while hiding in a crowd. At the same time, signers remain accountable since an authority is capable of de-anonymizing signatures via a process called opening. In many situations, this authority is granted too much power as it can identify the author of any signature. Sakai et al. proposed a flavor of the primitive, called Group Signature with Message-Dependent Opening (GS-MDO), where opening operations are only possible when a separate authority (called "admitter") has revealed a trapdoor for the corresponding message. So far, all existing GS-MDO constructions rely on bilinear maps, partially because the message-dependent opening functionality inherently implies identity-based encryption. In [40], the team proposes the first GS-MDO candidate based on lattice assumptions. The construction combines the group signature of Ling, Nguyen and Wang (PKC'15) with two layers of identity-based encryption. These components are tied together using suitable zero-knowledge argument systems.

6.2.3. Practical "Signatures with Efficient Protocols" from Simple Assumptions

Digital signatures are perhaps the most important base for authentication and trust relationships in large scale systems. More specifically, various applications of signatures provide privacy and anonymity preserving mechanisms and protocols, and these, in turn, are becoming critical (due to the recently recognized need to protect individuals according to national rules and regulations). A specific type of signatures called "signatures with efficient protocols", as introduced by Camenisch and Lysyanskaya (CL), efficiently accommodates various basic protocols and extensions like zero-knowledge proofs, signing committed messages, or re-randomizability. These are, in fact, typical operations associated with signatures used in typical anonymity and privacy-preserving scenarios. To date there are no "signatures with efficient protocols" which are based on simple assumptions and truly practical. These two properties assure us a robust primitive: First, simple assumptions are needed for ensuring that this basic primitive is mathematically robust and does not require special ad hoc assumptions that are more risky, imply less efficiency, are more tuned to the protocol itself, and are perhaps less trusted. In the other dimension, efficiency is a must given the anonymity applications of the protocol, since without proper level of efficiency the future adoption of the primitives is always questionable (in spite of their need). In [41], the team presents a new CL-type signature scheme that is re-randomizable under a simple, well-studied, and by now standard, assumption (SXDH). The signature is efficient (built on the recent QA-NIZK constructions), and is, by design, suitable to work in extended contexts that typify privacy settings (like anonymous credentials, group signature, and offline e-cash). The paper demonstrates its power by presenting practical protocols based on it.

6.2.4. Functional Commitment Schemes: From Polynomial Commitments to Pairing-Based Accumulators from Simple Assumptions

In [42], the team formalizes a cryptographic primitive called functional commitment (FC) which can be viewed as a generalization of vector commitments (VCs), polynomial commitments and many other special kinds of commitment schemes. A non-interactive functional commitment allows committing to a message in such a way that the committer has the flexibility of only revealing a function $F(M)$ of the committed message during the opening phase. We provide constructions for the functionality of linear functions, where messages consist of a vectors of n elements over some domain D (e.g., $m = (m_1, ..., m_n) \in D_n$) and commitments can later be opened to a specific linear function of the vector coordinates. An opening for a function $F : D_n \rightarrow R$ thus generates a witness for the fact that $F(m)$ indeed evaluates to $y \in R$. One security requirement is called function binding and requires that no adversary be able to open a commitment to two different evaluations y, y' for the same function F. The paper [42] proposes a construction of functional commitment for linear functions based on constant-size assumptions in composite order groups endowed with a bilinear map. The construction has commitments and openings of constant size (i.e., independent of n or function description) and is perfectly hiding – the underlying message is information theoretically hidden. Our security proofs builds on the Déjà Q framework of Chase and Meiklejohn (Eurocrypt 2014) and its extension by Wee (TCC 2016) to encryption primitives, thus relying on constant-size subgroup decisional assumptions. The paper shows that the FC for linear functions are sufficiently powerful to solve four open problems. They, first, imply

polynomial commitments, and, then, give cryptographic accumulators (i.e., an algebraic hash function which makes it possible to efficiently prove that some input belongs to a hashed set). In particular, specializing the new FC construction leads to the first pairing-based polynomial commitments and accumulators for large universes known to achieve security under simple assumptions. We also substantially extend our pairing-based accumulator to handle subset queries which requires a non-trivial extension of the Déjà Q framework.

6.2.5. *Fully Secure Functional Encryption for Inner Products, from Standard Assumptions*

Functional encryption is a modern public-key paradigm where a master secret key can be used to derive sub-keys SKF associated with certain functions F in such a way that the decryption operation reveals $F(M)$, if M is the encrypted message, and nothing else. Recently, Abdalla *et al.* gave simple and efficient realizations of the primitive for the computation of linear functions on encrypted data: given an encryption of a vector y over some specified base ring, a secret key SK_x for the vector x allows computing $\langle x, y\rangle$. Their technique surprisingly allows for instantiations under standard assumptions, like the hardness of the Decision Diffie-Hellman (DDH) and Learning-with-Errors (LWE) problems. Their constructions, however, are only proved secure against selective adversaries, which have to declare the challenge messages M_0 and M_1 at the outset of the game. In [22], we provide constructions that provably achieve security against more realistic adaptive attacks (where the messages M_0 and M_1 may be chosen in the challenge phase, based on the previously collected information) for the same inner product functionality. The constructions of [22] are obtained from hash proof systems endowed with homomorphic properties over the key space. They are (almost) as efficient as those of Abdalla *et al.* and rely on the same hardness assumptions. In addition, the paper [22] obtains a solution based on Paillier's composite residuosity assumption, which was an open problem even in the case of selective adversaries. We also propose LWE-based schemes that allow evaluation of inner products modulo a prime p, as opposed to the schemes of Abdalla et al. that are restricted to evaluations of integer inner products of short integer vectors. The paper [22] finally proposes a solution based on Paillier's composite residuosity assumption that enables evaluation of inner products modulo an RSA integer $N = pq$. The paper [22] demonstrates that the functionality of inner products over a prime field is powerful and can be used to construct bounded collusion FE for all circuits.

6.2.6. *Signature Schemes with Efficient Protocols and Dynamic Group Signatures from Lattice Assumptions*

A recent line of works – initiated by Gordon, Katz and Vaikuntanathan (Asiacrypt 2010) – gave lattice-based realizations of privacy-preserving protocols allowing users to authenticate while remaining hidden in a crowd. Despite five years of efforts, known constructions remain limited to static populations of users, which cannot be dynamically updated. For example, none of the existing lattice-based group signatures seems easily extendable to the more realistic setting of dynamic groups. In [37], the team provides new tools enabling the design of anonymous authen-tication systems whereby new users can register and obtain credentials at any time. The first contribution of [37] is a signature scheme with efficient protocols, which allows users to obtain a signature on a committed value and subsequently prove knowledge of a signature on a committed message. This construction, which builds on the lattice-based signature of Böhl *et al.* (Eurocrypt'13), is well-suited to the design of anonymous credentials and dynamic group signatures. As a second technical contribution, [37] provides a simple, round-optimal joining mechanism for introducing new members in a group. This mechanism consists of zero-knowledge arguments allowing registered group members to prove knowledge of a secret short vector of which the corresponding public syndrome was certified by the group manager. This method provides similar advantages to those of structure-preserving signatures in the realm of bilinear groups. Namely, it allows group members to generate their public key on their own without having to prove knowledge of the underlying secret key. This results in a two-round join protocol supporting concurrent enrollments, which can be used in other settings such as group encryption.

6.2.7. *Zero-Knowledge Arguments for Matrix-Vector Relations and Lattice-Based Group Encryption*

Group encryption (GE) is the natural encryption analogue of group signatures in that it allows verifiably encrypting messages for some anonymous member of a group while providing evidence that the receiver is

a properly certified group member. Should the need arise, an opening authority is capable of identifying the receiver of any ciphertext. As introduced by Kiayias, Tsiounis and Yung (Asiacrypt'07), GE is motivated by applications in the context of oblivious retriever storage systems, anonymous third parties and hierarchical group signatures. In [38], we provide the first realization of group encryption under lattice assumptions. The construction of [38] is proved secure in the standard model (assuming interaction in the proving phase) under the Learning-With-Errors (LWE) and Short-Integer-Solution (SIS) assumptions. As a crucial component of our system, [38] describes a new zero-knowledge argument system allowing to demonstrate that a given ciphertext is a valid encryption under some hidden but certified public key, which incurs to prove quadratic statements about LWE relations. Specifically, the protocol of [38] allows arguing knowledge of witnesses consisting of $X \in \mathbb{Z}_q^{m\times}$, $s \in \mathbb{Z}_q^n$ and a small-norm $e \in \mathbb{Z}^m$ which underlie a public vector $b = X \cdot s + e \in \mathbb{Z}_q^m$ while simultaneously proving that the matrix $X \in \mathbb{Z}_q^{m\times n}$ has been correctly certified.

6.2.8. Efficient Cryptosystems From 2^k-th Power Residue Symbols

Goldwasser and Micali (1984) highlighted the importance of randomizing the plaintext for public-key encryption and introduced the notion of semantic security. They also realized a cryptosystem meeting this security notion under the standard complexity assumption of deciding quadratic residuosity modulo a composite number. The Goldwasser-Micali cryptosystem is simple and elegant but is quite wasteful in bandwidth when encrypting large messages. A number of works followed to address this issue and proposed various modifications. In [4], we revisit the original Goldwasser-Micali cryptosystem using 2^k-th power residue symbols. The so-obtained cryptosystems appear as a very natural generalization for $k \geq 2$ (the case $k = 1$ corresponds exactly to the Goldwasser-Micali cryptosystem). Advantageously, they are efficient in both bandwidth and speed; in particular, they allow for fast decryption. Further, the cryptosystems described in this paper inherit the useful features of the original cryptosystem (like its homomorphic property) and are shown to be secure under a similar complexity assumption. As a prominent application, the paper [4] describes an efficient lossy trapdoor function based thereon.

6.2.9. 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 [17], we construct 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). The schemes of [17] 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, the new schemes [17] are erasure-free (reliable erasure is hard to assure and hard to administer properly in actual systems). To the best of our knowledge, 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).

6.2.10. Non-Zero Inner Product Encryption with Short Ciphertexts and Private Keys

In [28], the team describes two constructions of non-zero inner product encryption (NIPE) systems in the public index setting, both having ciphertexts and secret keys of constant size. Both schemes are obtained by tweaking the Boneh-Gentry-Waters broadcast encryption system (Crypto 2005) and are proved selectively secure without random oracles under previously considered assumptions in groups with a bilinear map. Our first realization builds on prime-order bilinear groups and is proved secure under the Decisional Bilinear Diffie-Hellman Exponent assumption, which is parameterized by the length n of vectors over which the inner product is defined. By moving to composite order bilinear groups, the paper [28] obtains security under static subgroup decision assumptions following the Déjà Q framework of Chase and Meiklejohn (Eurocrypt 2014) and its extension by Wee (TCC 2016). The schemes of [28] are the first NIPE systems to achieve such parameters, even in the selective security setting. Moreover, they are the first proposals to feature optimally short private keys, which only consist of one group element. The prime-order-group realization of [28] is also the first one with a deterministic key generation mechanism.

6.2.11. More Efficient Constructions for Inner-Product Encryptions

In [48], the team describes new constructions for inner product encryption (called IPE1 and IPE2), which are both secure under the eXternal Diffie-Hellman assumption (SXDH) in asymmetric pairing groups. The IPE1 scheme of [48] has constant-size ciphertexts whereas the second one is weakly attribute hiding. The second scheme is derived from the identity-based encryption scheme of Jutla and Roy (Asiacrypt 2013), that was extended from tag-based quasi-adaptive non-interactive zero-knowledge (QA-NIZK) proofs for linear subspaces of vector spaces over bilinear groups. The verifier common reference string (CRS) in these tag-based systems are split into two parts, that are combined during verification. The paper [48] considers an alternate form of the tag-based QA-NIZK proof with a single verifier CRS that already includes a tag, different from the one defining the language. The verification succeeds as long as the two tags are unequal. Essentially, we embed a two-equation revocation mechanism in the verification. The new QA-NIZK proof system leads to IPE1, a constant-sized ciphertext IPE scheme with very short ciphertexts. Both the IPE schemes are obtained by applying the n-equation revocation technique of Attrapadung and Libert (PKC 2010) to the corresponding identity based encryption schemes and proved secure under SXDH assumption. As an application, the paper [48] shows how the new schemes can be specialized to obtain the first fully secure identity-based broadcast encryption based on SXDH with a trade-off among the public parameters, ciphertext and key sizes, all of them being sub-linear in the maximum number of recipients of a broadcast.

6.2.12. Verifiable Message-Locked Encryption

One of today's main challenge related to cloud storage is to maintain the functionalities and the efficiency of customers' and service providers' usual environments, while protecting the confidentiality of sensitive data. Deduplication is one of those functionalities: it enables cloud storage providers to save a lot of memory by storing only once a file uploaded several times. But classical encryption blocks deduplication. One needs to use a "message-locked encryption" (MLE), which allows the detection of duplicates and the storage of only one encrypted file on the server, which can be decrypted by any owner of the file. However, in most existing scheme, a user can bypass this deduplication protocol. In [27], we provide servers verifiability for MLE schemes: the servers can verify that the ciphertexts are well-formed. This property that we formally define forces a customer to prove that she complied to the deduplication protocol, thus preventing her to deviate from *the prescribed functionality* of MLE. We call it *deduplication consistency*. To achieve this deduplication consistency, we provide (i) a generic transformation that applies to any MLE scheme and (ii) an ElGamal-based deduplication-consistent MLE, which is secure in the random oracle model.

6.2.13. Privately Outsourcing Exponentiation to a Single Server: Cryptanalysis and Optimal Constructions

In [29], we address the problem of speeding up group computations in cryptography using a single untrusted computational resource. We analyze the security of an efficient protocol for securely outsourcing multi-exponentiations proposed at ESORICS 2014. We show that this scheme does not achieve the claimed security

guarantees and we present several practical polynomial-time attacks on the delegation protocol which allows the untrusted helper to recover part (or the whole) of the device secret inputs. We then provide simple constructions for outsourcing group exponentiations in different settings (e.g. public/secret, fixed/variable bases and public/secret exponents). Finally, we prove that our attacks on the ESORICS 2014 protocol are unavoidable if one wants to use a single untrusted computational resource and to limit the computational cost of the limited device to a constant number of (generic) group operations. In particular, we show that our constructions are actually optimal.

6.3. Algebraic computing and high-performance kernels

6.3.1. Algebraic Diagonals and Walks: Algorithms, Bounds, Complexity

The diagonal of a multivariate power series F is the univariate power series $\mathrm{Diag}(F)$ generated by the diagonal terms of F. Diagonals form an important class of power series; they occur frequently in number theory, theoretical physics and enumerative combinatorics. We study algorithmic questions related to diagonals in the case where F is the Taylor expansion of a bivariate rational function. It is classical that in this case $\mathrm{Diag}(F)$ is an algebraic function. We propose an algorithm that computes an annihilating polynomial for $\mathrm{Diag}(F)$. We give a precise bound on the size of this polynomial and show that generically, this polynomial is the minimal polynomial and that its size reaches the bound. The algorithm runs in time quasi-linear in this bound, which grows exponentially with the degree of the input rational function. We then address the related problem of enumerating directed lattice walks. The insight given by our study leads to a new method for expanding the generating power series of bridges, excursions and meanders. We show that their first N terms can be computed in quasi-linear complexity in N, without first computing a very large polynomial equation [6].

6.3.2. Multiple Binomial Sums

Multiple binomial sums form a large class of multi-indexed sequences, closed under partial summation, which contains most of the sequences obtained by multiple summation of products of binomial coefficients and also all the sequences with algebraic generating function. We study the representation of the generating functions of binomial sums by integrals of rational functions. The outcome is twofold. Firstly, we show that a univariate sequence is a multiple binomial sum if and only if its generating function is the diagonal of a rational function. Secondly, we propose algorithms that decide the equality of multiple binomial sums and that compute recurrence relations for them. In conjunction with geometric simplifications of the integral representations, this approach behaves well in practice. The process avoids the computation of certificates and the problem of the appearance of spurious singularities that afflicts discrete creative telescoping, both in theory and in practice [7].

6.3.3. Fast and Accurate Computation of Orbital Collision Probability for Short-Term Encounters

We provide a new method for computing the probability of collision between two spherical space objects involved in a short-term encounter under Gaussian-distributed uncertainty. In this model of conjunction, classical assumptions reduce the probability of collision to the integral of a two-dimensional Gaussian probability density function over a disk. The computational method is based on an analytic expression for the integral, derived by use of Laplace transform and D-finite functions properties. The formula has the form of a product between an exponential term and a convergent power series with positive coefficients. Analytic bounds on the truncation error are also derived and are used to obtain a very accurate algorithm. Another contribution is the derivation of analytic bounds on the probability of collision itself, allowing for a very fast and — in most cases — very precise evaluation of the risk. The only other analytical method of the literature — based on an approximation — is shown to be a special case of the new formula. A numerical study illustrates the efficiency of the proposed algorithms on a broad variety of examples and favorably compares the approach to the other methods of the literature [20].

6.3.4. Efficient Algorihtms for Mixed Creative Telescoping

Creative telescoping is a powerful computer algebra paradigm — initiated by Doron Zeilberger in the 90's — for dealing with definite integrals and sums with parameters. We address the mixed continuous-discrete case, and focus on the integration of bivariate hypergeometric-hyperexponential terms. We design a new creative telescoping algorithm operating on this class of inputs, based on a Hermite-like reduction procedure. The new algorithm has two nice features: it is efficient and it delivers, for a suitable representation of the input, a minimal-order telescoper. Its analysis reveals tight bounds on the sizes of the telescoper it produces [26].

6.3.5. Symbolic-Numeric Tools for Analytic Combinatorics in Several Variables

Analytic combinatorics studies the asymptotic behaviour of sequences through the analytic properties of their generating functions. This article provides effective algorithms required for the study of analytic combinatorics in several variables, together with their complexity analyses. Given a multivariate rational function we show how to compute its smooth isolated critical points, with respect to a polynomial map encoding asymptotic behaviour, in complexity singly exponential in the degree of its denominator. We introduce a numerical Kronecker representation for solutions of polynomial systems with rational coefficients and show that it can be used to decide several properties (0 coordinate, equal coordinates, sign conditions for real solutions, and vanishing of a polynomial) in good bit complexity. Among the critical points, those that are minimal—a property governed by inequalities on the moduli of the coordinates—typically determine the dominant asymptotics of the diagonal coefficient sequence. When the Taylor expansion at the origin has all non-negative coefficients (known as the 'combinatorial case') and under regularity conditions, we utilize this Kronecker representation to determine probabilistically the minimal critical points in complexity singly exponential in the degree of the denominator, with good control over the exponent in the bit complexity estimate. Generically in the combinatorial case, this allows one to automatically and rigorously determine asymptotics for the diagonal coefficient sequence. Examples obtained with a preliminary implementation show the wide applicability of this approach [43].

6.3.6. Tableau sequences, open diagrams, and Baxter families

Walks on Young's lattice of integer partitions encode many objects of algebraic and combinatorial interest. Chen *et al.* established connections between such walks and arc diagrams. We show that walks that start at $\varnothing$, end at a row shape, and only visit partitions of bounded height are in bijection with a new type of arc diagram — open diagrams. Remarkably, two subclasses of open diagrams are equinumerous with well known objects: standard Young tableaux of bounded height, and Baxter permutations. We give an explicit combinatorial bijection in the former case, and a generating function proof and new conjecture in the second case [9].

6.3.7. On 3-dimensional lattice walks confined to the positive octant

Many recent papers deal with the enumeration of 2-dimensional walks with prescribed steps confined to the positive quadrant. The classification is now complete for walks with steps in $\{0,\pm 1\}^2$: the generating function is differentially finite if and only if a certain group associated with the step set is finite. We explore in this paper the analogous problem for 3-dimensional walks confined to the positive octant. The first difficulty is their number: we have to examine no less than 11074225 step sets in $\{0,\pm 1\}^3$ (instead of 79 in the quadrant case). We focus on the 35548 that have at most six steps. We apply to them a combined approach, first experimental and then rigorous. On the experimental side, we try to guess differential equations. We also try to determine if the associated group is finite. The largest finite groups that we find have order 48 — the larger ones have order at least 200 and we believe them to be infinite. No differential equation has been detected in those cases. On the rigorous side, we apply three main techniques to prove D-finiteness. The algebraic kernel method, applied earlier to quadrant walks, works in many cases. Certain, more challenging, cases turn out to have a special Hadamard structure, which allows us to solve them via a reduction to problems of smaller dimension. Finally, for two special cases, we had to resort to computer algebra proofs. We prove with these techniques all the guessed differential equations. This leaves us with exactly 19 very intriguing step sets for which the group is finite, but the nature of the generating function still unclear [5].

6.3.8. Asymptotic Lattice Path Enumeration Using Diagonals

We consider d-dimensional lattice path models restricted to the first orthant whose defining step sets exhibit reflective symmetry across every axis. Given such a model, we provide explicit asymptotic enumerative formulas for the number of walks of a fixed length: the exponential growth is given by the number of distinct steps a model can take, while the sub-exponential growth depends only on the dimension of the underlying lattice and the number of steps moving forward in each coordinate. The generating function of each model is first expressed as the diagonal of a multivariate rational function, then asymptotic expressions are derived by analyzing the singular variety of this rational function. Additionally, we show how to compute subdominant growth, reflect on the difference between rational diagonals and differential equations as data structures for D-finite functions, and show how to determine first order asymptotics for the subset of walks that start and end at the origin [18].

6.3.9. Asymptotics of lattice walks via analytic combinatorics in several variables

We consider the enumeration of walks on the two-dimensional non-negative integer lattice with steps defined by a finite set $S \subset \{0, \pm 1\}^2$. Up to isomorphism there are 79 unique two-dimensional models to consider, and previous work in this area has used the kernel method, along with a rigorous computer algebra approach, to show that 23 of the 79 models admit D-finite generating functions. In 2009, Bostan and Kauers used Padé-Hermite approximants to guess differential equations which these 23 generating functions satisfy, in the process guessing asymptotics of their coefficient sequences. In this article we provide, for the first time, a complete rigorous verification of these guesses. Our technique is to use the kernel method to express 19 of the 23 generating functions as diagonals of tri-variate rational functions and apply the methods of analytic combinatorics in several variables (the remaining 4 models have algebraic generating functions and can thus be handled by univariate techniques). This approach also shows the link between combinatorial properties of the models and features of its asymptotics such as asymptotic and polynomial growth factors. In addition, we give expressions for the number of walks returning to the x-axis, the y-axis, and the origin, proving recently conjectured asymptotics of Bostan, Chyzak, van Hoeij, Kauers, and Pech [44].

6.3.10. Linear Time Interactive Certificates

With J.G. Dumas (LJK, Grenoble), E. Kaltofen (NCSU, USA), and E. Thomé (Inria Nancy) we work on interactive certificates. Computational problem certificates are additional data structures for each output, which can be used by a (possibly randomized) verification algorithm that proves the correctness of each output. In [32] we give a new certificate for the minimal polynomial of sparse or structured matrices whose Monte Carlo verification complexity requires a single matrix-vector multiplication and a linear number of extra field operations (sufficiently large cardinality field). We also propose a novel preconditioner that ensures irreducibility of the characteristic polynomial of the generically preconditioned matrix. This preconditioner takes linear time to be applied and uses only two random entries. We combine these two techniques to give algorithms that compute certificates for the determinant, and thus for the characteristic polynomial, whose Monte Carlo verification complexity is therefore also linear.

6.3.11. Computing minimal interpolation bases

With É. Schost (U. Waterloo, Canada), we consider the problem of computing minimal bases of solutions for a general interpolation problem, which encompasses Hermite-Padé approximation and constrained multivariate interpolation, and has applications in coding theory and security. The problem is classically solved using iterative algorithms based on recurrence relations. First, we discuss in [62] a fast, divide-and-conquer version of this recurrence, taking advantage of fast matrix computations over the scalars and over the polynomials. This new algorithm is deterministic, and for computing shifted minimal bases of relations between m vectors of size σ it uses $\widetilde{O}(m^{\omega-1}(\sigma + |s|))$ field operations, where the notation $\widetilde{O}(\cdot)$ indicates that logarithmic terms are omitted, $\omega \in [2, 2.38]$ is the exponent of matrix multiplication, and $|s|$ is the sum of the entries of the input shift s, with $\min(s) = 0$. This complexity bound improves in particular on earlier algorithms in the case of bivariate interpolation for soft decoding, while matching fastest existing algorithms for simultaneous Hermite-Padé approximation. Then we propose in [33] an algorithm for the computation of an interpolation

basis in shifted-Popov normal form with a cost of $\widetilde{O}(m^{\omega-1}\sigma)$ field operations. Previous works, in the case of Hermite-Padé approximation and in the general interpolation case, compute non-normalized bases. Since for arbitrary shifts such bases may have size $\Theta(m^2\sigma)$, the cost bound $\widetilde{O}(m^{\omega-1}\sigma)$ was feasible only with restrictive assumptions on the shift that ensure small output sizes. The question of handling arbitrary shifts with the same complexity bound was left open. To obtain the target cost for any shift, we strengthen the properties of the output bases, and of those obtained during the course of the algorithm: all the bases are computed in shifted Popov form, whose size is always $O(m\sigma)$. Then, we design a divide-and-conquer scheme. We recursively reduce the initial interpolation problem to sub-problems with more convenient shifts by first computing information on the degrees of the intermediate bases.

6.3.12. Fast computation of shifted Popov forms of polynomial matrices via systems of modular polynomial equations

In [46] we give a Las Vegas algorithm which computes the shifted Popov form of an $m \times m$ nonsingular polynomial matrix of degree d in expected $\widetilde{O}(m^\omega d)$ field operations, where ω is the exponent of matrix multiplication and $\widetilde{O}(\cdot)$ indicates that logarithmic factors are omitted. This is the first algorithm in $\widetilde{O}(m^\omega d)$ for shifted row reduction with arbitrary shifts. Using partial linearization, we reduce the problem to the case $d \leq \lceil \sigma/m \rceil$ where σ is the generic determinant bound, with σ/m bounded from above by both the average row degree and the average column degree of the matrix. The cost above becomes $\widetilde{O}(m^\omega \lceil \sigma/m \rceil)$, improving upon the cost of the fastest previously known algorithm for row reduction, which is deterministic. Our algorithm first builds a system of modular equations whose solution set is the row space of the input matrix, and then finds the basis in shifted Popov form of this set. We give a deterministic algorithm for this second step supporting arbitrary moduli in $\widetilde{O}(m^{\omega-1}\sigma)$ field operations, where m is the number of unknowns and σ is the sum of the degrees of the moduli. This extends previous results with the same cost bound in the specific cases of order basis computation and M-Padé approximation, in which the moduli are products of known linear factors.

6.3.13. Fast, deterministic computation of the Hermite normal form and determinant of a polynomial matrix

With G. Labahn and W. Zhou (U. Waterloo, Canada) we give in [64] fast and deterministic algorithms to compute the determinant and Hermite normal form of a nonsingular $n \times n$ matrix of univariate polynomials over a field $\mathbb{K}$. Our algorithms use $\widetilde{O}(n^\omega \lceil s \rceil)$ operations in $\mathbb{K}$, where s is bounded from above by both the average of the degrees of the rows and that of the columns of the matrix and ω is the exponent of matrix multiplication. The soft-O notation indicates that logarithmic factors in the big-O are omitted while the ceiling function indicates that the cost is $\widetilde{O}(n^\omega)$ when $s = o(1)$. Our algorithms are based on a fast and deterministic triangularization method for computing the diagonal entries of the Hermite form of a nonsingular matrix.

6.3.14. Fast Computation of the Rank Profile Matrix and the Generalized Bruhat Decomposition

The row (resp. column) rank profile of a matrix describes the stair-case shape of its row (resp. column) echelon form. With J. G. Dumas and Z. Sultan (LJK, Grenoble), we propose in [11] a new matrix invariant, the rank profile matrix, summarizing all information on the row and column rank profiles of all the leading sub-matrices. We show that this normal form exists and is unique over any ring, provided that the notion of McCoy's rank is used, in the presence of zero divisors. We then explore the conditions for a Gaussian elimination algorithm to compute all or part of this invariant, through the corresponding PLUQ decomposition. This enlarges the set of known Elimination variants that compute row or column rank profiles. As a consequence a new Crout base case variant significantly improves the practical efficiency of previously known implementations over a finite field. With matrices of very small rank, we also generalize the techniques of Storjohann and Yang to the computation of the rank profile matrix, achieving an $(r^\omega + mn)^{1+o(1)}$ time complexity for an $m \times n$ matrix of rank r, where ω is the exponent of matrix multiplication. Finally, by give connections to the Bruhat decomposition, and several of its variants and generalizations. Thus, our algorithmic improvements for the PLUQ factorization, and their implementations, directly apply to these decompositions. In particular, we show how a PLUQ decomposition revealing the rank profile matrix also reveals both a row and

a column echelon form of the input matrix or of any of its leading sub-matrices, by a simple post-processing made of row and column permutations.

6.3.15. Computing with quasiseparable matrices

The class of quasiseparable matrices is defined by a pair of bounds, called the quasiseparable orders, on the ranks of the sub-matrices entirely located in their strictly lower and upper triangular parts. These arise naturally in applications, as e.g. the inverse of band matrices, and are widely used for they admit structured representations allowing to compute with them in time linear in the dimension. In [47] we show the connection between the notion of quasiseparability and the rank profile matrix invariant of Dumas et al. This allows us to propose an algorithm computing the quasiseparable orders (r_L, r_U) in time $O(n^2 s^{\omega-2})$, where $s = \max(r_L, r_U)$ and ω is the exponent of matrix multiplication. We then present two new structured representations, a binary tree of PLUQ decompositions, and the Bruhat generator, using respectively $O(ns\log(n/s))$ and $O(ns)$ field elements instead of $O(ns^2)$ for the classical generator and $O(nslogn)$ for the hierarchically semiseparable representations. We present algorithms computing these representations in time $O(n^2 s^{\omega-2})$. These representations allow a matrix-vector product in time linear in the size of their representation. Lastly we show how to multiply two such structured matrices in time $O(n^2 s^{\omega-2})$.

6.3.16. A Real QZ Algorithm for Structured Companion Pencils

With Y. Eidelman (U. Tel Aviv) and L. Gemignani (U. Pisa), we design in [54] a fast implicit real QZ algorithm for eigenvalue computation of structured companion pencils arising from linearizations of polynomial rootfind-ing problems. The modified QZ algorithm computes the generalized eigenvalues of an $N \times N$ structured matrix pencil using $O(N^2)$ flops and $O(N)$ memory storage. Numerical experiments and comparisons confirm the effectiveness and the stability of the proposed method.

6.3.17. Efficient Solution of Parameter Dependent Quasiseparable Systems and Computation of Meromorphic Matrix Functions

In [55], with Y. Eidelman (U. Tel Aviv) and L. Gemignani (U. Pisa), we focus on the solution of shifted quasiseparable systems and of more general parameter dependent matrix equations with quasiseparable representations. We propose an efficient algorithm exploiting the invariance of the quasiseparable structure under diagonal shifting and inversion. This algorithm is applied to compute various functions of matrices. Numerical experiments show the effectiveness of the approach.

AROMATH Project-Team

6. New Results

6.1. Flat extensions in $_*$-algebras

Participant: Bernard Mourrain.

The main result of the paper [9] is 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.

This is a joint work with K. Schmüdgen.

6.2. On deflation and multiplicity structure

Participant: Bernard Mourrain.

The paper [6] presents two new constructions related to singular solutions of polynomial systems. The first is a new deflation method for an isolated singular root. This construction uses a single linear differential form defined from the Jacobian matrix of the input, and defines the deflated system by applying this differential form to the original system. The advantages of this new deflation is that it does not introduce new variables and the increase in the number of equations is linear in each iteration instead of the quadratic increase of previous methods. The second construction gives the coefficients of the so-called inverse system or dual basis, which defines the multiplicity structure at the singular root. We present a system of equations in the original variables plus a relatively small number of new variables that completely deflates the root in one step. We show that the isolated simple solutions of this new system correspond to roots of the original system with given multiplicity structure up to a given order. Both constructions are "exact " in that they permit one to treat all conjugate roots simultaneously and can be used in certification procedures for singular roots and their multiplicity structure with respect to an exact rational polynomial system.

This is a joint work with J. Hauenstein and A. Szanto.

6.3. On the construction of general cubature formula by flat extensions

Participant: Bernard Mourrain.

We describe a new method to compute general cubature formulae [1]. The problem is initially transformed into the computation of truncated Hankel operators with flat extensions. We then analyse the algebraic properties associated to flat extensions and show how to recover the cubature points and weights from the truncated Hankel operator. We next present an algorithm to test the flat extension property and to additionally compute the decomposition. To generate cubature formulae with a minimal number of points, we propose a new relaxation hierarchy of convex optimization problems minimizing the nuclear norm of the Hankel operators. For a suitably high order of convex relaxation, the minimizer of the optimization problem corresponds to a cubature formula. Furthermore cubature formulae with a minimal number of points are associated to faces of the convex sets. We illustrate our method on some examples, and for each we obtain a new minimal cubature formula.

This is a joint work with Marta Abril-Bucero and C. Bajaj (Univ. of Austin, Texas, USA).

6.4. Geometrically continuous splines for surfaces of arbitrary topology

Participant: Bernard Mourrain.

In the paper [10], we analyze the space of geometrically continuous piecewise polynomial functions or splines for quadrangular and triangular patches with arbitrary topology and general rational transition maps. To define these spaces of G^1 spline functions, we introduce the concept of topological surface with gluing data attached to the edges shared by faces. The framework does not require manifold constructions and is general enough to allow non-orientable surfaces. We describe compatibility conditions on the transition maps so that the space of differentiable functions is ample and show that these conditions are necessary and sufficient to construct ample spline spaces. We determine the dimension of the space of G^1 spline functions which are of degree k on triangular pieces and of bi-degree (k, k) on quadrangular pieces, for k big enough. A separability property on the edges is involved to obtain the dimension formula. An explicit construction of basis functions attached respectively to vertices, edges and faces is proposed and examples of bases of G^1 splines of small degree for topological surfaces with boundary and without boundary are detailed.

This is a joint work with N. Villamizar and R. Vidunas.

6.5. Border Basis for Polynomial System Solving and Optimization

Participant: Bernard Mourrain.

We describe in [15] the software package BORDERBASIX dedicated to the computation of border bases and the solutions of polynomial equations. We present the main ingredients of the border basis algorithm and the other methods implemented in this package: numerical solutions from multiplication matrices, real radical computation, polynomial optimization. The implementation parameterized by the coefficient type and the choice function provides a versatile family of tools for polynomial computation with modular arithmetic, floating point arithmetic or rational arithmetic. It relies on linear algebra solvers for dense and sparse matrices for these various types of coefficients. A connection with SDP solvers has been integrated for the combination of relaxation approaches with border basis computation. Extensive benchmarks on typical polynomial systems are reported, which show the very good performance of the tool.

This is a joint work with M. Abril Bucero and Ph. Trébuchet.

6.6. Bit complexity of bivariate systems

Participant: Ioannis Emiris.

The paper [14] studies the bit complexity of solving systems of bivariate polynomial equations. By means of adapted resultant formulations we thus improve upon the existing general bounds.

6.7. Compact formulae in sparse elimination

Participant: Ioannis Emiris.

This invited talk [12] describes three aspects of constructing compact formulae in toric (or sparse) elimination algebraic theory. We start with the most general existing formula for computing the mixed volume of a square algebraic system, then sketch older and recent progress in matrix formulae for the sparse resultant of an overconstrained system, and conclude with recent work in a matrix formula for the multivariate disccriminant of a specific class of well-constrained systems.

6.8. Computation of the Invariants of Finite Abelian Groups

Participant: Evelyne Hubert.

In [7] we investigate the computation and applications of rational invariants of the linear action of a finite abelian group in the nonmodular case. By diagonalization, such a group action can be described by integer matrices of orders and exponents. We make use of integer linear algebra to compute a minimal generating set of invariants along with the substitution needed to rewrite any invariant in terms of this generating set. In addition, we show how to construct a minimal generating set that consists only of polynomial invariants. As an application, we provide a symmetry reduction scheme for polynomial systems whose solution set is invariant by a finite abelian group action. Finally, we also provide an algorithm to find such symmetries given a polynomial system.

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

6.9. Extraction of cylinders and cones from minimal point sets

Participants: Laurent Busé, André Galligo.

In [3], we propose new algebraic methods for extracting cylinders and cones from minimal point sets, including oriented points. More precisely, we are interested in computing efficiently cylinders through a set of three points, one of them being oriented, or through a set of five simple points. We are also interested in computing efficiently cones through a set of two oriented points, through a set of four points, one of them being oriented, or through a set of six points. For these different interpolation problems, we give optimal bounds on the number of solutions. Moreover, we describe algebraic methods targeted to solve these problems efficiently.

6.10. 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, we prove in [4] 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.

6.11. A Line/Trimmed NURBS Surface Intersection Algorithm Using Matrix Representations

Participant: Laurent Busé.

In the work [11], we contribute a reliable line/surface intersection method for trimmed NURBS surfaces, based on a novel matrix-based implicit representation and numerical methods in linear algebra such as singular value decomposition and the computation of generalized eigenvalues and eigenvectors. A careful treatment of degenerate cases makes our approach robust to intersection points with multiple pre-images. We then apply our intersection algorithm to mesh NURBS surfaces through Delaunay refinement. We demonstrate the added value of our approach in terms of accuracy and treatment of degenerate cases, by providing comparisons with other intersection approaches as well as a variety of meshing experiments.

This is a joint work in collaboration with Pierre Alliez from TITANE Inria project-team and Jingjing SHen and Neil Dodgson both from Cambridge University.

6.12. Effective criteria for bigraded birational maps

Participant: Laurent Busé.

In [2], we consider rational maps whose source is a product of two subvarieties, each one being embedded in a projective space. Our main objective is to investigate birationality criteria for such maps. First, a general criterion is given in terms of the rank of a couple of matrices that became to be known as *Jacobian dual matrices*. Then, we focus on rational maps from $\mathbb{P}^1 \times \mathbb{P}^1$ to $\mathbb{P}^2$ in very low bidegrees and provide new matrix-based birationality criteria by analyzing the syzygies of the defining equations of the map, in particular by looking at the dimension of certain bigraded parts of the syzygy module. Finally, applications of our results to the context of geometric modeling are discussed at the end of the paper.

This is a joint work with N. Botbol (University of Buenos Aires, ARgentina), M. Chardin (UMPC, France), S. H. Hassanzadeh (University of Rio, Brazil), A. Simis (University of Pernambuco, Brazil), Q. H. Tran (UMPC, France). It has been done in the framework of the SYRAM project.

6.13. Geometric model for shape deformation

Participants: Elisa Berrini, Bernard Mourrain.

In [13], we describe a new parametric modeller for an automatic shape optimization loop. The modeller enables the generation of shapes by selecting a set of design parameters that controls a twofold parameterization: geometrical – based on a skeleton approach – and architectural – based on the experience of practitioners, to impact the system performance. The resulting forms are relevant and effective, thanks to a smoothing procedure that ensures the consistency of the shapes produced.

The skeleton consists of a set of B-Spline curves composed of a generating curve and section curves. The deformation of the shape is performed by changing explicit parameters of the representation or implicit parameters such as architectural parameters. The new shape is obtained by minimizing a distance function between the current parameters and the target parameters in combination with a smoothing term to ensure shape consistency. Finally, a 3D surface is reconstructed around the skeleton with an iterative method handling multi-patches and boundary constraints.

Thanks to this approach, architects can directly use a CAD-model based on NURBS representations in the modeller tool that allows a straightforward modification of the initial design to improve performance. The methodology developed can be applied to any shape that can be described by a skeleton, e.g. hulls, foils, bulbous bows, but also wind turbines, airships, etc.

As application, we consider the optimization of the shape of a bulbous bow. The modeller is linked to the RANSE-CFD solver FINE/Marine. The aim is to reduce the total drag of the hull with variation of its bulbous bow shape.

6.14. Shape-optimization of 2D hydrofoils using an Isogeometric BEM solver

Participant: Panagiotis Kaklis.

In [8], an optimization procedure, based on an Isogeometric BEM solver for the potential flow, is developed and used for the shape optimization of hydrofoils. The formulation of the exterior potential-flow problem reduces to a Boundary-Integral Equation (BIE) for the associated velocity potential exploiting the null-pressure jump Kutta condition at the trailing edge. The numerical solution of the BIE is performed by an Isogeometric Boundary-Element Method (BEM) combining a generic B-splines parametric modeler for generating hydrofoil shapes, using a set of eight parameters, the very same basis of the geometric representation for representing the velocity potential and collocation at the Greville abscissas of the knot vector of the hydrofoil's B-splines representation. Furthermore, the optimization environment is developed based on the geometric parametric modeler for the hydrofoil, the Isogeometric BEM solver and an optimizer employing a controlled elitist genetic algorithm. Multi-objective hydrofoil shape optimization examples are demonstrated with respect to the criteria i) maximum lift coefficient and ii) minimum deviation of the hydrofoil area from a reference area.

This is a joint work with K. Kostas (Nazarbayev University), A. Ginnis (National Technical University of Athens), C. Politis (Technological Educational Institute of Athens).

6.15. Algebraic method for constructing singular steady solitary waves: A case study

Participant: André Galligo.

The article [5] describes the use of algebraic methods in a phase plane analysis of ordinary differential equations. The method is illustrated by the study of capillary-gravity steady surface waves propagating in shallow water. We consider the (fully nonlinear, weakly dispersive) Serre-Green-Naghdi equations with surface tension, because it provides a tractable model that, in the same time, is not too simple so the interest of the method can be emphasised. In particular, we analyse a special class of solutions, the solitary waves, which play an important role in many fields of Physics. In capillary-gravity regime, there are two kinds of localised infinitely smooth travelling wave solutions – solitary waves of elevation and of depression. However, if we

allow the solitary waves to have an angular point, the "zoology" of solutions becomes much richer and the main goal of this study is to provide a complete classification of such singular localised solutions using the methods of the effective Algebraic Geometry.

This is a joint work with D. Clamond (Laboratoire Jean Alexandre Dieudonné, Université de Nice Sophia-Antipolis) and Denys Dutykh (Laboratoire de Mathématiques, Université de Savoie).

CARAMBA Project-Team

7. New Results

7.1. Collecting Relation for the Number Field Sieve in Medium Characteristic

Participants: Pierrick Gaudry, Laurent Grémy [contact], Marion Videau.

We study the relation collection of NFS in medium characteristic, especially in $\mathrm{GF}(p^6)$ [4]. We compare different polynomial selections that affect drastically the relation collection step, by giving the explicit formula in 3 dimensions of two functions to select the best polynomials. For the relation collection, we design new sieve algorithms in 3 dimensions and do the practical comparison of the different polynomial selections for different p. Finally, we perform the relation collection step for a field of 389 bits in 800 days, the largest computed relation collection in this type of field.

7.2. Recent Progress on the Elliptic Curve Discrete Logarithm Problem

Participant: Pierrick Gaudry [contact].

A survey on the elliptic curve discrete logarithm problem has been written in collaboration with S. Galbraith (Auckland). It appeared in a special issue of DCC [3], for the 25th birthday of the journal.

7.3. A Modified Block Lanczos Algorithm with Fewer Vectors

Participant: Emmanuel Thomé [contact].

In the context of a book project entitled "Topics in Computational Number Theory inspired by Peter L. Montgomery" (edited by Joppe W. Bos and Arjen K. Lenstra), E. Thomé contributed a chapter on "the Block Lanczos algorithm" (owed to Peter L. Montgomery [35]). This was the occasion to rework and streamline the presentation of the block Lanczos algorithm. In fact, several new characteristics of the algorithm were obtained in this process: a version adapted to homogeneous systems, an improvement on the memory footprint of the algorithm, and a heuristic justification for the success probability of the algorithm. While the collated book is still not published yet (publication is expected in 2017), the chapter is published in preprint form as [14].

7.4. Factorization of RSA-220 with CADO-NFS

Participants: Pierrick Gaudry, Emmanuel Thomé, Paul Zimmermann [contact].

In May 2016 we have completed with CADO-NFS the factorization of RSA-220 [15], which was started in December 2013. The sieving was completed in September 2014, and the first phase of the linear algebra (`krylov`) in October 2014. However we had to improve CADO-NFS to be able to run the `lingen` sub-step of the linear algebra. This was completed in January 2016, and the end of the factorization ran smoothly. This factorization is the largest one done with CADO-NFS, and the third largest one overall, after RSA-768 (232 digits) factored in December 2009, and $3^{697}+1$ (221 digits) factored by NFS@Home in February 2015.

7.5. Linear Time Interactive Certificates for the Minimal Polynomial and the Determinant of a Sparse Matrix

Participant: Emmanuel Thomé [contact].

Following discussion with Jean-Guillaume Dumas which began in March 2015 on the topic of computing checkpoints for the `krylov` step of the block Wiedemann algorithm, we determined that a scheme very similar to this checkpointing technique (originally designed to spot data corruption errors) was able to provide a proving algorithm —in the cryptographic sense— for the computation of the minimal polynomial of a sparse matrix, or for its determinant. This led to a joint paper with Jean-Guillaume Dumas, Erich Kaltofen and Gilles Villard, published at ISSAC 2016 [8].

7.6. A Kilobit Hidden SNFS Discrete Logarithm Computation

Participants: Pierrick Gaudry, Emmanuel Thomé [contact].

In collaboration with Josh Fried and Nadia Heninger from University of Pennsylvania, we worked on discrete logarithm computation modulo primes of a special form, amenable to computation with the Special Number Field Sieve (SNFS). Our original interest in this question came from the observation that primes which are conspicuous SNFS targets *are* found in the wild, as we observed in the context of the LogJam attack in 2015. We first ran a test computation on such a prime in March ($p = 2^{784}-2^{28}+1027679$, found in the LibTomcrypt library. For modern cryptographic uses, such a prime qualifies undoubtedly as "not good'). Based on the computational data obtained, and on further work, we expanded to larger sizes. We crafted a prime which was chosen as a "best case" for SNFS, yet with the property that this SNFS-optimality cannot be detected. We call such primes "trapdoored primes". We showed that computing discrete logarithms modulo trapdoored primes is entirely feasible for 1024-bit primes. In the article [18], we also showed that there are primes which are found in the wild (e.g., in RFC 5114) which could plausibly be trapdoored primes, given that no justification of their origin is provided. In fact, while cryptographic best practice is to provide "rigid" choices whenever random choices are to be set publicly, the sad truth is that random data lacking a justification is found quite often.

In the context of [18], we also put into practice an improvement of the implementation of the block Wiedemann algorithm in Cado-NFS, that allowed to reduce the time for the linear algebra computation significantly.

7.7. Solving Discrete Logarithms on a 170-bit MNT Curve by Pairing Reduction

Participants: Aurore Guillevic [contact], Emmanuel Thomé [contact].

The project of computing discrete logarithms in finite fields of the form $\mathrm{GF}(p^n)$ for small n comes from the need to estimate precisely the security level of pairing-based cryptography. After the two record computations of 2014 and 2015 in $\mathrm{GF}(p^2)$ of 160 and 180 decimal digits (532 and 597 bits) we investigated $\mathrm{GF}(p^3)$ and took a real-life elliptic curve proposed in 2001 by Miyaji, Nakabayashi and Takano (MNT-3 curve). Thanks to a pairing computation (in few milliseconds), a discrete logarithm computation in the 170-bit MNT-3 curve, which is hard, can be done instead by a discrete logarithm computation in $\mathrm{GF}(p^3)$ of 508 bits, which is much faster. This computation involved Aurore Guillevic (post-doctoral fellow in 2016 at the University of Calgary, Canada), Emmanuel Thomé, and François Morain (LIX/École Polytechnique/Inria Saclay, GRACE team). The computation took 2.97 years in total: 1.81 years for the relation collection, 1.16 years for the linear algebra and 2 days for the individual discrete logarithm computation. The work was presented at the Selected Areas in Cryptography conference in Newfoundland, Canada, and published in the proceedings [11].

The next step will be to adapt the new NFS variant called Extended-Tower-NFS to attack MNT-4 and MNT-6 curves, which means computing discrete logarithms in $\mathrm{GF}(p^4)$ and $\mathrm{GF}(p^6)$. This new challenge will require the higher dimension sieve developed by Laurent Grémy.

7.8. Computing Jacobi's Theta in Quasi-linear Time

Participant: Hugo Labrande [contact].

Most of the results have been obtained in 2015. The article was accepted for publication in 2016 [5].

We study the multiprecision computation of the theta function in genus 1, *i.e.,* the Jacobi theta function. The main result is that $\theta(z,\tau)$ can be computed in time that is quasi-linear in the precision P, using an algorithm which follows the same strategy as the case of theta-constants (Dupont, 2006). A thorough analysis of the precision loss is given in order to prove correctness.

Along with this work, we have publicly released an open source implementation of the algorithm in C (using the GNU MPC library). This implementation shows this algorithm is faster than a more naive approach for precisions greater than 300,000 digits.

7.9. Computing Theta Functions in Quasi-linear Time in Genus 2 and Above

Participants: Hugo Labrande, Emmanuel Thomé [contact].

We study the multiprecision computation of the theta function in genus 2. We extend the quasi-linear algorithm for Jacobi's theta to genus 2, generalizing the approach we undertook in previous work; this required finding workarounds, most notably for the choice of signs and for being able to apply Newton's method. We also give an outline of an algorithm for the theta function in genus g, but the workarounds we found in genus 2 would need to be generalized to this case before claiming any sort of result in genus g [6].

We released along with this work a Magma implementation of our fast genus 2 algorithm, along with an implementation of a somewhat naive (but previously state-of-the-art) algorithm for genus 2. Our results show that our algorithm is faster than the naive one for precisions greater than 3,000 digits.

7.10. Computing Small Certificates of Inconsistency of Quadratic Fewnomial Systems

Participant: Pierre-Jean Spaenlehauer [contact].

This is a joint work with Jean-Charles Faugère (Inria, EPI Polsys) and Jules Svartz (Inria EPI Polsys/Ministère Éducation Nationale). Most of the results have been obtained in 2015. This work was finalized and published in 2016 [10].

We study how Gröbner bases algorithms can be adapted to compute certificates that *quadratic fewnomial systems* (*i.e.*, systems in which only a small subset of monomials occur in the equations) do not have any solution. The main results are algorithms and complexity bounds which take into account the sparsity of the monomial support of the system, under some mild genericity assumptions on the coefficients of the systems.

7.11. Critical Point Computations on Smooth Varieties: Degree and Complexity Bounds

Participant: Pierre-Jean Spaenlehauer [contact].

This is a joint work with Mohab Safey El Din (Univ. Paris 6, EPI Polsys). This work led to a publication in the proceedings of the ISSAC conference [13].

Let $V \subset \mathbb{C}^n$ be an equidimensional algebraic set and g be an n-variate polynomial with rational coefficients. Computing the critical points of the map that evaluates g at the points of V is a cornerstone of several algorithms in real algebraic geometry and optimization. Under the assumption that the critical locus is finite and that the projective closure of V is smooth, we provide sharp upper bounds on the degree of the critical locus which depend only on $\deg(g)$ and the degrees of the generic polar varieties associated to V. Using these degree bounds and an algorithm due to Bank, Giusti, Heintz, Lecerf, Matera and Solernó, we derive complexity bounds which are quadratic in the degree bounds (up to logarithmic factors) and polynomial in all the other parameters of the problem.

7.12. Constructing Sparse Polynomial Systems with Many Positive Solutions

Participant: Pierre-Jean Spaenlehauer [contact].

This is a joint work with Frédéric Bihan (Univ. de Savoie, LAMA). Most of the results have been obtained in 2015 [25]; we improved the results during 2016.

Consider a regular triangulation of the convex-hull P of a set $\mathcal{A}$ of n points in $\mathbb{R}^d$, and a real matrix C of size $d \times n$. A version of Viro's method allows to construct from these data an unmixed polynomial system with support $\mathcal{A}$ and coefficient matrix C whose number of positive solutions is bounded from below by the number of d-simplices which are positively decorated by C (a d-simplex is positively decorated by C if the $d \times (d+1)$ sub-matrix of C corresponding to the simplex has a kernel vector all coefficients of which are positive). We show that all the d-simplices of a triangulation can be positively decorated if and only if the triangulation is balanced, which in turn is equivalent to the fact that its dual graph is bipartite. This allows us to identify, among classical families, monomial supports which admit maximally positive systems, giving some evidence in favor of a conjecture due to Bihan. We also use this technique in order to construct fewnomial systems with many positive solutions.

7.13. Modular Arithmetic and ECM on the Kalray MPPA-256 Processor

Participants: Jérémie Detrey [contact], Pierrick Gaudry.

In collaboration with Masahiro Ishii from the Nara Institute of Science and Technology, Nara (Japan) we have developed a fast modular arithmetic library for the Kalray MPPA-256, which is a many-core processor with a VLIW architecture. Carefully written assembly allowed us to obtain a close to optimal use of the computing units of all the cores for the multiprecision multiplication of integers. As an application, the ECM factoring algorithm was implemented on top of our library. The performances are very interesting compared to other architectures like GPU, especially in terms of power consumption [19].

7.14. Determinism and Computational Power of Real Measurement-based Quantum Computation

Participant: Luc Sanselme [contact].

This is a joint work with Simon Perdrix (CNRS, Carte Team at Loria). This work has begun in 2014.

The starting point for this work was about a problem in «Quantum cloud computing». A person with a classical resource wants to perform a quantum computation. To do so he asks some quantum resources to perform his computation. The difficult part is that he wants to be sure that the quantum resources he asks to perform his computation don't cheat and return him the good results. This kind of «Quantum cloud computing» is called interactive proofs. The quantum resources are called the provers. Real Measurement-based quantum computing (MBQC) has been used for interactive proofs by McKague.

Measurement-based quantum computing (MBQC) is a universal model for quantum computation. The combinatorial characterization of determinism in this model, powered by measurements, and hence, fundamentally probabilistic, is the cornerstone of most of the breakthrough results in this field. To answer our question, we needed to develop some tools in this MBQC field. The most general known sufficient condition for a deterministic MBQC to be driven is that the underlying graph of the computation has a particular kind of flow called Pauli flow. The necessity of the Pauli flow was an open question. We showed that the Pauli flow is necessary for real-MBQC, and not in general providing counter-examples for (complex) MBQC. We explored the consequences of this result for real MBQC and its applications. Real MBQC and more generally real quantum computing is known to be universal for quantum computing. In the interactive proofs developed by McKague, the two-prover case corresponds to real-MBQC on bipartite graphs. While (complex) MBQC on bipartite graphs are universal, the universality of real MBQC on bipartite graphs was an open question. We showed that real bipartite MBQC is not universal: we proved that all measurements of real bipartite MBQC can be parallelized. Therefore, real bipartite MBQC leads to constant depth computations. As a consequence, McKague techniques cannot lead to two-prover interactive proofs.

7.15. Fast Integer Multiplication Using Generalized Fermat Primes

Participants: Svyatoslav Covanov [contact], Emmanuel Thomé.

The paper [17] describes an algorithm for the multiplication of two n-bit integers. It achieves the best asymptotic complexity bound $O(n \log n \cdot 4^{\log^* n})$ under a hypothesis on the distribution of generalized Fermat primes of the form $r^{2^\lambda} + 1$. This hypothesis states that there always exists a sufficiently small interval in which we can find such a prime. Experimental results give evidence in favor of this assumption. This article has been submitted to Mathematics of Computation and some corrections, that have been requested, are processed currently.

7.16. Search for Primitive Trinomials

Participant: Paul Zimmermann [contact].

This is a joint work with Richard Brent (University of Newcastle, Australia).

We have performed a search for primitive trinomials $x^r + x^s + 1$ over $\mathrm{GF}(2)$ of degree $r = 42\,643\,801$, $r = 43\,112\,609$, $r = 57\,885\,161$ and $r = 74\,207\,281$, which are the new Mersenne prime exponents found by the GIMPS project. We found respectively 5, 4, 0 and 3 primitive trinomials [16], for example the three primitive trinomials of degree $74\,207\,281$ are (with their reverse trinomials):

$$x^{74207281} + x^{9156813} + 1, \qquad x^{74207281} + x^{9999621} + 1, \qquad x^{74207281} + x^{30684570} + 1.$$

CASCADE Project-Team

6. New Results

6.1. Results

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

- More efficient constructions with lattices
- New e-cash constructions
- Advanced primitives for the privacy in the cloud
- Efficient functional encryption
- Various predicate encryption schemes

DATASHAPE Team

7. New Results

7.1. Algorithmic aspects of topological and geometric data analysis

7.1.1. An Efficient Representation for Filtrations of Simplicial Complexes

Participant: Jean-Daniel Boissonnat.

In collaboration with Karthik C.S. (Department of Computer Science and Applied Mathematics, Weizmann Institute of Science, Israel)

A filtration over a simplicial complex K is an ordering of the simplices of K such that all prefixes in the ordering are subcomplexes of K. Filtrations are at the core of Persistent Homology, a major tool in Topological Data Analysis. In order to represent the filtration of a simplicial complex, the entire filtration can be appended to any data structure that explicitly stores all the simplices of the complex such as the Hasse diagram or the recently introduced Simplex Tree by Boissonnat and Maria [Algorithmica '14]. However, with the popularity of various computational methods that need to handle simplicial complexes, and with the rapidly increasing size of the complexes, the task of finding a compact data structure that can still support efficient queries is of great interest.

This direction has been recently pursued for the case of maintaining simplicial complexes. For instance, Boissonnat et al. [SoCG '15] considered storing the simplices that are maximal for the inclusion and Attali et al. [IJCGA '12] considered storing the simplices that block the expansion of the complex. Nevertheless, so far there has been no data structure that compactly stores the *filtration* of a simplicial complex, while also allowing the efficient implementation of basic operations on the complex.

In this work [22], we propose a new data structure called the Critical Simplex Diagram (CSD) which is a variant of our work on the Simplex Array List (SAL) introduced in [SoCG '15]. Our data structure allows to store in a compact way the filtration of a simplicial complex, and allows for the efficient implementation of a large range of basic operations. Moreover, we prove that our data structure is essentially optimal with respect to the requisite storage space. Next, we show that the CSD representation admits the following construction algorithms.

- A new *edge-deletion* algorithm for the fast construction of Flag complexes, which only depends on the number of critical simplices and the number of vertices.
- A new *matrix-parsing* algorithm to quickly construct the relaxed strong Delaunay complexes, depending only on the number of witnesses and the dimension of the complex.

7.1.2. Discretized Riemannian Delaunay triangulations

Participants: Mael Rouxel-Labbé, Mathijs Wintraecken, Jean-Daniel Boissonnat.

Anisotropic meshes are desirable for various applications, such as the numerical solving of partial differential equations and graphics. In [27], we introduce an algorithm to compute discrete approximations of Riemannian Voronoi diagrams on 2-manifolds. This is not straightforward because geodesics, shortest paths between points, and therefore distances cannot in general be computed exactly. Our implementation employs recent developments in the numerical computation of geodesic distances and is accelerated through the use of an underlying anisotropic graph structure. We give conditions that guarantee that our discrete Riemannian Voronoi diagram is combinatorially equivalent to the Riemannian Voronoi diagram and that its dual is an embedded triangulation, using both approximate geodesics and straight edges. Both the theoretical guarantees on the approximation of the Voronoi diagram and the implementation are new and provide a step towards the practical application of Riemannian Delaunay triangulations.

7.1.3. Efficient and Robust Persistent Homology for Measures

Participants: Frédéric Chazal, Steve Oudot.

In collaboration with M. Buchet (Tohoku University), D. Sheehy (Univ. Connecticut).

A new paradigm for point cloud data analysis has emerged recently, where point clouds are no longer treated as mere compact sets but rather as empirical measures. A notion of distance to such measures has been defined and shown to be stable with respect to perturbations of the measure. This distance can easily be computed pointwise in the case of a point cloud, but its sublevel-sets, which carry the geometric information about the measure, remain hard to compute or approximate. This makes it challenging to adapt many powerful techniques based on the Euclidean distance to a point cloud to the more general setting of the distance to a measure on a metric space. We propose an efficient and reliable scheme to approximate the topological structure of the family of sublevel-sets of the distance to a measure. We obtain an algorithm for approximating the persistent homology of the distance to an empirical measure that works in arbitrary metric spaces. Precise quality and complexity guarantees are given with a discussion on the behavior of our approach in practice [17].

7.1.4. Shallow Packings in Geometry

Participants: Kunal Dutta, Arijit Ghosh.

A merged paper with Ezra, Esther (School of Mathematics, Georgia Institute of Technology, Atlanta, U.S.A.)

We refine the bound on the packing number, originally shown by Haussler, for shallow geometric set systems. Specifically, let V be a finite set system defined over an n-point set X; we view V as a set of indicator vectors over the n-dimensional unit cube. A $delta$-separated set of V is a subcollection W, such that the Hamming distance between each pair $u, v \in W$ is greater than δ, where $\delta > 0$ is an integer parameter. The δ-packing number is then defined as the cardinality of the largest δ-separated subcollection of V. Haussler showed an asymptotically tight bound of $\Theta((n/delta)^d)$ on the δ-packing number if V has VC-dimension (or primal shatter dimension) d. We refine this bound for the scenario where, for any subset, $X' \subset X$ of size $m \leq n$ and for any parameter $1 \leq k \leq m$, the number of vectors of length at most k in the restriction of V to X' is only $O(m^{d_1} k^{d-d_1})$, for a fixed integer $d > 0$ and a real parameter $1 \leq d_1 \leq d$ (this generalizes the standard notion of bounded primal shatter dimension when $d_1 = d$). In this case when V is "k-shallow" (all vector lengths are at most k), we show that its δ-packing number is $O(n^{d_1} k^{d-d_1}/\delta^d)$, matching Haussler's bound for the special cases where $d_1 = d$ or $k = n$. We present two proofs, the first is an extension of Haussler's approach, and the second extends the proof of Chazelle, originally presented as a simplification for Haussler's proof. [21]

- A new *tight upper bound* for shallow-packings in δ-separated set systems of bounded primal shatter dimension.

7.1.5. On Subgraphs of Bounded Degeneracy in Hypergraphs

Participants: Kunal Dutta, Arijit Ghosh.

A k-uniform hypergraph has degeneracy bounded by d if every induced subgraph has a vertex of degree at most d. Given a k-uniform hypergraph $H = (V(H), E(H))$, we show there exists an induced subgraph of size at least

$$\sum_{v \in V(H)} \min 1, ck \left(\frac{d+1}{d_H(v)+1} \right)^{1/(k-1)},$$

where $c_k = 2^{-(1+\frac{1}{k-1})}\left(1-\frac{1}{k}\right)$ and $d_H(v)$ denotes the degree of vertex v in the hypergraph H. This extends and generalizes a result of Alon-Kahn-Seymour (Graphs and Combinatorics, 1987) for graphs, as well as a result of Dutta-Mubayi-Subramanian (SIAM Journal on Discrete Mathematics, 2012) for linear hypergraphs, to general k-uniform hypergraphs. We also generalize the results of Srinivasan and Shachnai (SIAM Journal on Discrete Mathematics, 2004) from independent sets (0-degenerate subgraphs) to d-degenerate subgraphs. We further give a simple non-probabilistic proof of the Dutta-Mubayi-Subramanian bound for linear k-uniform hypergraphs, which extends the Alon-Kahn-Seymour proof technique to hypergraphs. Our proof combines the random permutation technique of Bopanna-Caro-Wei (see e.g. The Probabilistic Method, N. Alon and J. H. Spencer; Dutta-Mubayi-Subramanian) and also Beame-Luby (SODA, 1990) together with a new local density argument which may be of independent interest. We also provide some applications in discrete geometry, and address some natural algorithmic questions. [28]

- A new algorithmic *lower bound* for largest d-degenerate subgraphs in k-uniform hypergraphs.

7.1.6. A Simple Proof of Optimal Epsilon Nets

Participants: Kunal Dutta, Arijit Ghosh.

In collaboration with Nabil Mustafa (Université Paris-Est, Laboratoire d'Informatique Gaspard-Monge, ESIEE Paris, France.)

Showing the existence of ε-nets of small size has been the subject of investigation for almost 30 years, starting from the initial breakthrough of Haussler and Welzl (1987). Following a long line of successive improvements, recent results have settled the question of the size of the smallest ε-nets for set systems as a function of their so-called shallow-cell complexity.

In this paper we give a short proof of this theorem in the space of a few elementary paragraphs, showing that it follows by combining the ε-net bound of Haussler and Welzl (1987) with a variant of Haussler's packing lemma (1991).

This implies all known cases of results on unweighted ε-nets studied for the past 30 years, starting from the result of Matoušek, Seidel and Welzl (1990) to that of Clarkson and Varadajan (2007) to that of Varadarajan (2010) and Chan, Grant, Könemann and Sharpe (2012) for the unweighted case, as well as the technical and intricate paper of Aronov, Ezra and Sharir (2010). [40]

- A new *unified proof* for all known bounds on unweighted ε-nets studied in the last 30 years.

7.1.7. Combinatorics of Set Systems with Small Shallow Cell Complexity: Optimal Bounds via Packings

Participants: Kunal Dutta, Arijit Ghosh.

In collaboration with Bruno Jartoux and Nabil Mustafa (Université Paris-Est Marne-la-Vallée, Laboratoire d'Informatique Gaspard-Monge, ESIEE Paris, France.)

The packing lemma of Haussler states that given a set system (X, R) with bounded VC dimension, if every pair of sets in R are 'far apart' (i.e., have large symmetric difference), then R cannot contain too many sets. This has turned out to be the technical foundation for many results in geometric discrepancy using the entropy method as well as recent work on set systems with bounded VC dimension. Recently it was generalized to the shallow packing lemma [Dutta-Ezra-Ghosh SoCG 2015, Mustafa DCG 2016], applying to set systems as a function of their shallow cell complexity. In this paper we present several new results and applications related to packings:

1. an optimal lower bound for shallow packings, thus settling the open question in Ezra (SODA 2014) and Dutta et al. (SoCG 2015),
2. improved bounds on Mnets, providing a combinatorial analogue to Macbeath regions in convex geometry (Annals of Mathematics, 1952),
3. simplifying and generalizing the main technical tool in Fox et al. (J. of the EMS, 2016).

Besides using the packing lemma and a combinatorial construction, our proofs combine tools from polynomial partitioning and the probabilistic method. [37]

- A new *optimal lower bound* for shallow packings.
- New *improved bounds* for M-nets - combinatorial analogs of Macbeath regions in convex geometry.

7.1.8. A new asymmetric correlation inequality for Gaussian measure

Participants: Kunal Dutta, Arijit Ghosh.

In collaboration with Nabil Mustafa (Université Paris-Est Marne-la-Vallée, Laboratoire d'Informatique Gaspard-Monge, ESIEE Paris, France.)

The Khatri-Šidák lemma says that for any Gaussian measure μ over $\mathbb{R}^n$, given a convex set K and a slab L, both symmetric about the origin, one has $\mu(K \cap L) \geq \mu(K)\mu(L)$. We state and prove a new asymmetric version of the Khatri-Šidák lemma when K is a symmetric convex body and L is a slab (not necessarily symmetric about the barycenter of K). Our result also extends that of Szarek and Werner (1999), in a special case.

- A new *asymmetric* inequality for gaussian measure. [38].

7.2. Statistical aspects of topological and geometric data analysis

7.2.1. Stability and Minimax Optimality of Tangential Delaunay Complexes for Manifold Reconstruction

Participant: Eddie Aamari.

In collaboration with C. Levrard (Univ. Paris Diderot).

we consider the problem of optimality in manifold reconstruction. A random sample $\mathbb{X}_n = \{X_1, \ldots, X_n\} \subset \mathbb{R}^D$ composed of points lying on a d-dimensional submanifold M, with or without outliers drawn in the ambient space, is observed. Based on the tangential Delaunay complex, we construct an estimator$\widehat{M}$ that is ambient isotopic and Hausdorff-close to M with high probability. $\widehat{M}$ is built from existing algorithms. In a model without outliers, we show that this estimator is asymptotically minimax optimal for the Hausdorff distance over a class of submanifolds with reach condition. Therefore, even with no a priori information on the tangent spaces of M, our estimator based on tangential Delaunay complexes is optimal. This shows that the optimal rate of convergence can be achieved through existing algorithms. A similar result is also derived in a model with outliers. A geometric interpolation result is derived, showing that the tangential Delaunay complex is stable with respect to noise and perturbations of the tangent spaces. In the process, a denoising procedure and a tangent space estimator both based on local principal component analysis (PCA) are studied [32].

7.2.2. Rates in the Central Limit Theorem and diffusion approximation via Stein's Method

Participant: Thomas Bonis.

We present a way to apply Stein's method in order to bound the Wasserstein distance between a, possibly discrete, measure and another measure assumed to be the invariant measure of a diffusion operator. We apply this construction to obtain convergence rates, in terms of p-Wasserstein distance for $p \geq 2$, in the Central Limit Theorem in dimension 1 under precise moment conditions. We also establish a similar result for the Wasserstein distance of order 2 in the multidimensional setting. In a second time, we study the convergence of stationary distributions of Markov chains in the context of diffusion approximation, with applications to density estimation from geometric random graphs and to sampling using the Langevin Monte Carlo algorithm [33].

7.2.3. Rates of Convergence for Robust Geometric Inference

Participants: Frédéric Chazal, Bertrand Michel.

In collaboration with P. Massart (Univ. Paris Sud et Inria Select team).

Distances to compact sets are widely used in the field of Topological Data Analysis for inferring geometric and topological features from point clouds. In this context, the distance to a probability measure (DTM) has been introduced by Chazal et al. as a robust alternative to the distance a compact set. In practice, the DTM can be estimated by its empirical counterpart, that is the distance to the empirical measure (DTEM). In this paper we give a tight control of the deviation of the DTEM. Our analysis relies on a local analysis of empirical processes. In particular, we show that the rate of convergence of the DTEM directly depends on the regularity at zero of a particular quantile function which contains some local information about the geometry of the support. This quantile function is the relevant quantity to describe precisely how difficult is a geometric inference problem. Several numerical experiments illustrate the convergence of the DTEM and also confirm that our bounds are tight [19].

7.2.4. Data driven estimation of Laplace-Beltrami operator

Participants: Frédéric Chazal, Bertrand Michel, Ilaria Giulini.

Approximations of Laplace-Beltrami operators on manifolds through graph Laplacians have become popular tools in data analysis and machine learning. These discretized operators usually depend on bandwidth parameters whose tuning remains a theoretical and practical problem. In this paper, we address this problem for the unnormalized graph Laplacian by establishing an oracle inequality that opens the door to a well-founded data-driven procedure for the bandwidth selection. Our approach relies on recent results by Lacour and Massart on the so-called Lepski's method [26].

7.3. Topological approach for multimodal data processing

7.3.1. Persistence-based Pooling for Shape Pose Recognition

Participants: Thomas Bonis, Frédéric Chazal, Steve Oudot, Maksim Ovsjanikov.

We propose a novel pooling approach for shape classification and recognition using the bag-of-words pipeline, based on topological persistence, a recent tool from Topological Data Analysis. Our technique extends the standard max-pooling, which summarizes the distribution of a visual feature with a single number, thereby losing any notion of spatiality. Instead, we propose to use topological persistence, and the derived persistence diagrams, to provide significantly more informative and spatially sensitive characterizations of the feature functions, which can lead to better recognition performance. Unfortunately, despite their conceptual appeal, persistence diagrams are difficult to handle, since they are not naturally represented as vectors in Euclidean space and even the standard metric, the bottleneck distance is not easy to compute. Furthermore, classical distances between diagrams, such as the bottleneck and Wasserstein distances, do not allow to build positive definite kernels that can be used for learning. To handle this issue, we provide a novel way to transform persistence diagrams into vectors, in which comparisons are trivial. Finally, we demonstrate the performance of our construction on the Non-Rigid 3D Human Models SHREC 2014 dataset, where we show that topological pooling can provide significant improvements over the standard pooling methods for the shape pose recognition within the bag-of-words pipeline [23].

7.3.2. Structure and Stability of the 1-Dimensional Mapper

Participants: Steve Oudot, Mathieu Carrière.

Given a continuous function $f : X \to \mathbb{R}$ and a cover $\mathcal{I}$ of its image by intervals, the Mapper is the nerve of a refinement of the pullback cover $f^{-1}(\mathcal{I})$. Despite its success in applications, little is known about the structure and stability of this construction from a theoretical point of view. As a pixelized version of the Reeb graph of f, it is expected to capture a subset of its features (branches, holes), depending on how the interval cover is positioned with respect to the critical values of the function. Its stability should also depend on this positioning. We propose a theoretical framework that relates the structure of the Mapper to the one of the Reeb graph, making it possible to predict which features will be present and which will be absent in the Mapper given the function and the cover, and for each feature, to quantify its degree of (in-)stability. Using this framework, we can derive guarantees on the structure of the Mapper, on its stability, and on its convergence to the Reeb graph as the granularity of the cover $\mathcal{I}$ goes to zero [25].

7.3.3. Decomposition of exact pfd persistence bimodules

Participants: Steve Oudot, Jérémy Cochoy.

We characterize the class of persistence modules indexed over $\mathbb{R}^2$ that are decomposable into summands whose support have the shape of a *block*—i.e. a horizontal band, a vertical band, an upper-right quadrant, or a lower-left quadrant. Assuming the modules are *pointwise finite-dimensional* (pfd), we show that they are decomposable into block summands if and only if they satisfy a certain local property called *exactness*. Our proof follows the same scheme as the proof of decomposition for pfd persistence modules indexed over $\mathbb{R}$, yet it departs from it at key stages due to the product order not being a total order on $\mathbb{R}^2$, which leaves some important gaps open. These gaps are filled in using more direct arguments. Our work is motivated primarily by the stability theory for zigzags and interlevel-sets persistence modules, in which block-decomposable bimodules play a key part. Our results allow us to drop some of the conditions under which that theory holds, in particular the Morse-type conditions [39].

7.4. Experimental research and software development

7.4.1. Topological Microstructure Analysis Using Persistence Landscapes

Participant: Paweł Dłotko.

In collaboration with T. Wanner (George Mason University).

Phase separation mechanisms can produce a variety of complicated and intricate microstructures, which often can be difficult to characterize in a quantitative way. In recent years, a number of novel topological metrics for microstructures have been proposed, which measure essential connectivity information and are based on techniques from algebraic topology. Such metrics are inherently computable using computational homology, provided the microstructures are discretized using a thresholding process. However, while in many cases the thresholding is straightforward, noise and measurement errors can lead to misleading metric values. In such situations, persistence landscapes have been proposed as a natural topology metric. Common to all of these approaches is the enormous data reduction, which passes from complicated patterns to discrete information. It is therefore natural to wonder what type of information is actually retained by the topology. In the present paper, we demonstrate that averaged persistence landscapes can be used to recover central system information in the Cahn-Hilliard theory of phase separation. More precisely, we show that topological information of evolving microstructures alone suffices to accurately detect both concentration information and the actual decomposition stage of a data snapshot. Considering that persistent homology only measures discrete connectivity information, regardless of the size of the topological features, these results indicate that the system parameters in a phase separation process affect the topology considerably more than anticipated. We believe that the methods discussed in this paper could provide a valuable tool for relating experimental data to model simulations [36].

7.4.2. Topological analysis of the connectome of digital reconstructions of neural microcircuits

Participant: Paweł Dłotko.

In collaboration with K. Hess, L. Ran, H. Markram, E. Muller, M. Nolte, M. Reimann, M. Scolamiero, K. Turner (Univ. of Aberden, EPFL, Brain and Mind Institute).

A first draft digital reconstruction and simulation of a microcircuit of neurons in the neocortex of a two-week-old rat was recently published. Since graph-theoretical methods may not be sufficient to understand the immense complexity of the network formed by the neurons and their connections, we explored whether application of methods from algebraic topology can provide a novel and useful perspective on the structural and functional organization of the microcircuit. Structural topological analysis revealed that directed graphs representing the connectivity between neurons are significantly different from random graphs and that there exist an enormous number of simplicial complexes of different dimensions representing all-to-all connections within different sets of neurons, the most extreme motif of neuronal clustering reported so far in the brain. Functional topological analysis based on data from simulations confirmed the interest of a new approach to

studying the relationship between the structure of the connectome and its emergent functions. In particular, functional responses to different stimuli can readily be distinguished by topological methods. This study represents the first algebraic topological analysis of connectomics data from neural microcircuits and shows promise for general applications in network science.

7.4.3. A persistence landscapes toolbox for topological statistics

Participant: Paweł Dłotko.

In collaboration with P. Bubenik (University of Florida).

Topological data analysis provides a multiscale description of the geometry and topology of quantitative data. The persistence landscape is a topological summary that can be easily combined with tools from statistics and machine learning. We give efficient algorithms for calculating persistence landscapes, their averages, and distances between such averages. We discuss an implementation of these algorithms and some related procedures. These are intended to facilitate the combination of statistics and machine learning with topological data analysis. We present an experiment showing that the low-dimensional persistence landscapes of points sampled from spheres (and boxes) of varying dimensions differ.

7.5. Miscellaneous

7.5.1. Monotone Simultaneous Paths Embeddings in $\mathbb{R}^d$

Participant: Marc Glisse.

In collaboration with O. Devillers and S. Lazard (Inria Nancy), David Bremner (University of New Brunswick, Canada), Giuseppe Liotta (University of Perugia, Italy), Tamara Mchedlidze (KIT, Germany), Sue Whitesides (University of Victoria, Canada), Stephen Wismath (University of Lethbridge, Canada).

We study[24] the following problem: Given k paths that share the same vertex set, is there a simultaneous geometric embedding of these paths such that each individual drawing is monotone in some direction? We prove that for any dimension $d \geq 2$, there is a set of $d + 1$ paths that does not admit a monotone simultaneous geometric embedding.

GRACE Project-Team

7. New Results

7.1. Faster elliptic and hyperelliptic curve cryptography

B. Smith made several contributions to the development of faster arithmetic on elliptic curves and genus 2 Jacobians in 2016. In joint work with C. Costello and P.-N. Chung, he gave a new, efficient, uniform, and constant-time scalar multiplication algorithm for genus 2 Jacobians exploiting fast Kummer surface arithmetic and features of differential addition chains; this was presented at SAC 2016. The theory in this article was the basis of a highly competitive implementation of key exchange and signatures for microcontroller platforms, in joint work with J. Renes, P. Schwabe, and L. Batina, presented at CHES 2016.

7.2. Quantum factoring

Integer factorization via Shor's algorithm is a benchmark problem for general quantum computers, but surprisingly little work has been done on optimizing the algorithm for use as a serious factoring tool once large quantum computers are built (rather than as a proof of concept). In the meantime, given the limited size of contemporary quantum computers and the practical difficulties involved in building them, any optimizations to quantum factoring algorithms can lead to significant practical improvements. In a new interdisciplinary project with physicists F. Grosshans and T. Lawson, F. Morain and B. Smith have derived a simple new quantum factoring algorithm for cryptographic integers; its expected runtime is lower than Shor's factoring algorithm, and it should also be easier to implement in practice [22].

7.3. Advances in point counting

Determining the number of points on an elliptic curve, or more generally on the Jacobian of an algebraic curve, is a classic problem in algorithmic number theory that is now crucial for efficiently generating secure cryptographic parameters. Together with C. Scribot, F. Morain and B. Smith developed an improved version of the state-of-the-art SEA algorithm for certain families of elliptic curves with special endomorphisms; this was presented at ANTS-XII [10]. B. Smith also led a project group on special genus-2 point counting algorithms at the "Algebraic Geometry for Coding Theory and Cryptography" workshop at IPAM, UCLA, in 2016.

7.4. Cryptanalysis of code based cryptosystems by filtration attacks

The McEliece encryption scheme based on binary Goppa codes was one of the first public-key encryption schemes [31]. 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 [32], 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 [26], 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 [3]. A detailed long version has been written and recently published [9]. A. Couvreur, Irene Márquez–Corbella, and R. Pellikaan broke McEliece based on algebraic geometry codes from curves of arbitrary genus [2], [27] by reconstructing optimal polynomial time decoding algorithms decoding up to the half minimum distance minus half the genus. This can be computed from the raw data of a generator matrix. In a recently submitted long version [21] the algorithm has been improved and permits to reconstruct a decoding algorithm up to the half minimum distance.

7.5. Quantum LDPC codes

Quantum codes are the analogous of error correcting codes for a quantum computer. A well known family of quantum codes are the CSS codes due to Calderbank, Shor and Steane can be represented by a pair of matrices (H_X, H_Z) such that $H_X H_Z^T = 0$. As in classical coding theory, if these matrices are sparse, then the code is said to be LDPC. An open problem in quantum coding theory is to get a family of quantum LDPC codes whose asymptotic minimum distance is in $\Omega(n^\alpha)$ for some $\alpha > 1/2$. No such family is known and actually, only few known families of quantum LDPC codes have a minimum distance tending to infinity.

In [24], Benjamin Audoux (I2M, Marseille) and A. Couvreur investigate a problem suggested by Bravyi and Hastings. They studied the behaviour of iterated tensor powers of CSS codes and prove in particular that such families always have a minimum distance tending to infinity. They propose also 3 families of LDPC codes whose minimum distance is in $\Omega(n^\beta)$ for all $\beta < 1/2$.

7.6. Discrete Logarithm computations in finite fields with the NFS algorithm

The best discrete logarithm record computations in prime fields and large characteristic finite fields are obtained with Number Field Sieve algorithm (NFS) at the moment. This algorithm is made of four steps:

1. polynomial selection;
2. relation collection (with a sieving technique);
3. linear algebra (computing the kernel of a huge matrix, of millions of rows and columns);
4. individual discrete logarithm computation.

The two more time consuming steps are the relation collection step and the linear algebra step. The polynomial selection is quite fast but is very important since it determines the complexity of the algorithm. Selecting better polynomials is a key to improve the overall running-time of the NFS algorithm.

A. Guillevic and F. Morain have written a chapter [18] on discrete logarithm computations for a book on pairings.

7.6.1. Breaking a MNT curve using DL computations

There is a reduction between an elliptic curve E defined over $\mathbf{F}_p$ and a finite extension of degree k (aka *embedding degree*) of the base field, using pairing computations. In brief, one can transport the discrete logarithm problem from E to $\mathbf{F}_{p^k}$. If k is relatively small, this yields a DLP much easier to solve than directly on E. To give some highlight on current easyness, A. Guillevic, F. Morain and E. Thomé (from CARAMBA EPC in LORIA) computed a discrete log on a curve of embedding degree 3 and cryptographic size. This clearly showed that curves with small embedding degrees are indeed weak. The article [14] was presented by A. Guillevic during the SAC 2016 conference in New Foundland.

7.7. 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.

7.8. Hash function cryptanalysis

Cryptographic hash functions are versatile primitives that are used in many cryptographic protocols. The security of a hash function h is usually evaluated through two main notions: its preimage resistance (given a target t, the difficulty of finding a message m s.t. $h(m) = t$) and its collision resistance (the difficulty of finding two messages m, m' s.t. $h(m) = h(m')$).

A popular hash function is the SHA-1 algorithm. Although theoretical collision attacks were found in 2005, it is still being used in some applications, for instance as the hash function in some TLS certificates. Hence cryptanalysis of SHA-1 is still a major topic in cryptography.

In 2015, we improved the state-of-the-art on SHA-1 analysis in two ways:

- T. Espitau, P.-A. Fouque and P. Karpman improved the previous preimage attacks on SHA-1, reaching up to 62 rounds (out of 80), up from 57. The corresponding paper was published at CRYPTO 2015.
- P. Karpman, T. Peyrin and M. Stevens developed collision attacks on the compression function of SHA-1 (i.e. freestart collisions). This exploits a model that is slightly more generous to the attacker in order to find explicit collisions on more rounds than what was previously possible. A first work resulted in freestart collisions for SHA-1 reduced to 76 steps; this attack takes less than a week to compute on a common GPU. The corresponding paper was published at CRYPTO 2015. This was later improved to attack the full compression function. Although the attack is more expensive it is still practical, taking less than two weeks on a 64 GPU cluster. The corresponding paper was accepted at EUROCRYPT 2016 [17].

7.9. Block cipher design and analysis

Block ciphers are one of the most basic cryptographic primitives, yet block cipher analysis is still a major research topic. In recent years, the community also shifted focus to the more general setting of *authenticated encryption*, where one specifies an (set of) algorithm(s) providing both encryption and authentication for messages of arbitrary length. A major current event in that direction is the CAESAR academic competition, which aims to select a portfolio of good algorithms.

In 2015, we helped to improve the state of the art in block cipher research in several ways:

- P. Karpman developed a compact 8-bit S-box with branch number three, which can be used as a basis to construct a lightweight block cipher particularly efficient on 8-bit microcontrollers [23].

In 2016, together with P.-A. Fouque, P. Kirchner and B. Minaud, P. Karpman designed a family of efficient provably incompressible symmetric primitives, which corresponds to a weak notion of white-box cryptography. The objective of such algorithms is that given an implementation of a certain target size, an adversary shouldn't be able to efficiently find a smaller implementation with comparable functionality. We introduced a security model that captures the behaviour of realistic adversaries and used this model to prove the security of a family of block cipher and a family of key generating functions. The corresponding paper was published at ASIACRYPT 2016 [13].

7.10. Weight distribution of Algebraic-Geometry codes

V. Ducet worked on the weight distribution of geometric codes following a method initiated by Duursma. More precisely he implemented his method in magma and was able to compute the weight distribution of the geometric codes coming from two optimal curves of genus 2 and 3 over the finite fields of size 16 and 9 respectively. The aim is to compute the weight distribution of the Hermitian code over the finite field of size 16, for which computational improvements of the implementation are necessary.

7.11. Update on the Chor-Rivest cryptosystem

The Chor-Rivest cryptosystem from the 90's was "broken" by Vaudenay. However, Vaudenay's attack applies only for the range of parameters originally proposed. The major recent breakthrough in discrete logarithm computations enable to redesign the system with a completly different range of paramaters, possibly thwarting Vaudenay's attack. D. Augot and C. Barbin tried to find a new attack against this discrete log and knapsack-based cryptosystem, using the Sidelnikov-Shestakov algorithm for recovering a Reed-Solomon code. Apparently, our new attack does not outperform S. Vaudenay's original attack, and it may be possible that the Chor-Rivest could be redesigned in a secure way.

7.12. Proofs or Retrievability

A Proof of Retrievability (PoR) is a cryptographic protocol which aims at ensuring a user that he can retrieve files he previously stored on a server. J. Lavauzelle and F. Levy-dit-Vehel studied a new approach for the construction of PoRs. The idea is to encode the file so that the user can check with low communication whether its file has been damaged. Such an encoding can be efficiently done with locally decodable and testable codes, and especially with the family of lifted codes introduced by Guo, Kopparty and Sudan [30]. In practice, PoRs thus defined achieve very efficient storage overhead and acceptable communication, compared to the existing litterature. This new construction [15] has been presented during the ISIT2016 conference in Barcelona.

7.13. Fast Encoding of Multiplicity Codes

N. Coxon has produced a fast implementation which demonstrates that the multiplicity codes from Kopparty, Saraf and Yehkanin are indeed practical for very large databases (when used in the Private Information Retrieval setting). For instance, we can encode a 10^8 bit longs message in two seconds on a regular laptop, and 10^9 in thirty seconds. We envisioned a scenario where DNA sequences are encoded using these multiplicity codes: 10^8 bits is the size of Drosophila melanogaster (flies), and 10^9 bits is the order of magnitude of the human genome.

7.14. Private Information Retrieval

Imagine the following scenario, in which a researcher wants to access many sustrings a DNA sequences, while maintaining the privacy of the request. The privacy or the secrecy of the database is not a concern here: for instance, this researcher wants to access many DNA subsequences of drosophila melanogaster, hosted on a remote data broker, and clearly the concern is not to protect the private life of flies. But the information leaked about the queries may endanger the novel aspect of the discovery the researcher is about to make, by revealing which DNA sequences he is studying.

Private Information Retrieval (PIR) schemes are designed to achieve this goal: a user queries a database T hosted on a remote server, and wants the i-th entry, i.e. $T[i]$. A cryptographic protocol is then run, and at the end of the protocol, the server must not know i, neither the $T[i]$ he answered, yet the user gets $T[i]$.

These PIR schemes can be achieved in an unconditionally secure way using the above Multiplicity codes, which N. Coxon made practical. In September, we explained this scenario and demoed our software at Nokia Bell Lab's Future X days a use case of Multiplicity codes for private access to DNA sequences.

7.15. Compact McEliece Keys from Algebraic-geometry codes

In 1978, McEliece [31], introduced a public key cryptosystem based on linear codes and suggested to use classical Goppa codes which belong to the family of alternant codes. This proposition remains secure but leads to very large public keys compared to other public-key cryptosystems. Many proposals have been made in order to reduce the key size, in particular quasi-cyclic alternant codes. Quasi-cyclic alternant codes refer to alternant codes admitting a generator matrix made of severals cyclic bloks. These alternant codes contains weakness because they have a non-trivial automorphism group. Thanks to this property we can build, from a quasi-cyclic alternant code, an alternant code with smaller parameters which has almost same private elements than the original code. Faugère, Otmani, Tillich, Perret and Portzamparc [29] showed this fact for alternant codes obtained by using supports $x \in \mathbb{F}_{q^m}^n$ globally stable by an affine map $\phi : z \mapsto az + b$, with $a, b \in \mathbb{F}_{q^m}^n$. E. Barelli has extended this proof to the non-affine case: for all codes obtained by using supports $x \in \mathbb{F}_{q^m}^n$ globally stable by a map $\phi : z \mapsto \frac{az+b}{cz+d}$, with $a, b, c, d \in \mathbb{F}_{q^m}^n$.

In order to suggest compact keys for the McEliece cryptosystem E. Barelli and A. Couvreur studied quasi-cyclic alternant gemeotric codes. Alternant geometric codes means a subfield subcode of an algebraic-geometry codes. To build these codes, we need curves with automorphisms. In particular, we studied Kummer cover of plane curves.

LFANT Project-Team

6. New Results

6.1. Class invariants in genus 2

Abelian surfaces, or equivalently, Jacobian varieties of genus 2 hyperelliptic curves, offer the same security as elliptic curves in a cryptographic setting and often better efficiency, and could thus be an attractive alternative. The theory of complex multiplication can be used to obtain cryptographically secure curves. Relying on Shimura reciprocity for Siegel modular forms, we have developed the necessary mathematical theory in [24]. It requires deeper algebraic reasoning than for elliptic curves: Ideals of the endomorphism rings of the abelian varieties are no more two-dimensional modules over the integers, but two-dimensional projective modules over quadratic number rings. We succeed in proving results adapted from the elliptic curve case by suitably normalising quadratic forms over number rings and using strong approximation. The result is an elegant theory that leads to clearly formulated and practical algorithms, which we illustrate by examples.

6.2. Elliptic curve and Abelian varieties cryptology

Participant: Damien Robert.

The paper [15] in which David Lubicz and Damien Robert explain how to improve the arithmetic of Abelian and Kummer varieties has been published in the journal Finite Fields and Their Applications. The speed of the arithmetic is a crucial factor in the performance of cryptosystems based on abelian varieties. Depending on the cryptographic application, the speed record holders 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 problem in certain cryptographic protocols. The previous known models to work on the Jacobian rather than the Kummer surface (Mumford coordinates or the 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 addition), and uses it to speed up multi-exponentiations in Kummer varieties and to obtain new models of abelian surfaces in which the scalar multiplication is as fast as on the Kummer surface.

Theta functions, and in particular the Dedekind eta function, are at the heart of complex multiplication constructions of curves. They can be written as sparse power series with coefficients ± 1. In [23] we devise optimised addition sequences for the occurring exponents, with a proof relying on classical number theory, which help us gain a factor of 2 compared to the standard approach and which is validated in practice by our two independent implementations. Using an approach from computer algebra and a proof relying on analytic number theory, we obtain another factor of 2.

6.3. Symbolic computation

The article [27], of which F. Johansson is a coauthor, was published. The article describes SymPy, an open source computer algebra system written in pure Python. It is built with a focus on extensibility and ease of use, through both interactive and programmatic applications. These characteristics have led SymPy to become a popular symbolic library for the scientific Python ecosystem. This paper presents the architecture of SymPy, a description of its features, and a discussion of select submodules. The supplementary material provide additional examples and further outline details of the architecture and features of SymPy.

Hypergeometric functions are among the most important mathematical functions, with a wide range of applications in everything from physics to number theory. The practical computation of such functions is a challenging problem. The preprint [26]. presents an efficient implementation of hypergeometric functions in arbitrary-precision interval arithmetic. The functions ${}_0F_1$, ${}_1F_1$, ${}_2F_1$ and ${}_2F_0$ (or the Kummer U-function) are supported for unrestricted complex parameters and argument, and by extension, we cover exponential and trigonometric integrals, error functions, Fresnel integrals, incomplete gamma and beta functions, Bessel functions, Airy functions, Legendre functions, Jacobi polynomials, complete elliptic integrals, and other special functions. The output can be used directly for interval computations or to generate provably correct floating-point approximations in any format. Performance is competitive with earlier arbitrary-precision software, and sometimes orders of magnitude faster. We also partially cover the generalized hypergeometric function ${}_pF_q$ and computation of high-order parameter derivatives.

The preprint [25] is the corresponding paper for the software Arb developed by F. Johansson. Arb is a C library for arbitrary-precision interval arithmetic using the midpoint-radius representation, also known as ball arithmetic. It supports real and complex numbers, polynomials, power series, matrices, and evaluation of many special functions. The core number types are designed for versatility and speed in a range of scenarios, allowing performance that is competitive with non-interval arbitrary-precision types such as MPFR and MPC floating-point numbers. This paper discusses the low-level number representation, strategies for precision and error bounds, and the implementation of efficient polynomial arithmetic with interval coefficients.

6.4. Logarithmic Class Groups

Logarithmic class groups and units, introduced by Jaulent in 1994, are an intriguing ℓ-adic variation on the classical class and unit groups related to Iwasawa theory and the wild kernels of algebraic K-theory. These $\mathbb{Z}_\ell$-modules of finite type provide direct access to invariants studied in standard conjectures about $\mathbb{Z}_\ell$-extensions. In [12] we devised a new algorithm to explicitly compute them in subexponential time under standard conjectures (GRH and Gross-Kuz'min) and to validate unconditionaly the computed results (now in exponential time). The algorithm has been implementated in the PARI/GP system.

6.5. Class groups and other invariants of number fields

The article by H. Cohen and F. Thorne on Dirichlet series associated to quartic fields with given cubic 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 quartic fields whose resolvent field is isomorphic to a fixed cubic field k.

The article [22] by H. Cohen and F. Thorne generalizes the work of A. Morra and the authors, on giving explicit formulas for the Dirichlet series generating function of D_ℓ extensions of odd prime degree ℓ with given quadratic resolvent. Over the course of the proof, the authors explain connections between their formulas and the Ankeny-Artin-Chowla conjecture, the Ohno-Nakagawa relation for binary cubic forms, and other topics.

In her thesis, Iuliana Ciocanea-Teodorescu describes algorithms that answer questions arising in ring and module theory. The first main result of this thesis concerns the module isomorphism problem, how to compute a set of generators of minimal cardinality, and how to construct projective covers and injective hulls. The thesis also describe tests for module simplicity, projectivity, and injectivity, and constructive tests for existence of surjective module homomorphisms between two finite modules, one of which is projective. As a negative result, the problem of testing for existence of injective module homomorphisms between two finite modules, one of which is projective, is NP-complete. The last part of the thesis is concerned with finding a good working approximation of the Jacobson radical of a finite ring, that is, a two-sided nilpotent ideal such that the corresponding quotient ring is almost semisimple. The notion used to approximate semisimplicity is that of separability.

In her thesis [11], Pinar Kiliçer determines all CM curves of genus 2 defined over the reflex field. This extends the previous CM class number one problem for elliptic curves which asked to find all elliptic curves defined over the rationals with non-trivial endomorphism ring.

6.6. Number and function fields

The article [13] written by J. Brau and J. Nathan on “Elliptic curves with 2-torsion contained in the 3-torsion field” has been published. This article 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 $\mathbb{Q}(E[3])$. The authors characterize the j-invariants of elliptic curves with this property by exhibiting an explicit model of $X'(6)$. $X'(6)(\mathbb{Q})$ then 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 $\mathbb{Q}$.

POLSYS Project-Team

6. New Results

6.1. Fundamental algorithms and structured polynomial systems

6.1.1. Linear Algebra for Computing Gröbner Bases of Linear Recursive Multidimensional Sequences

The so-called Berlekamp – Massey – Sakata algorithm computes a Gröbner basis of a 0-dimensional ideal of relations satisfied by an input table. It extends the Berlekamp – Massey algorithm to n-dimensional tables, for $n > 1$.

In the extended version [6], we investigate this problem and design several algorithms for computing such a Gröbner basis of an ideal of relations using linear algebra techniques. The first one performs a lot of table queries and is analogous to a change of variables on the ideal of relations.

As each query to the table can be expensive, we design a second algorithm requiring fewer queries, in general. This FGLM-like algorithm allows us to compute the relations of the table by extracting a full rank submatrix of a *multi-Hankel* matrix (a multivariate generalization of Hankel matrices).

Under some additional assumptions, we make a third, adaptive, algorithm and reduce further the number of table queries. Then, we relate the number of queries of this third algorithm to the *geometry* of the final staircase and we show that it is essentially linear in the size of the output when the staircase is convex. As a direct application to this, we decode n-cyclic codes, a generalization in dimension n of Reed Solomon codes.

We show that the multi-Hankel matrices are heavily structured when using the LEX ordering and that we can speed up the computations using fast algorithms for quasi-Hankel matrices. Finally, we design algorithms for computing the generating series of a linear recursive table.

6.1.2. Guessing Linear Recurrence Relations of Sequence Tuples and P-recursive Sequences with Linear Algebra

Given several n-dimensional sequences, we first present in [23] an algorithm for computing the Gröbner basis of their module of linear recurrence relations.

A P-recursive sequence $(u_{\mathbf{i}})_{\mathbf{i}\in\mathbb{N}^n}$ satisfies linear recurrence relations with polynomial coefficients in $\mathbf{i}$, as defined by Stanley in 1980. Calling directly the aforementioned algorithm on the tuple of sequences $\left((\mathbf{i}^{\mathbf{j}}\, u_{\mathbf{i}})_{\mathbf{i}\in\mathbb{N}^n}\right)_{\mathbf{j}}$ for retrieving the relations yields redundant relations. Since the module of relations of a P-recursive sequence also has an extra structure of a 0-dimensional right ideal of an Ore algebra, we design a more efficient algorithm that takes advantage of this extra structure for computing the relations.

Finally, we show how to incorporate Gröbner bases computations in an Ore algebra $\mathbb{K}\langle t_1, ..., t_n, x_1, ..., x_n\rangle$, with commutators $x_k\, x_\ell - x_\ell\, x_k = t_k\, t_\ell - t_\ell\, t_k = t_k\, x_\ell - x_\ell\, t_k = 0$ for $k \neq \ell$ and $t_k\, x_k - x_k\, t_k = x_k$, into the algorithm designed for P-recursive sequences. This allows us to compute faster the Gröbner basis of the ideal spanned by the first relations, such as in 2D/3D-space walks examples.

6.1.3. On the Connection Between Ritt Characteristic Sets and Buchberger-Gröbner Bases

For any polynomial ideal I, let the minimal triangular set contained in the reduced Buchberger–Gröbner basis of I with respect to the purely lexicographical term order be called the W-characteristic set of I. In [18], we establish a strong connection between Ritt's characteristic sets and Buchberger's Gröbner bases of polynomial ideals by showing that the W-characteristic set C of I is a Ritt characteristic set of I whenever C is an ascending set, and a Ritt characteristic set of I can always be computed from C with simple pseudo-division when C is regular. We also prove that under certain variable ordering, either the W-characteristic set of I is normal, or irregularity occurs for the jth, but not the $(j+1)$th, elimination ideal of I for some j. In the

latter case, we provide explicit pseudo-divisibility relations, which lead to nontrivial factorizations of certain polynomials in the Buchberger–Gröbner basis and thus reveal the structure of such polynomials. The pseudo-divisibility relations may be used to devise an algorithm to decompose arbitrary polynomial sets into normal triangular sets based on Buchberger–Gröbner bases computation.

6.1.4. On the complexity of computing 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 [12], 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 $\left(\prod w_i\right)^{\omega}$. 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 $\left(\prod w_i\right)^{\omega}$. 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.

6.1.5. A Superfast Randomized Algorithm to Decompose Binary Forms

Symmetric Tensor Decomposition is a major problem that arises in areas such as signal processing, statistics, data analysis and computational neuroscience. It is equivalent to a homogeneous polynomial in n variables of degree D as a sum of Dth powers of linear forms, using the minimal number of summands. This minimal number is called the rank of the polynomial/tensor. We consider the decomposition of binary forms, that corresponds to the decomposition of symmetric tensors of dimension 2 and order D. This problem has its roots in Invariant Theory, where the decompositions are known as canonical forms. As part of that theory, different algorithms were proposed for the binary forms. In recent years, those algorithms were extended for the general symmetric tensor decomposition problem. We present in [22] a new randomized algorithm that enhances the previous approaches with results from structured linear algebra and techniques from linear recurrent sequences. It achieves a softly linear arithmetic complexity bound. To the best of our knowledge, the previously known algorithms have quadratic complexity bounds.

6.1.6. On the Bit Complexity of Solving Bilinear Polynomial Systems

In [29] we bound the Boolean complexity of computing isolating hyperboxes for all complex roots of systems of bilinear polynomials. The resultant of such systems admits a family of determinantal Sylvester-type formulas, which we make explicit by means of homological complexes. The computation of the determinant of the resultant matrix is a bottleneck for the overall complexity. We exploit the quasi-Toeplitz structure to reduce the problem to efficient matrix-vector products, corresponding to multivariate polynomial multiplication. For zero-dimensional systems, we arrive at a primitive element and a rational univariate representation of the roots. The overall bit complexity of our probabilistic algorithm is $\widetilde{O}_B(n^4D^4 + n^2D^4\tau)$, where n is the number of variables, D equals the bilinear Bézout bound, and τ is the maximum coefficient bitsize. In addition, a careful infinitesimal symbolic perturbation of the system allows us to treat degenerate and positive dimensional systems, thus making our algorithms and complexity analysis applicable to the general case.

6.2. Solving Systems over the Reals and Applications

6.2.1. Exact algorithms for linear matrix inequalities

Let $A(x) = A_0 + x_1A_1 + ... + x_nA_n$ be a linear matrix, or pencil, generated by given symmetric matrices $A_0, A_1, ..., A_n$ of size m with rational entries. The set of real vectors x such that the pencil is positive semidefinite is a convex semi-algebraic set called spectrahedron, described by a linear matrix inequality (LMI). In [13], we design an exact algorithm that, up to genericity assumptions on the input matrices, computes an exact algebraic representation of at least one point in the spectrahedron, or decides that it is empty. The algorithm does not assume the existence of an interior point, and the computed point minimizes the rank of the pencil on the spectrahedron. The degree d of the algebraic representation of the point coincides experimentally with the algebraic degree of a generic semidefinite program associated to the pencil. We provide explicit bounds for the complexity of our algorithm, proving that the maximum number of arithmetic operations that are performed is essentially quadratic in a multilinear Bézout bound of d. When m (resp. n) is fixed, such a bound, and hence the complexity, is polynomial in n (resp. m). We conclude by providing results of experiments showing practical improvements with respect to state-of-the-art computer algebra algorithms.

6.2.2. Real root finding for determinants of linear matrices

Let $A_0, A_1, ..., A_n$ be given square matrices of size m with rational coefficients. In [14], we focus on the exact computation of one point in each connected component of the real determinantal variety $\{x \in \mathbb{R}^n : \det(A_0 + x_1A_1 + \cdots + x_nA_n) = 0\}$. Such a problem finds applications in many areas such as control theory, computational geometry, optimization, etc. Using standard complexity results this problem can be solved using $m^{O(n)}$ arithmetic operations. Under some genericity assumptions on the coefficients of the matrices, we provide an algorithm solving this problem whose runtime is essentially quadratic in $\binom{n+m}{n}^3$. We also report on experiments with a computer implementation of this algorithm. Its practical performance illustrates the complexity estimates. In particular, we emphasize that for subfamilies of this problem where m is fixed, the complexity is polynomial in n.

6.2.3. A nearly optimal algorithm for deciding connectivity queries in smooth and bounded real algebraic sets

A roadmap for a semi-algebraic set S is a curve which has a non-empty and connected intersection with all connected components of S. Hence, this kind of object, introduced by Canny, can be used to answer connectivity queries (with applications, for instance, to motion planning) but has also become of central importance in effective real algebraic geometry, since it is used in higher-level algorithms. In [15], we provide a probabilistic algorithm which computes roadmaps for smooth and bounded real algebraic sets. Its output size and running time are polynomial in $(nD)^{n\log(d)}$, where D is the maximum of the degrees of the input polynomials, d is the dimension of the set under consideration and n is the number of variables. More precisely, the running time of the algorithm is essentially subquadratic in the output size. Even under our assumptions, it is the first roadmap algorithm with output size and running time polynomial in $(nD)^{n\log(d)}$.

6.2.4. Determinantal sets, singularities and application to optimal control in medical imagery

Control theory has recently been involved in the field of nuclear magnetic resonance imagery. The goal is to control the magnetic field optimally in order to improve the contrast between two biological matters on the pictures. Geometric optimal control leads us here to analyze mero-morphic vector fields depending upon physical parameters, and having their singularities defined by a determinantal variety. The involved matrix has polynomial entries with respect to both the state variables and the parameters. Taking into account the physical constraints of the problem, one needs to classify, with respect to the parameters, the number of real singularities lying in some prescribed semi-algebraic set. In [24], we develop a dedicated algorithm for real root classification of the singularities of the rank defects of a polynomial matrix, cut with a given semi-algebraic set. The algorithm works under some genericity assumptions which are easy to check. These assumptions are not so restrictive and are satisfied in the aforementioned application. As more general strategies for real root classification do, our algorithm needs to compute the critical loci of some maps,

intersections with the boundary of the semi-algebraic domain, etc. In order to compute these objects, the determinantal structure is exploited through a stratification by the rank of the polynomial matrix. This speeds up the computations by a factor 100. Furthermore, our implementation is able to solve the application in medical imagery, which was out of reach of more general algorithms for real root classification. For instance, computational results show that the contrast problem where one of the matters is water is partitioned into three distinct classes.

6.2.5. Optimal Control of an Ensemble of Bloch Equations with Applications in MRI

The optimal control of an ensemble of Bloch equations describing the evolution of an ensemble of spins is the mathematical model used in Nuclear Resonance Imaging and the associated costs lead to consider Mayer optimal control problems. The Maximum Principle allows to parameterize the optimal control and the dynamics is analyzed in the framework of geometric optimal control. This leads to numerical implementations or suboptimal controls using averaging principle as presented in [25].

6.2.6. Critical Point Computations on Smooth Varieties: Degree and Complexity bounds

Let $V \subset \mathbb{C}^n$ be an equidimensional algebraic set and g be an n-variate polynomial with rational coefficients. Computing the critical points of the map that evaluates g at the points of V is a cornerstone of several algorithms in real algebraic geometry and optimization. Under the assumption that the critical locus is finite and that the projective closure of V is smooth, we provide in [31] sharp upper bounds on the degree of the critical locus which depend only on $\deg(g)$ and the degrees of the generic polar varieties associated to V. Hence, in some special cases where the degrees of the generic polar varieties do not reach the worst-case bounds, this implies that the number of critical points of the evaluation map of g is less than the currently known degree bounds. We show that, given a lifting fiber of V, a slight variant of an algorithm due to Bank, Giusti, Heintz, Lecerf, Matera and Solernó computes these critical points in time which is quadratic in this bound up to logarithmic factors, linear in the complexity of evaluating the input system and polynomial in the number of variables and the maximum degree of the input polynomials.

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

6.3.1. Structural Cryptanalysis of McEliece Schemes with Compact Key.

A very popular trend in code-based cryptography is to decrease the public-key size by focusing on subclasses of alternant/Goppa codes which admit a very compact public matrix, typically quasi-cyclic (QC), quasi-dyadic (QD), or quasi-monoidic (QM) matrices. We show in [11] that the very same reason which allows to construct a compact public-key makes the key-recovery problem intrinsically much easier. The gain on the public-key size induces an important security drop, which is as large as the compression factor p on the public-key. The fundamental remark is that from the $k \times n$ public generator matrix of a compact McEliece, one can construct a $k/p \times n/p$ generator matrix which is – from an attacker point of view – as good as the initial public-key. We call this new smaller code the *folded code*. Any key-recovery attack can be deployed equivalently on this smaller generator matrix. To mount the key-recovery in practice, we also improve the algebraic technique of Faugère, Otmani, Perret and Tillich (FOPT). In particular, we introduce new algebraic equations allowing to include codes defined over any prime field in the scope of our attack. We describe a so-called "structural elimination" which is a new algebraic manipulation which simplifies the key-recovery system. As a proof of concept, we report successful attacks on many cryptographic parameters available in the literature. All the parameters of CFS-signatures based on QD/QM codes that have been proposed can be broken by this approach. In most cases, our attack takes few seconds (the hardest case requires less than 2 hours). In the encryption case, the algebraic systems are harder to solve in practice. Still, our attack succeeds against several cryptographic challenges proposed for QD and QM encryption schemes. We mention that some parameters that have been proposed in the literature remain out of reach of the methods given here. weakness arising from Goppa codes with QM or QD symmetries. Indeed, the security of such schemes is not relying on the bigger compact public matrix but on the small folded code which can be efficiently broken in practice with an algebraic attack for a large set of parameters

6.3.2. 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 [10], 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 no symmetry anymore. This result is obtained thanks to an 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. Lastly, 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.

6.3.3. Factoring $N = p^r q^s$ for Large r and s

D. Boneh, G. Durfee, and N. Howgrave-Graham showed at Crypto 99 that moduli of the form $N = p^r q$ can be factored in polynomial time when $r \simeq \log p$. Their algorithm is based on Coppersmith's technique for finding small roots of polynomial equations. In [27], we show that $N = p^r q^s$ can also be factored in polynomial time when r or s is at least $(\log p)^3$; therefore we identify a new class of integers that can be efficiently factored. We also generalize our algorithm to moduli equal to a product of k factors of prime powers $p_i^{r_i}$; we show that a non-trivial factor of N can be extracted in polynomial-time if one of the exponents r_i is large enough.

6.3.4. On the p-adic stability of the FGLM algorithm

Nowadays, many strategies to solve polynomial systems use the computation of a Gröbner basis for the graded reverse lexicographical ordering, followed by a change of ordering algorithm to obtain a Gröbner basis for the lexicographical ordering. The change of ordering algorithm is crucial for these strategies. In [33], we study the p-adic stability of the main change of ordering algorithm, FGLM. We show that FGLM is stable and give explicit upper bound on the loss of precision occuring in its execution. The variant of FGLM designed to pass from the grevlex ordering to a Gröbner basis in shape position is also stable. Our study relies on the application of Smith Normal Form computations for linear algebra.

6.3.5. Binary Permutation Polynomial Inversion and Application to Obfuscation Techniques

Whether it is for constant obfusation, opaque predicate or equation obfuscation, Mixed Boolean-Arithmetic (MBA) expressions are a powerful tool providing concrete ways to achieve obfuscation. Recent results introduced ways to mix such a tool with permutation polynomials modulo 2^n in order to make the obfuscation technique more resilient to SMT solvers. However, because of limitations regarding the inversion of such permutations, the set of permutation polynomials presented suffers some restrictions. Those restrictions allow several methods of arithmetic simplification, decreasing the effectiveness of the technique at hiding information. In [19], we present general methods for permutation polynomials inversion. These methods allow us to remove some of the restrictions presented in the literature, making simplification attacks less effective. We discuss complexity and limits of these methods, and conclude that not only current simplification attacks may not be as effective as we thought, but they are still many uses of polynomial permutations in obfuscation that are yet to be explored.

6.3.6. Horizontal Side-Channel Attacks and Countermeasures on the ISW Masking Scheme

A common countermeasure against side-channel attacks consists in using the masking scheme originally introduced by Ishai, Sahai and Wagner (ISW) at Crypto 2003, and further generalized by Rivain and Prouff at CHES 2010. The countermeasure is provably secure in the probing model, and it was showed by Duc, Dziembowski and Faust at Eurocrypt 2014 that the proof can be extended to the more realistic noisy leakage model. However the extension only applies if the leakage noise increases at least linearly with the masking order n, which is not necessarily possible in practice. In [20], we investigate the security of an implementation when the previous condition is not satisfied, for example when the masking order n increases for a constant noise. We exhibit two (template) horizontal side-channel attacks against the Rivain-Prouff's secure multiplication scheme and we analyze their efficiency thanks to several simulations and experiments. Eventually, we describe a variant of Rivain-Prouff's multiplication that is still provably secure in the original ISW model, and also heuristically secure against our new attacks.

6.3.7. Faster Evaluation of SBoxes via Common Shares

In [28], we describe a new technique for improving the efficiency of the masking countermeasure against side-channel attacks. Our technique is based on using common shares between secret variables, in order to reduce the number of finite field multiplications. Our algorithms are proven secure in the ISW probing model with $n > t + 1$ shares against t probes. For AES, we get an equivalent of 2.8 non-linear multiplications for every SBox evaluation, instead of 4 in the Rivain-Prouff countermeasure. We obtain similar improvements for other block-ciphers. Our technique is easy to implement and performs relatively well in practice, with roughly a 20% speed-up compared to existing algorithms.

6.3.8. Information Extraction in the Presence of Masking with Kernel Discriminant Analysis

To reduce the memory and timing complexity of the Side-Channel Attacks (SCA), dimensionality reduction techniques are usually applied to the measurements. They aim to detect the so-called Points of Interest (PoIs), which are time samples which (jointly) depend on some sensitive information (e.g. secret key sub-parts), and exploit them to extract information. The extraction is done through the use of functions which combine the measurement time samples. Examples of combining functions are the linear combinations provided by the Principal Component Analysis or the Linear Discriminant Analysis. When a masking countermeasure is properly implemented to thwart SCAs, the selection of PoIs is known to be a hard task: almost all existing methods have a combinatorial complexity explosion, since they require an exhaustive search among all possible d-tuples of points. In this paper we propose an efficient method for informative feature extraction in presence of masking countermeasure. This method, called Kernel Discriminant Analysis, consists in completing the Linear Discriminant Analysis with a so-called kernel trick, in order to efficiently perform it over the set of all possible d-tuples of points without growing in complexity with d. We identify and analyse the issues related to the application of such a method. Afterwards, its performances are compared to those of the Projection Pursuit (PP) tool for PoI selection up to a 4th-order context. Experiments show that the Kernel Discriminant Analysis remains effective and efficient for high-order attacks, leading to a valuable alternative to the PP in constrained contexts where the increase of the order d does not imply a growth of the profiling datasets.

6.3.9. Polynomial Evaluation and Side Channel Analysis

Side Channel Analysis (SCA) is a class of attacks that exploits leakage of information from a cryptographic implementation during execution. To thwart it, masking is a common countermeasure. The principle is to randomly split every sensitive intermediate variable occurring in the computation into several shares and the number of shares, called the masking order, plays the role of a security parameter. The main issue while applying masking to protect a block cipher implementation is to specify an efficient scheme to secure the S-box computations. Several masking schemes, applicable for arbitrary orders, have been recently introduced. Most of them follow a similar approach originally introduced in the paper of Carlet et al published at FSE 2012; the S-box to protect is viewed as a polynomial and strategies are investigated which minimize the number of field multiplications which are not squarings. The paper [32] aims at presenting all these works

in a comprehensive way. The methods are discussed, their differences and similarities are identified and the remaining open problems are listed.

6.3.10. Redefining the Transparency Order

In [7], we consider the multi-bit Differential Power Analysis (DPA) in the Hamming weight model. In this regard, we revisit the definition of Transparency Order (TO) from the work of Prouff (FSE 2005) and find that the definition has certain limitations. Although this work has been quite well referred in the literature, surprisingly, these limitations remained unexplored for almost a decade. We analyse the definition from scratch, modify it and finally provide a definition with better insight that can theoretically capture DPA in Hamming weight model for hardware implementation with precharge logic. At the end, we confront the notion of (revised) transparency order with attack simulations in order to study to what extent the low transparency order of an s-box impacts the efficiency of a side channel attack against its processing.

SECRET Project-Team

7. New Results

7.1. Symmetric cryptology

Participants: Xavier Bonnetain, Anne Canteaut, Pascale Charpin, Sébastien Duval, Virginie Lallemand, Gaëtan Leurent, Nicky Mouha, María Naya Plasencia, Yann Rotella.

7.1.1. Block ciphers

Our recent results mainly concern either the analysis and design of lightweight block ciphers.
Recent results:

- 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 [13].
- Design of a new permutation for wide-block block ciphers: N. Mouha and S. Gueron have proposed a family of cryptographic permutations, named Simpira, that supports inputs of $128b$ bits, where b is a positive integer [50]. This wide-block permutation is mainly based on the AES round-function. It then achieves a very high throughput on virtually all modern 64-bit processors that have native instructions for AES.
- Analysis of the division property against block ciphers [42], [26]: A. Canteaut, together with C. Boura, gave a new approach to the division property, which has been recently introduced as a distinguishing property on block ciphers. This work provides a simpler and more general view of the division property which allows the attacker to take into account the characteristics of the building-blocks of the cipher. As an illustration, this new approach provides low-data distinguishers against reduced-round Present, which reach a much higher number of rounds than previously known distinguishers of the same type.
- Modes of operation for full disk encryption [52]: L. Khati, N. Mouha and D. Vergnaud have classified various FDE modes of operation according to their security in a setting where there is no space to store additional data, like an IV or a MAC value. They also introduce the notion of a diversifier, which does not require additional storage, but allows the plaintext of a particular sector to be encrypted into different ciphertexts.

7.1.2. Authenticated encryption and MACs

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]. The project-team is involved in a national cryptanalytic effort in this area led by the BRUTUS project funded by the ANR.

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

Recent results:

- Attack against π-Cipher : G. Leurent and his coauthors have presented a guess-and-determine attack against some variants of the π-Cipher family, which is a second-round candidate to the Caesar competition. More precisely, they showed a key recovery attack with time complexity little higher than $2^{4\omega}$, and low data complexity, against variants of the cipher with ω-bit words, when the internal permutation is reduced to 2.5 rounds out of 3.
- Improved generic attacks against hash-based MAC [20]
- Cryptanalysis of 7 (out of 8) rounds of the Chaskey MAC [54]. This work has led the designers of Chaskey to increase the number of rounds.

7.1.3. Stream ciphers

Stream ciphers provide an alternative to block-cipher-based encryption schemes. They are especially well-suited in applications which require either extremely fast encryption or a very low-cost hardware implementation.

Recent results:

- Design of encryption schemes for efficient homomorphic-ciphertext compression (see Section 5.1.3): A. Canteaut, M. Naya-Plasencia together with their coauthors have investigated the constraints on the symmetric cipher imposed by this application and they have proposed some solutions based on additive IV-based stream ciphers [44], [30].
- Cryptanalysis of the FLIP family of stream ciphers: S. Duval, V. Lallemand and Y. Rotella have exhibited an attack against a new family of stream ciphers intended for use in Fully Homomorphic Encryption systems, and proposed by Méaux et al. at Eurocrypt 2016 [48], [32]. More precisely, their attack applies to the early version of FLIP. It exploits the structure of the filter function and the constant internal state of the cipher. The proposed algorithm then recovers the secret key for the two instantiations originally proposed by Méaux et al.
- New types of correlation attacks against filter generators: A. Canteaut and Y. Rotella presented a new family of attacks against filter generators, which exploit a change of the primitive root defining the LFSR [45]. Most notably, an attack can often be mounted by considering non-bijective monomial mappings. In this setting, a divide-and-conquer strategy applies, based on a search within a multiplicative subgroup of $\mathbb{F}_{2^n}$ where n is the LFSR length. If the LFSR length is not a prime, a fast correlation involving a shorter LFSR can then be performed.

7.1.4. 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.

Recent results:

- Cryptographic properties of involutions: P. Charpin, together with S. Mesnager and S. Sarkar, has provided a rigorous study of involutions over the finite field of order 2^n which are relevant primitives for cryptographic designs [19]. Most notably, they have focused on the class of involutions defined by Dickson polynomials [61].
- Construction of a new family of permutations over binary fields of dimension $(4k+2)$ with good cryptographic properties. An interesting property is that this family includes as a specific case the only known APN permutation of an even number of variables [64].

- Construction of cryptographic permutations over finite fields with a sparse representation: P. Charpin, together with N. Cepak and E. Pasalic, exhibited permutations which are derived from sparse functions via linear translators [14].
- New methods for determining the differential spectrum of an Sbox: P. Charpin and G. Kyureghyan have proved that the whole differential spectrum of an Sbox can be determined without examining all derivatives of the mapping, but only the derivatives with respect to an element within a hyperplane [18]. Also, they have proved that, for mappings of a special shape, it is enough to consider the derivatives with respect to all elements within a suitable multiplicative subgroup of $\mathbb{F}_{2^n}$.

7.1.5. Side-channel attacks

Physical attacks must be taken into account in the evaluation of the security of lightweight primitives. Indeed, these primitives are often dedicated to IoT devices in pervasive environments, where an attacker has an easy access to the devices where the primitive is implemented.

Recent results:

- Differential fault attack against the block cipher PRIDE [53]: the efficiency of this attack mainly originate from the design of the linear layer of the cipher which relies on the interleaved construction.
- Study of the criteria to quantify the resistance offered by an Sbox to differential power analysis [17]. This work by K. Chakraborty and his coauthors shows that the classical criterion, called transparency order, has many limitations; an alternative definition is then proposed.

7.1.6. Security of Internet protocols

Cryptographic primitives are used to in key-exchange protocols such as TLS, IKE and SSH, to verify the integrity of the exchange. The recent works by K. Bhargavan and G. Leurent show the real-word impact of some recent theoretical cryptanalytic works.

Recent results:

- Impact of hash function collisions on the security of TLS: most practitioners believe that the hash function only need to resist preimage attacks for this use. However, K. Bhargavan and G. Leurent have shown that collisions in the hash function are sufficient to break the integrity of these protocols, and to impersonate some of the parties [41], [34]. Since many protocols still allow the use of MD5 or SHA-1 (for which collision attacks are known), this results in some practical attacks, and extends the real-world impact of the collision attacks against MD5 and SHA-1. This work has already influenced the latest TLS 1.3 draft, and the main TLS libraries are removing support of MD5 signatures.
- Use of block ciphers operating on small blocks: It is well-known that most modes of operation, like CBC, are not secure if the same key is used for encrypting $2^{n/2}$ blocks of plaintext, where n is the block size. But this threat has traditionally been dismissed as impractical, even for 64-bit blocks, since it requires some prior knowledge of the plaintext and even then, it only leaks a few secret bits per gigabyte. In this context, K. Bhargavan and G. Leurent demonstrated two concrete attacks that exploit such short block ciphers [40]. First, they presented an attack on the use of 3DES in HTTPS that can be used to recover a secret session cookie. Second, they showed how a similar attack on Blowfish can be used to recover HTTP BasicAuth credentials sent over OpenVPN connections.

7.2. Code-based cryptography

Participants: Rodolfo Canto Torres, Julia Chaulet, Thomas Debris, Adrien Hauteville, Ghazal Kachigar, Irene Márquez Corbella, Nicolas Sendrier, Jean-Pierre Tillich.

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, including against a quantum adversary, 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:

- J. Chaulet and N. Sendrier are working on the analysis Gallager's bit flipping algorithm for the decoding of QC-MDPC codes. A first outcome is an improved decoder with an adaptative threshold [47]. The ultimate goal of this work is to avoid side-channel attacks on QC-MDPC-McEliece by designing a failure-free constant-time decoder.
- We have started to explore whether generalized Reed-Solomon codes, and more generally MDS codes, can be used in a McEliece cryptosystem. We have first started by a fundamental work about MDS codes by first characterizing which MDS codes can be efficiently decoded with the rather general technique using error correcting pairs [25] We have also studied whether it is possible, if we know only a random generator matrix of a code admitting an error correcting pair, to recover the pair itself [55]. The latter problem is precisely the problem that an attacker wants to solve when he wants to perform a key attack on a McEliece system based on MDS codes admitting an error correcting pair. Finally, we have come up with what we believe to be a viable McEliece scheme based on Reed-Solomon codes by combining them with a generalized $U|U+V$ construction which hides at the same time the algebraic structure and even improves the decoding capacity of the code [57].
- Design of a new code-based stream cipher, named RankSynd, variant of Synd for the rank metric [49] and of the first Identity based Encryption Scheme relying on error correcting codes (paper currently under submission which is joint work of P. Gaborit, A. Hauteville, H. Phan and J.P. Tillich).
- Structural attacks against some variants of the McEliece cryptosystem based on subclasses of alternant/Goppa codes which admit a very compact public matrix, typically quasi-cyclic, quasi-dyadic, or quasi-monoidic matrices [22]. This result is obtained thanks to a new operation on codes called folding that exploits the knowledge of the automorphism group of the code [21].
- Cryptanalysis of a variant of McEliece cryptosystem based on polar codes [38].
- The previous work has been extended by exploring some structural properties of polar codes in [39]. In particular, we have been able to show that these codes have a very large automorphism group and have found an efficient way of counting the number of codewords of low weight.
- Cryptanalysis of all McEliece cryptosystems relying on algebraic geometry codes [73].
- Cryptanalysis of a code-based signature scheme proposed at PQCrypto 2013 by Baldi at al. [58]. This paper has received the best paper award of PQCrypto 2016.
- R. Canto Torres and N. Sendrier have investigated the information-set decoding algorithms applied to the case where the number of errors is sub-linear in the code length [46]. This situation appears in the analysis of the McEliece scheme based on quasi-cyclic Moderate Density Parity Check (MDPC) codes.
- We have also investigated other decoding techniques such as statistical decoding [74] or quantum algorithms [75]. The last work has led to the best known quantum algorithms for decoding a linear code.

7.3. Quantum Information

Participants: Xavier Bonnetain, Rémi Bricout, Kaushik Chakraborty, André Chailloux, Antoine Grospellier, Gaëtan Leurent, Anthony Leverrier, Vivien Londe, María Naya Plasencia, Jean-Pierre Tillich.

7.3.1. 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.

Two PhD theses started in September 2016 on this topic. First, Antoine Grospellier, co-advised by A. Leverrier and O. Fawzi (Ens Lyon), will study efficient decoding algorithms for quantum LDPC codes. Beyond their intrinsic interest for channel coding problems, such algorithms would be particularly relevant in the context of quantum fault-tolerance, since they would allow to considerably reduce the required overhead to obtain fault-tolerance in quantum computation. Vivien Londe is co-advised by A. Leverrier and G. Zémor (IMB) and his thesis is devoted to the design of better quantum LDPC codes: the main idea is to generalize the celebrated toric code of Kitaev by considering cellulations of manifolds in higher dimensions. A recent surprising result was that this approach leads to a much better behaviour than naively expected and a major challenge is to explore the mathematics behind this phenomenon in order to find even better constructions, or to uncover potential obstructions.

Recent results:

- Introduction of a new class of quantum LDPC codes, "Quantum expander codes", featuring a simple and very efficient decoding algorithm which can correct arbitrary patterns of errors of size scaling as the square-root of the length of the code. These are the first codes with constant rate for which such an efficient decoding algorithm is known [36], [59].

7.3.2. 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:

- A. Chailloux, together with colleagues from IRIF and Jerusalem, established the existence of quantum weak coin flipping with arbitrarily small bias [12].
- A. Chailloux and international collaborators performed an experimental verification of multipartite entanglement in quantum networks [24].
- A. Chailloux and collaborators established the optimal bounds for quantum weak oblivious transfer [15].
- Security analysis of quantum key distribution with continuous variables [35].

7.3.3. Relativistic cryptography

Two-party cryptographic tasks are well-known to be impossible without complexity assumptions, either in the classical or the quantum world. Remarkably, such no-go theorems become invalid when adding the physical assumption that no information can travel faster than the speed of light. This additional assumption gives rise to the emerging field of relativistic cryptography. We recently started investigating such questions through the task of bit commitment. In a paper in *Physical Review Letters* in 2015, K. Chakraborty, A. Chailloux and A. Leverrier developed a security proof for a simple and easily implementable protocol that can achieve arbitrarily long commitment times, thereby establishing that relativistic cryptography is a very practical solution.

André Chailloux was awarded an ANR "Jeune chercheur" to develop the field of relativistic cryptography [31].

Recent results:

- R. Bricout and A. Chailloux [70] considered explicit attacks against the relativistic protocol for bit commitment mentioned above and proved that the security analysis published in *Physical Review Letters* 2015 is essentially tight.
- A drawback of the relativistic bit commitment protocol is that it requires that all communications remain perfectly synchronized during the entire commitment time, and a single network failure leads to aborting the protocol. K. Chakraborty, A. Chailloux and A. Leverrier proposed a more robust version of the protocol allowing to deal with such network failures, a required feature in order to implement the protocol in realistic conditions [16], [71].

7.3.4. Quantum cryptanalysis of symmetric primitives

Symmetric cryptography seems at first sight much less affected in the post-quantum world than asymmetric cryptography: its main known threat is Grover's algorithm, which allows for an exhaustive key search in the square root of the normal complexity. For this reason, it is usually believed that doubling key lengths suffices to maintain an equivalent security in the post-quantum world. However, a lot of work is certainly required in the field of symmetric cryptography in order to "quantize" the classical families of attacks in an optimized way. M. Naya Plasencia has recently been awarded an ERC Starting grant for her project named QUASYModo on this topic.

Recent results:

- Differential and linear attacks in the quantum setting: G. Leurent, A. Leverrier and M. Naya Plasencia, in collaboration with M. Kaplan, have obtained some results on quantum versions of differential and linear cryptanalysis [23]. They show that it is usually possible to use quantum computations to obtain a quadratic speed-up for these attacks, but not for all variants. Therefore, the best attack in the classical world does not necessarily lead to the best quantum one.
- Application of Simon's algorithm to symmetric cryptanalysis [51], [33]: Leurent et al. also proved that several attacks can be dramatically sped up using a quantum procedure known as Simon's algorithm for finding the period of a function. As a first application, the most widely used modes of operation for authentication and authenticated encryption (e.g. CBC-MAC, PMAC, GMAC, GCM, and OCB) are completely broken in this security model. These quantum attacks are also applicable to many CAESAR candidates: CLOC, AEZ, COPA, OTR, POET, OMD, and Minalpher. Second, Simon's algorithm can also be applied to slide attacks, leading to an exponential speed-up of a classical symmetric cryptanalysis technique in the quantum model.

SPECFUN Project-Team

6. New Results

6.1. Formally certified computation of definite integrals

Assia Mahboubi and Thomas Sibut-Pinote, in collaboration with Guillaume Melquiond (Toccata), have developed a Coq library for the computation of intervals approximating the value of definite integrals for elementary mathematical functions. This library provides an automated tool which builds automatically a formal proof of the correctness of the output, that is: a formal proof that the interval contains the mathematical values and a formal proof of the integrability of the input function on the input interval. A description of this work was published in the proceeding of the ITP 2016 conference [13]. An extension to domains including singularities of the integrand is in progress.

6.2. Real closed fields

Assia Mahboubi has worked with Henri Lombardi (Université de Franche Comté) on a constructive axiomatization of real closed fields. For this purpose, they have proposed an equational theory based on virtual roots and close to the classical notion of local real closed rings. This is a first step toward a constructive understanding of o-minimal structures. This work has been accepted for publication in Contemporary Mathematics [19].

6.3. Combinatorial walks with small steps in the quarter plane

Alin Bostan and Frédéric Chyzak, together with Mark van Hoeij (Florida State University), Manuel Kauers (Johannes Kepler University), and Lucien Pech (former intern), have applied their algorithms on special functions to generate complete, quantitative results in the enumerative theory of combinatorial walks with small steps in the quarter plane [2]. They gave the first proof that differential equations conjectured years ago by Bostan and Kauers are indeed satisfied by the corresponding generating functions. They also obtained explicit hypergeometric expressions for the latter, and could provably determine which of the generating functions are transcendental or algebraic.

6.4. Multiple binomial sums

Multiple binomial sums form a large class of multi-indexed sequences, closed under partial summation, which contains most of the sequences obtained by multiple summation of products of binomial coefficients and also all the sequences with algebraic generating function. Alin Bostan and Pierre Lairez, together with Bruno Salvy (Inria and ENS Lyon), have studied in [5] the representation of the generating functions of binomial sums by integrals of rational functions. The outcome is twofold. Firstly, we show that a univariate sequence is a multiple binomial sum if and only if its generating function is the diagonal of a rational function. Secondly, we propose algorithms that decide the equality of multiple binomial sums and that compute recurrence relations for them. In conjunction with geometric simplifications of the integral representations, this approach behaves well in practice. The process avoids the computation of certificates and the problem of the appearance of spurious singularities that afflicts discrete creative telescoping, both in theory and in practice.

6.5. Algebraic diagonals and walks

The diagonal of a multivariate power series F is the univariate power series $\mathsf{Diag}F$ generated by the diagonal terms of F. Diagonals form an important class of power series; they occur frequently in number theory, theoretical physics and enumerative combinatorics. In [35], Alin Bostan and Louis Dumont, together with Bruno Salvy (Inria and ENS Lyon), have studied algorithmic questions related to diagonals in the case where F is the Taylor expansion of a bivariate rational function. It is classical that in this case $\mathsf{Diag}F$ is an algebraic function. We propose an algorithm that computes an annihilating polynomial for $\mathsf{Diag}F$. We give a precise bound on the size of this polynomial and show that generically, this polynomial is the minimal polynomial and that its size reaches the bound. The algorithm runs in time quasi-linear in this bound, which grows exponentially with the degree of the input rational function. We then address the related problem of enumerating directed lattice walks. The insight given by our study leads to a new method for expanding the generating power series of bridges, excursions and meanders. We show that their first N terms can be computed in quasi-linear complexity in N, without first computing a very large polynomial equation. An extended version of this work is presented in [3].

6.6. A human proof of the Gessel conjecture

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 notoriously difficult case concerns the so-called *Gessel walks*: they 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 such 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. Alin Bostan, together with Irina Kurkova (Univ. Paris 6) and Kilian Raschel (CNRS and Univ. Tours), have proposed in [4] the first "human proofs" of these results. They are derived from a new expression for the generating function of Gessel walks in terms of special functions.

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

Small step walks in 2D are by now quite well understood, but almost everything remains to be done in higher dimensions. Alin Bostan, together with Mireille Bousquet-Mélou (CNRS and Univ. Bordeaux), Manuel Kauers (Johannes Kepler Univ.) and Stephen Melczer (Univ. of Waterloo and ENS Lyon), have explored in [1] 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.

6.8. Computation of the similarity class of the p-curvature

The p-curvature of a system of linear differential equations in positive characteristic p is a matrix that measures how far the system is from having a basis of polynomial solutions. Alin Bostan, together with Xavier Caruso (CNRS and Univ. Rennes) and Éric Schost (Univ. Waterloo), have showed in [10] that the similarity class of the p-curvature can be determined without computing the p-curvature itself. More precisely, we have designed an algorithm that computes the invariant factors of the p-curvature in time quasi-linear in $\sqrt{p}$. This is much less than the size of the p-curvature, which is generally linear in p. The new algorithm allowed to answer a question originating from the study of the Ising model in statistical physics.

6.9. Efficient algorithms for mixed creative telescoping

Creative telescoping is a powerful computer algebra paradigm –initiated by Doron Zeilberger in the 90's– for dealing with definite integrals and sums with parameters. Alin Bostan and Louis Dumont, together with Bruno Salvy (Inria and ENS Lyon), have addressed in [12] the mixed continuous–discrete case, and have focussed on the integration of bivariate hypergeometric-hyperexponential terms. We have designed a new creative telescoping algorithm operating on this class of inputs, based on a Hermite-like reduction procedure. The new algorithm has two nice features: it is efficient and it delivers, for a suitable representation of the input, a minimal-order telescoper. Its analysis reveals tight bounds on the sizes of the telescoper it produces.

6.10. Fast computation of the Nth term of an algebraic series over a finite prime field

Alin Bostan and Philippe Dumas, together with Gilles Christol (IMJ), have addressed in [11] the question of computing one selected term of an algebraic power series. In characteristic zero, the best algorithm currently known for computing the Nth coefficient of an algebraic series uses differential equations and has arithmetic complexity quasi-linear in $\sqrt{N}$. We show that over a prime field of positive characteristic p, the complexity can be lowered to $O(\log N)$. The mathematical basis for this dramatic improvement is a classical theorem stating that a formal power series with coefficients in a finite field is algebraic if and only if the sequence of its coefficients can be generated by an automaton. We revisit and enhance two constructive proofs of this result for finite prime fields. The first proof uses Mahler equations, whose sizes appear to be prohibitively large. The second proof relies on diagonals of rational functions; we turn it into an efficient algorithm, of complexity linear in $\log N$ and quasi-linear in p.

6.11. Formal methods for cryptocurrencies

Georges Gonthier and Thomas Sibut-Pinote, along with a team of researchers from Microsoft Research and Inria, participated in a hackathon internal to Microsoft Research with the goal to apply formal methods to the verification of the smart contracts involved in the Ethereum platform. They outlined a framework to analyze and verify both the runtime safety and the functional correctness of Ethereum contracts by translation to F*, a functional programming language aimed at program verification. This work was published in the proceedings of the PLAS 2016 conference [9].

6.12. Computing solutions of linear Mahler equations

Mahler equations relate evaluations of the same function f at iterated bth powers of the variable. They arise in particular in the study of automatic sequences and in the complexity analysis of divide-and-conquer algorithms. Recently, the problem of solving Mahler equations in closed form has occurred in connection with number-theoretic questions. A difficulty in the manipulation of Mahler equations is the exponential blow-up of degrees when applying a Mahler operator to a polynomial. In [17], Frédéric Chyzak and Philippe Dumas, together with Thomas Dreyfus (Université Claude Bernard Lyon 1) and Marc Mezzarobba (visiting scientist from UPMC), have presented algorithms for solving linear Mahler equations for series, polynomials, and rational functions, and have obtained polynomial-time complexity under a mild assumption.

6.13. Formal solutions of singularly perturbed linear differential systems

Suzy Maddah, together with Boulay Barkatou (Université de Limoges), has obtained algorithms for computing formal invariants of singularly-perturbed linear differential systems [20].

VEGAS Project-Team

6. New Results

6.1. Non-linear Computational Geometry

Participants: Laurent Dupont, Rémi Imbach, Sylvain Lazard, Guillaume Moroz, Marc Pouget.

6.1.1. Numeric and Certified Algorithm for the Topology of the Projection of a Smooth Space Curve

Let a smooth real analytic curve embedded in $\mathbb{R}^3$ be defined as the solution of real analytic equations of the form $P(x,y,z) = Q(x,y,z) = 0$ or $P(x,y,z) = \frac{\partial P}{\partial z} = 0$. Our main objective is to describe its projection $\mathcal{C}$ onto the (x,y)-plane. In general, the curve $\mathcal{C}$ is not a regular submanifold of $\mathbb{R}^2$ and describing it requires to isolate the points of its singularity locus Σ.

In previous work, we have shown how to describe the set of singularities Σ of $\mathcal{C}$ as regular solutions of a so-called ball system suitable for a numerical subdivision solver. In our current work, the space curve is first enclosed in a set of boxes with a certified path-tracker to restrict the domain where the ball system is solved. Boxes around singular points are then computed such that the correct topology of the curve inside these boxes can be deduced from the intersections of the curve with their boundaries. The tracking of the space curve is then used to connect the smooth branches to the singular points. The subdivision of the plane induced by the curve is encoded as an extended planar combinatorial map allowing point location. This work is not already published but has been presented by R. Imbach at the Summer Workshop on Interval Methods (https://swim2016.sciencesconf.org/).

The technical report [28] describes the software SubdivisionSolver (see Section 5.2) used within this project.

6.1.2. A Fast Algorithm for Computing the Truncated Resultant

Let P and Q be two polynomials in $\mathbb{K}[x,y]$ with degree at most d, where $\mathbb{K}$ is a field. Denoting by $R \in \mathbb{K}[x]$ the resultant of P and Q with respect to y, we present an algorithm to compute $R \mod x^k$ in $\widetilde{O}(kd)$ arithmetic operations in $\mathbb{K}$, where the $\widetilde{O}$ notation indicates that we omit polylogarithmic factors. This is an improvement over state-of-the-art algorithms that require to compute R in $\widetilde{O}(d^3)$ operations before computing its first k coefficients [24].

This work was done in collaboration with Éric Schost (Waterloo University, Canda).

6.1.3. Quadric Arrangement in Classifying Rigid Motions of a 3D Digital Image

Rigid motions are fundamental operations in image processing. While bijective and isometric in $\mathbb{R}^3$, they lose these properties when digitized in $\mathbb{Z}^3$. To understand how the digitization of 3D rigid motions affects the topology and geometry of a chosen image patch, we classify the rigid motions according to their effect on the image patch. This classification can be described by an arrangement of hypersurfaces in the parameter space of 3D rigid motions of dimension six. However, its high dimensionality and the existence of degenerate cases make a direct application of classical techniques, such as cylindrical algebraic decomposition or critical point method, difficult. We show that this problem can be first reduced to computing sample points in an arrangement of quadrics in the 3D parameter space of rotations. Then we recover information about the three remaining parameters of translation. We implemented an ad-hoc variant of state-of-the-art algorithms and applied it to an image patch of cardinality 7. This leads to an arrangement of 81 quadrics and we recovered the classification in less than one hour on a machine equipped with 40 cores [25].

This work was done in collaboration with Kacper Pluta (LIGM - Laboratoire d'Informatique Gaspard-Monge), Yukiko Kenmochi (LIGM - Laboratoire d'Informatique Gaspard-Monge), Pascal Romon (LAMA - Laboratoire d'Analyse et de Mathématiques Appliquées).

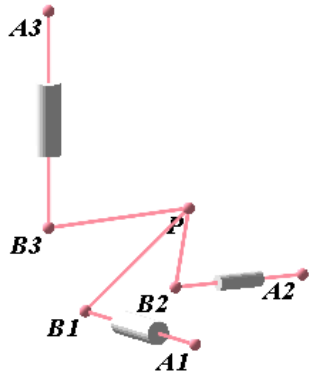

Figure 1. A configuration of the orthoglide manipulator has three orthogonal prismatic joints.

6.1.4. *Influence of the Trajectory Planning on the Accuracy of the Orthoglide 5-axis manipulator*

Usually, the accuracy of parallel manipulators depends on the architecture of the robot, the design parameters, the trajectory planning and the location of the path in the workspace. This paper reports the influence of static and dynamic parameters in computing the error in the pose associated with the trajectory planning made and analyzed with the Orthoglide 5-axis (Figure 1). An error model is proposed based on the joint parameters (velocity and acceleration) and experimental data coming from the Orthoglide 5-axis. Newton and Gröbner based elimination methods are used to project the joint error in the workspace to check the accuracy/error in the Cartesian space. For the analysis, five similar trajectories with different locations inside the workspace are defined using fifth order polynomial equation for the trajectory planning. It is shown that the accuracy of the robot depends on the location of the path as well as the starting and the ending posture of the manipulator due to the acceleration parameters [23].

This work was done in collaboration with Ranjan Jha (IRCCyN - Institut de Recherche en Communications et en Cybernétique de Nantes), Damien Chablat (IRCCyN - Institut de Recherche en Communications et en Cybernétique de Nantes), Fabrice Rouillier (Inria).

6.1.5. *Solving Bivariate Systems and Topology of Plane Algebraic Curves*

In the context of our algorithm Isotop for computing the topology of plane algebraic curves (see Section 5.1), we work on the problem of solving a system of two bivariate polynomials. We are interested in certified numerical approximations or, more precisely, isolating boxes of the solutions. But we are also interested in computing, as intermediate symbolic objects, a Rational Univariate Representation (RUR) that is, roughly speaking, a univariate polynomial and two rational functions that map the roots of the univariate polynomial to the two coordinates of the solutions of the system. RURs are relevant symbolic objects because they allow to the transformation of many queries on the system into queries on univariate polynomials. However, such representations require the computation of a separating form for the system, that is a linear combination of the variables that takes different values when evaluated at the distinct solutions of the system.

We published this year [11] results showing that, given two polynomials of degree at most d with integer coefficients of bitsize at most τ, (i) a separating form, (ii) the associated RUR, and (iii) isolating boxes of the solutions can be computed in, respectively, $\widetilde{O}_B(d^5 + d^5\tau)$, bit operations in the worst case, where $\widetilde{O}$ refers to the complexity where polylogarithmic factors are omitted and O_B refers to the bit complexity. Furthermore, we also presented probabilistic Las Vegas variants of problems (i) and (ii), which have expected bit complexity $\widetilde{O}_B(d^5 + d^4\tau)$. We also showed that these complexities are "morally" optimal in the sense of that improving them would essentially require to improve bounds on several other fundamental problems (on resultants and roots isolation of univariate polynomials) that have hold for decades. These progresses are substential since, when we started woriking on these problems, their best know complexities were in $\widetilde{O}_B(d^{12} + d^{10}\tau^2)$ (2009).

This work was done in collaboration with Yacine Bouzidi (Inria Lille), Michael Sagraloff (MPII Sarrebruken, Germany) and Fabrice Rouillier (Inria Rocquencourt).

6.1.6. 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. We previously proposed a new algorithm for this problem, using a characterization of 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], [32]. Unfortunately, our new algorithm is not yet efficient in practice. This year, we made several improvements on this algorithm. First, we decreased from 11 to 4 the degree of a critical polynomial that we need to solve and whose solutions induce the coefficients in some other polynomials appearing later in the computations. Second, we implemented Descartes' algorithm for isolating the real roots of univariate polynomials in the case where the coefficients belong to extensions of $\mathbb{Q}$ generated by at most two square roots. Furthermore, we are currently implementing the generalization of that algorithm when the coefficients belong to extensions of $\mathbb{Q}$ generated by one root of an arbitrary polynomial. We are also interested by extensions decomposable in extensions of degree 2. These undergoing improvements should allow us to compute more directly the wanted reflection point, thus avoiding problematic approximations and making the overall algorithm tractable.

6.2. Non-Euclidean Computational Geometry

Participants: Iordan Iordanov, Monique Teillaud, Gert Vegter.

6.2.1. Closed Flat Orbifolds

The work on Delaunay triangulations of flat d-dimensional orbifolds, started several years ago in the Geometrica project team in Sophia Antipolis, was finalized this year [13].

We give a definition of the Delaunay triangulation of a point set in a closed Euclidean d-manifold, i.e. a compact quotient space of the Euclidean space for a discrete group of isometries (a so-called Bieberbach group or crystallographic group). We describe a geometric criterion to check whether a partition of the manifold actually forms a triangulation (which subsumes that it is a simplicial complex). We provide an incremental algorithm to compute the Delaunay triangulation of the manifold defined by a given set of input points, if it exists. Otherwise, the algorithm returns the Delaunay triangulation of a finite-sheeted covering space of the manifold. The algorithm has optimal randomized worst-case time and space complexity. It extends to closed Euclidean orbifolds. To the best of our knowledge, this is the first general result on this topic.

6.2.2. Closed Orientable Hyperbolic Surfaces

Motivated by applications in various fields, some packages to compute periodic Delaunay triangulations in the 2D and 3D Euclidean spaces have been introduced in the CGAL library and have attracted a number of users. To the best of our knowledge, no software is available to compute periodic triangulations in a hyperbolic space, though they are also used in diverse fields, such as physics, solid modeling, cosmological models, neuromathematics.

This would be a natural extension: 2D Euclidean periodic triangulations can be seen as triangulations of the two-dimensional (flat) torus of genus one; similarly, periodic triangulations in the hyperbolic plane can be seen as triangulations of hyperbolic surfaces. A closed orientable hyperbolic surface is the quotient of the hyperbolic plane under the action of a Fuchsian group only containing hyperbolic translations. Intuition is challenged there, in particular because such groups are non-Abelian in general.

We have obtained some theoretical results on Delaunay triangulations of general closed orientable hyperbolic surfaces, and we have investigated algorithms in the specific case of the Bolza surface, a hyperbolic surface with the simplest possible topology, as it is homeomorphic to a genus-two torus [20]. We are now studying more practical aspects and we propose a first implementation of an incremental construction of Delaunay triangulations of the Bolza surface [30].

6.3. Probabilistic Analysis of Geometric Data Structures and Algorithms

Participants: Olivier Devillers, Louis Noizet.

6.3.1. Stretch Factor of Long Paths in a Planar Poisson-Delaunay Triangulation

Let $X := X_n \cup \{(0,0),(1,0)\}$, where X_n is a planar Poisson point process of intensity n. We provide a first non-trivial lower bound for the distance between the expected length of the shortest path between $(0,0)$ and $(1,0)$ in the Delaunay triangulation associated with X when the intensity of X_n goes to infinity. Experimental values indicate that the correct value is about 1.04. We also prove that the expected number of Delaunay edges crossed by the line segment $[(0,0),(1,0)]$ is equivalent to $2.16\sqrt{n}$ and that the expected length of a particular path converges to 1.18 giving an upper bound on the stretch factor [26].

This work was done in collaboration with Nicolas Chenavier (Université Littoral Côte d'Opale).

6.3.2. Walking in a Planar Poisson-Delaunay Triangulation: Shortcuts in the Voronoi Path

Let X_n be a planar Poisson point process of intensity n. We give a new proof that the expected length of the Voronoi path between $(0,0)$ and $(1,0)$ in the Delaunay triangulation associated with X_n is $\frac{4}{\pi} \simeq 1.27$ when n goes to infinity; and we also prove that the variance of this length is $O(1/\sqrt{n})$. We investigate the length of possible shortcuts in this path, and defined a shortened Voronoi path whose expected length can be expressed as an integral that is numerically evaluated to $\simeq 1.16$. The shortened Voronoi path has the property to be *locally defined*; and is shorter than the previously known locally defined path in Delaunay triangulation such as the upper path whose expected length is $35/3\pi^2 \simeq 1.18$ [27].

6.3.3. Expected Length of the Voronoi Path in a High Dimensional Poisson-Delaunay Triangulation

Let X_n be a d dimensional Poisson point process of intensity n. We prove that the expected length of the Voronoi path between two points at distance 1 in the Delaunay triangulation associated with X_n is $\sqrt{\frac{2d}{\pi}} + O(d^{-\frac{1}{2}})$ for all $n \in \mathbb{N}$ and $d \to \infty$. In any dimension, we provide a precise interval containing the exact value, in 3D the expected length is between 1.4977 and 1.50007 [31].

This work was done in collaboration with Pedro Machado Manhães De Castro (Centro de Informática da Universidade Federal de Pernambuco).

6.4. Classical Computational Geometry and Graph Drawing

Participants: Olivier Devillers, Sylvain Lazard.

6.4.1. Monotone Simultaneous Path Embeddings in $\mathbb{R}^d$

We study the following problem: Given k paths that share the same vertex set, is there a simultaneous geometric embedding of these paths such that each individual drawing is monotone in some direction? We prove that for any dimension $d \geq 2$, there is a set of $d+1$ paths that does not admit a monotone simultaneous geometric embedding [21].

This work was done in collaboration with David Bremner (U. New Brunswick), Marc Glisse (Inria Datashape), Giuseppe Liotta (U. Perugia), Tamara Mchedlidze (Karlsruhe Institute for Technology), Sue Whitesides (U. Victoria), and Stephen Wismath (U. Lethbridge).

6.4.2. Analysis of Farthest Point Sampling for Approximating Geodesics in a Graph

A standard way to approximate the distance between two vertices p and q in a graph is to compute a shortest path from p to q that goes through one of k sources, which are well-chosen vertices. Precomputing the distance between each of the k sources to all vertices yields an efficient computation of approximate distances between any two vertices. One standard method for choosing k sources is the so-called *Farthest Point Sampling* (FPS), which starts with a random vertex as the first source, and iteratively selects the farthest vertex from the already selected sources.

We analyzed the stretch factor $\mathcal{F}_{\text{FPS}}$ of approximate geodesics computed using FPS, which is the maximum, over all pairs of distinct vertices, of their approximated distance over their geodesic distance in the graph. We showed that $\mathcal{F}_{\text{FPS}}$ can be bounded in terms of the minimal value $\mathcal{F}^*$ of the stretch factor obtained using an optimal placement of k sources as $\mathcal{F}_{\text{FPS}} \leq 2r_e^2\mathcal{F}^* + 2r_e^2 + 8r_e + 1$, where r_e is the length ratio of longest edge over the shortest edge in the graph. We further showed that the factor r_e is not an artefact of the analysis by providing a class of graphs for which $\mathcal{F}_{\text{FPS}} \geq \frac{1}{2}r_e\mathcal{F}^*$ [18].

This work was done in collaboration with Pegah Kamousi (Université Libre de Bruxelles), Anil Maheshwari (Carleton University), and Stefanie Wuhrer (Inria Grenoble Rhône-Alpes).

6.4.3. 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 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 [10].

This work was done in collaboration with Dominique Attali (CNRS, Grenoble) and Marc Glisse (Inria Datashape).

CAIRN Project-Team

7. New Results

7.1. Reconfigurable Architecture Design

7.1.1. Dynamic Reconfiguration Support in FPGA

Participants: Olivier Sentieys, 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 developed 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.

In [4], we proposed a design flow for generating compressed configuration bitstreams abstracted from their final position on the logic fabric, the Virtual Bit-Streams (VBS). 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 VPR (Versatile Place and Route) framework was expanded to include bitstream generation features. The configuration stream format was proposed along with its associated decoding architecture. 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.

The emergence of 2.5D and 3D packaging technologies enables the integration of FPGA dice into more complex systems. Both heterogeneous manycore designs, which include an FPGA layer, and interposer-based multi-FPGA systems support the inclusion of reconfigurable hardware in 3D-stacked integrated circuits. In these architectures, the communication between FPGA dice or between FPGA and fixed-function layers often takes place through dedicated communication interfaces spread over the FPGA logic fabric, as opposed to an I/O ring around the fabric. In [39], we investigate the effect of organizing FPGA fabric I/O into coarse-grained interface blocks distributed throughout the FPGA fabric. Specifically, we consider the quality of results for the placement and routing phases of the FPGA physical design flow. We evaluate the routing of I/O signals of large applications through dedicated interface blocks at various granularities in the logic fabric, and study its implications on the critical path delay of routed designs. We show that the impact of such I/O routing is limited and can improve chip routability and circuit delay in many cases.

7.1.2. Hardware Accelerated Simulation of Heterogeneous Platforms

Participant: François Charot.

When considering designing heterogeneous multi-core platforms, the number of possible design combinations leads to a huge design space, with subtle trade-offs and design interactions. To reason about what design is best for a given target application requires detailed simulation of many different possible solutions. Simulation frameworks exist (such as gem5) and are commonly used to carry out these simulations. Unfortunately, these are purely software-based approaches and they do not allow a real exploration of the design space. Moreover, they do not really support highly heterogeneous multi-core architectures. These limitations motivate the study of the use of hardware to accelerate the simulation, and in particular of FPGA components. In this context, we are currently investigating the possibility of building hardware accelerated simulators using the HAsim simulation infrastructure, jointly developed by MIT and Intel. HAsim is a FPGA-accelerated simulator that is able to simulate a multicore with a high-detailed pipeline, cache hierarchy and detailed on-chip network on a single FPGA. We work on integrating a model of the RISC-V instruction set architecture in the HAsim infrastructure. This work is done with the perspective of studying hardware accelerated simulation of heterogeneous multicore architectures mixing RISC-V cores and hardware accelerators.

7.1.3. Optical Interconnections for 3D Multiprocessor Architectures

Participants: Jiating Luo, Ashraf El-Antably, Pham Van Dung, Cédric Killian, Daniel Chillet, Olivier Sentieys.

To address the issue of interconnection bottleneck in multiprocessor on a single chip, we study how an Optical Network-on-Chip (ONoC) can leverage 3D technology by stacking a specific photonics die. The objectives of this study target: i) the definition of a generic architecture including both electrical and optical components, ii) the interface between electrical and optical domains, iii) the definition of strategies (communication protocol) to manage this communication medium, and iv) new techniques to manage and reduce the power consumption of optical communications. The first point is required to ensure that electrical and optical components can be used together to define a global architecture. Indeed, optical components are generally larger than electrical components, so a trade-off must be found between the size of optical and electrical parts. For example, if the need in terms of communications is high, several waveguides and wavelengths must be necessary, and can lead to an optical area larger than the footprint of a single processor. In this case, a solution is to connect (through the optical NoC) clusters of processors rather than each single processor. For the second point, we study how the interface can be designed to take applications needs into account. From the different possible interface designs, we extract a high-level performance model of optical communications from losses induced by all optical components to efficiently manage Laser parameters. Then, the third point concerns the definition of high-level mechanisms which can handle the allocation of the communication medium for each data transfer between tasks. This part consists in defining the protocol of wavelength allocation. Indeed, the optical wavelengths are a shared resource between all the electrical computing clusters and are allocated at run time according to application needs and quality of service. The last point concerns the definition of techniques allowing to reduce the power consumption of on-chip optical communications. The power of each Laser can be dynamically tuned in the optical/electrical interface at run time for a given targeted bit-error-rate. Due to the relatively high power consumption of such integrated Laser, we study how to define adequate policies able to adapt the laser power to the signal losses.

We are currently designing an Optical-Network-Interface (ONI) to connect one processor, or a cluster of several processors, to the optical communication medium. This interface, constrained by the 10 Gb/s data-rate of the Lasers, integrates Error Correcting Codes and a communication manager. This manager can select, at run-time, the communication mode to use depending on timing or power constraints. Indeed, as the use of ECC is based on redundant bits, it increases the transmission time, but saves power for a given Bit Error Rate (BER). Moreover, our ONI allows for data to be sent using several wavelengths in parallel, hence increasing transmission bandwidth.

However, multiple signals sharing simultaneously a waveguide can lead to inter-channel crosstalk noise. This problem impacts the Signal to Noise Ratio (SNR) of the optical signal, which leads to an increase in the Bit Error Rate (BER) at the receiver side. In [40], [59], we proposed a Wavelength Allocation (WA) method allowing to search for performance and energy trade-offs based on application constraints. We showed that for a 16-core WDM ring-based ONoC architecture using 12 wavelengths, more than 100,000 allocation solutions exist and only 51 are on a Pareto front giving a tradeoff between execution time and energy per bit (derived from the BER). The optimized solutions reached reduce the execution time by 37% or the energy from 7,6fJ/bit to 4,4fJ/bit.

7.1.4. Communication-Based Power Modelling for Heterogeneous Multiprocessor Architectures

Participants: Baptiste Roux, Olivier Sentieys, Steven Derrien.

Programming heterogeneous multiprocessor architectures is a real challenge dealing with a huge design space. Computer-aided design and development tools try to circumvent this issue by simplifying instantiation mechanisms. However, energy consumption is not well supported in most of these tools due to the difficulty to obtain fast and accurate power estimation. To this aim, in [46] we proposed and validated a power model for such platforms. The methodology is based on micro-benchmarking to estimate the model parameters. The energy model mainly relies on the energy overheads induced by communications between processors in a

parallel application. Power modelling and micro-benchmarks are validated using a Zynq-based heterogeneous architecture showing the accuracy of the model for several tested synthetic applications.

7.1.5. Arithmetic Operators for Cryptography and Fault-Tolerance

Participants: Arnaud Tisserand, Emmanuel Casseau, Pierre Guilloux, Karim Bigou, Gabriel Gallin, Audrey Lucas, Franck Bucheron, Jérémie Métairie.

Arithmetic Operators for Fast and Secure Cryptography.

Our paper [21], published in IEEE Transactions on Computers, extends our fast RNS modular inversion for finite fields arithmetic published at CHES 2013 conference. It is based on the binary version of the plus-minus Euclidean algorithm. In the context of elliptic curve cryptography (*i.e.* 160–550 bits finite fields), it significantly speeds-up modular inversions. In this extension, we propose an improved version based on both radix 2 and radix 3. This new algorithm leads to 30 % speed-up for a maximal area overhead about 4 % on Virtex 5 FPGAs. This work was done in the ANR PAVOIS project.

Our paper [32], presented at ARITH-23, presents an hybrid representation of large integers, or prime field elements, combining both positional and residue number systems (RNS). Our *hybrid position-residues* (HPR) number system mixes a high-radix positional representation and digits represented in RNS. RNS offers an important source of parallelism for addition, subtraction and multiplication operations. But, due to its non-positional property, it makes comparisons and modular reductions more costly than in a positional number system. HPR offers various trade-offs between internal parallelism and the efficiency of operations requiring position information. Our current application domain is asymmetric cryptography where HPR significantly reduces the cost of some modular operations compared to state-of-the-art RNS solutions. This work was done in the ANR PAVOIS project.

An ASIC circuit has been implemented in the 65nm ST CMOS technology and sent to fabrication in June 2016 (chip delivery is expected for January 2017). The implemented cryptoprocessor was designed for 256-bit prime finite fields elements and generic curves. It embeds: 1 multiplier, 1 adder and 1 inversion units for field-level computations. Various algorithms for scalar multiplication primitives can be programmed in software for curve-level computations. It was designed to evaluate algorithmic and arithmetic protections against side channel attacks (there is no hardware protection embedded in this ASIC version). This work was done in the ANR PAVOIS project.

In the HAH project, funded by CominLabs and Lebesgue Labex, we study hardware implementation of cryptoprocessors for hyperelliptic curves. The poster [61] presents the current state of the project for FPGA implementations.

Arithmetic Operators for Fault-Tolerance.

Various methods have been proposed for fault detection and fault tolerance in digital integrated circuits. In the case of *arithmetic circuits*, the selection of an efficient method depends on several elements: type of operation, type(s) of operand(s), computation algorithms, internal representations of numbers, optimizations at architecture and circuit levels, and acceptable accuracy level (i.e. mathematical error) of the result(s) including both rounding errors and errors due to the faults. High-level mathematical models are not sufficient to capture the effect of faults in arithmetic circuits. Simulation of intensive fault scenarios in all components of the arithmetic circuit (data-path, control, gates with important fan-out such as some partial products generation in large multipliers, etc.) is widely used. But cycle accurate and bit accurate software simulations at gate level are too slow for large circuits and numerous fault scenarios. *FPGA emulation* is a popular method to speed-up fault simulation.

We are developing an hardware-software platform dedicated to fault emulation for ASIC arithmetic circuits. The platform is based on a parallel cluster of Zynq FPGA cards and a Linux server. Various arithmetic circuits and fault models will be demonstrated in the context of digital signal and image processing. Our paper [57], presented at Compas, describes the very first version of our platform. This platform has also been presented in a poster at GDR SoC-SiP [58] and in a Demo Night at DASIP [56]. This work was done in the ANR ARDyT and Reliasic projects.

7.1.6. Adaptive Overclocking, Error Correction, and Voltage Over-Scaling for Error-Resilient Applications

Participants: Rengarajan Ragavan, Benjamin Barrois, Cédric Killian, Olivier Sentieys.

Error detection and correction based on double-sampling is used as common technique to handle timing errors while scaling V_{dd} for energy efficiency. Implementation and advantages of double-sampling technique in FPGAs are simpler and significant compared to the conventional highly pipelined processors due to the higher flexibility of the reconfigurable architectures. It is common practice to insert shadow flipflop in the critical paths of the design, which will fail while scaling down the supply voltage, or to correct timing errors while over clocking the datapaths. Overclocking, and error detection and correction capabilities of these methods are limited due to the fixed speculation window used by these methods. In [44], we presented a Dynamic Speculation Window in double-sampling for timing error detection and correction in FPGAs. The proposed method employs online slack measurement and conventional shadow flipflop approach to adaptively overclock the design and also to detect and correct timing errors due to temperature and other variability effects. We demonstrated this method in the Xilinx VC707 Virtex 7 FPGA for various benchmarks. We achieved maximum of 71% overclocking for unsigned 32-bit multiplier with the area overhead of 1.9% LUTs and 1.7% FFs.

Voltage scaling has been used as a prominent technique to improve energy efficiency in digital systems, scaling down supply voltage effects in quadratic reduction in energy consumption of the system. Reducing supply voltage induces timing errors in the system that are corrected through additional error detection and correction circuits. In [43], we proposed voltage over-scaling based approximate operators for applications that can tolerate errors. We characterized the basic arithmetic operators using different operating triads (combination of supply voltage, body-biasing scheme and clock frequency) to generate models for approximate operators. Error-resilient applications can be mapped with the generated approximate operator models to achieve optimum trade-off between energy efficiency and error margin. Based on the dynamic speculation technique, best possible operating triad is chosen at runtime based on the user definable error tolerance margin of the application. In our experiments in 28nm FDSOI, we achieved maximum energy efficiency of 89% for basic operators like 8-bit and 16-bit adders at the cost of 20% Bit Error Rate (ratio of faulty bits over total bits) by operating them in near-threshold regime.

7.2. Compilation and Synthesis for Reconfigurable Platform

7.2.1. Adaptive dynamic compilation for low power embedded systems

Participants: Steven Derrien, Simon Rokicki.

Dynamic binary translation (DBT) consists in translating – at runtime – a program written for a given instruction set to another instruction set. Dynamic Translation was initially proposed as a means to enable code portability between different instruction sets and can be implemented in software or hardware. DBT is also used to improve the energy efficiency of high performance processors, as an alternative to out-of-order microarchitectures. In this context, DBT is used to uncover instruction level parallelism (ILP) in the binary program, and then target an energy efficient wide issue VLIW architecture. This approach is used in Transmeta Crusoe [75] and NVidia Denver [68] processors. Since DBT operates at runtime, its execution time is directly perceptible by the user, hence severely constrained. As a matter of fact, this overhead has often been reported to have a huge impact on actual performance, and is considered as being the main weakness of DBT based solutions. This is particularly true when targeting a VLIW processor: the quality of the generated code depends on efficient scheduling; unfortunately scheduling is known to be the most time-consuming component of a JIT compiler or DBT. Improving the responsiveness of such DBT systems is therefore a key research challenge. This is however made very difficult by the lack of open research tools or platform to experiment with such platforms. In this work, we have been addressing these two issues by developing an open hardware/software platform supporting DBT. The platform was designed using HLS tools and validated on a FPGA board. The DBT uses RISC-V as host ISA, and can target varying issue width VLIW architectures. Our platform uses custom hardware accelerators to improve the reactivity of our optimizing DBT flow. Our results show that, compared to a software implementation, our approach offers speed-up by 8× while consuming 18× less energy.

7.2.2. Leveraging Power Spectral Density for Scalable System-Level Accuracy Evaluation

Participants: Benjamin Barrois, Olivier Sentieys.

The choice of fixed-point word-lengths critically impacts the system performance by impacting the quality of computation, its energy, speed and area. Making a good choice of fixed-point word-length generally requires solving an NP-hard problem by exploring a vast search space. Therefore, the entire fixed-point refinement process becomes critically dependent on evaluating the effects of accuracy degradation. In [30], a novel technique for the system-level evaluation of fixed-point systems, which is more scalable and that renders better accuracy, was proposed. This technique makes use of the information hidden in the power-spectral density of quantization noises. It is shown to be very effective in systems consisting of more than one frequency sensitive components. Compared to state-of-the-art hierarchical methods that are agnostic to the quantization noise spectrum, we show that the proposed approach is $5\times$ to $500\times$ more accurate on some representative signal processing kernels.

7.2.3. Approximate Computing

Participants: Benjamin Barrois, Olivier Sentieys.

Many applications are error-resilient, allowing for the introduction of approximations in the calculations, as long as a certain accuracy target is met. Traditionally, fixed-point arithmetic is used to relax accuracy, by optimizing the bit-width. This arithmetic leads to important benefits in terms of delay, power and area. Lately, several hardware approximate operators were invented, seeking the same performance benefits. However, a fair comparison between the usage of this new class of operators and classical fixed-point arithmetic with careful truncation or rounding, has never been performed. In [31], we first compare approximate and fixed-point arithmetic operators in terms of power, area and delay, as well as in terms of induced error, using many state-of-the-art metrics and by emphasizing the issue of data sizing. To perform this analysis, we developed a design exploration framework, APXPERF, which guarantees that all operators are compared using the same operating conditions. Moreover, operators are compared in several classical real-life applications leveraging relevant metrics. In [31], we show that considering a large set of parameters, existing approximate adders and multipliers tend to be dominated by truncated or rounded fixed-point ones. For a given accuracy level and when considering the whole computation data-path, fixed-point operators are several orders of magnitude more accurate while spending less energy to execute the application. A conclusion of this study is that the entropy of careful sizing is always lower than approximate operators, since it require significantly less bits to be processed in the data-path and stored. Approximated data therefore always contain on average a greater amount of costly erroneous, useless information.

7.2.4. Real-Time Scheduling of Reconfigurable Battery-Powered Multi-Core Platforms

Participants: Daniel Chillet, Aymen Gammoudi.

Reconfigurable real-time embedded systems are constantly increasingly used in applications like autonomous robots or sensor networks. Since they are powered by batteries, these systems have to be energy-aware, to adapt to their environment and to satisfy real-time constraints. For energy harvesting systems, regular recharges of battery can be estimated, and by including this parameter in the operating system, it is then possible to develop strategy able to ensure the best execution of the application until the next recharge. In this context, operating system services must control the execution of tasks to meet the application constraints. Our objective concerns the proposition of a new real-time scheduling strategy that considers execution constraints such as the deadline of tasks and the energy.

To address this issue, we first focus on mono-processor scheduling [38] and propose to classify the tasks that have similar periods (or WCETs) in packs and to manage the execution parameters of these packs. For each reconfiguration scenario, parameter modifications are performed on packs/tasks to meet the real-time and energy constraints. Compared to previous work, task delaying is significantly improved in [36]. Furthermore, we also develop a strategy for multi-cores systems considering the dependencies between tasks [37] by adding the cost of communication between cores.

7.2.5. Optimization of loop kernels using software and memory information

Participant: Angeliki Kritikakou.

Current compilers cannot generate code that can compete with hand-tuned code in efficiency, even for a simple kernel like matrix–matrix multiplication (MMM). A key step in program optimization is the estimation of optimal values for parameters such as tile sizes and number of levels of tiling. The scheduling parameter values selection is a very difficult and time-consuming task, since parameter values depend on each other; this is why they are found by using searching methods and empirical techniques. To overcome this problem, the scheduling sub-problems must be optimized together, as one problem and not separately. In [24], an MMM methodology is presented where the optimum scheduling parameters are found by decreasing the search space theoretically, while the major scheduling sub-problems are addressed together as one problem and not separately according to the hardware architecture parameters and input size; for different hardware architecture parameters and/or input sizes, a different implementation is produced. This is achieved by fully exploiting the software characteristics (e.g., data reuse) and hardware architecture parameters (e.g., data caches sizes and associativities), giving high-quality solutions and a smaller search space. This methodology refers to a wide range of CPU and GPU architectures.

The size required to store an array is crucial for an embedded system, as it affects the memory size, the energy per memory access and the overall system cost. Existing techniques for finding the minimum number of resources required to store an array are less efficient for codes with large loops and not regularly occurring memory accesses. They have to approximate the accessed parts of the array leading to overestimation of the required resources. Otherwise their exploration time is increased with an increase over the number of the different accessed parts of the array. In [25], we propose a methodology to compute the minimum resources required for storing an array which keeps the exploration time low and provides a near-optimal result for regularly and non-regularly occurring memory accesses and overlapping writes and reads.

7.2.6. Adaptive Software Control to Increase Resource Utilization in Mixed-Critical Systems

Participant: Angeliki Kritikakou.

Automotive embedded systems need to cope with antagonist requirements: on the one hand, the users and market pressure push car manufacturers to integrate more and more services that go far beyond the control of the car itself. On the other hand, recent standardization efforts in the safety domain has led to the development of the ISO 26262 norm that defines means and requirements to ensure the safe operation of automotive embedded systems. In particular, it led to the definition of ASIL (Automotive Safety and Integrity Levels), i.e., it formally defines several criticality levels. Handling the increased complexity of new services makes new architectures, such as multi or many-cores, appealing choices for the car industry. Yet, these architectures provide a very low level of timing predictability due to shared resources, which goes in contradiction with timing guarantees required by ISO 26262. For highest criticality level tasks, Worst-Case Execution Time analysis (WCET) is required to guarantee that timing constraints are respected. The WCET analyzers consider the worst-case scenario: whenever a critical task accesses a shared resource in a multi/many-core platform, a WCET analyzer considers that all cores use the same resource concurrently. To improve the system performance, we proposed in a earlier work an approach where a critical task can be run in parallel with less critical tasks, as long as the real-time constraints are met. When no further interferences can be tolerated, the proposed run-time control in [54] suspends the low critical tasks until the termination of the critical task. In an automotive context, the approach can be translated as a highly critical partition, namely a classic AUTOSAR one, that runs on one dedicated core, with several cores running less critical Adaptive AUTOSAR application(s). We briefly describe in [54] the design of our proven-correct approach. Our strategy is based on a graph grammar to formally model the critical task as a set of control flow graphs on which a safe partial WCET analysis is applied and used at run-time to control the safe execution of the critical task.

CAMUS Team

7. New Results

7.1. Formal Proofs about Happens-before in Explicitly Parallel Polyhedral Programs

Participants: Éric Violard, Alain Ketterlin.

Automatic parallelization has traditionally focused on sequential programs, but the widespread availability of explicitly parallel programming languages (such as OpenMP, Cilk, X10, and others) has led researchers to consider also the optimization and re-parallelization of parallel source programs. Most of these languages have constructions for parallel loops and parallel sections, with the accompanying synchronization primitives. The X10 language is especially interesting in this respect, because it provides simple and powerful constructions. Essentially, parallelism is expressed with the help of the `async` construct, whose body is to be executed in a parallel "activity", and the `finish` construct, which acts as a container for activities (and sub-activities) and waits for their completion. These constructions are complemented with "clocks", which are essentially synchronization barriers. Clocks can be used freely, in an unstructured manner, but are best associated with `finish` constructs, where they provide an intuitive and flexible phasing mechanism. In this case, activities spawned with `async` can either inherit or hide the clock provided by the nearest enclosing `finish`.

We are focusing on polyhedral programs, where all control is based on loops whose bounds are affine combinations of the enclosing loop counters and constant parameters. There is a large body of work on optimizing and parallelizing such programs, but most of them focus on sequential loop nests. Introducing X10's parallel constructions defines the class of *explicitly parallel polyhedral programs*, which is the focus of our work. Many polyhedral analyses and optimization techniques rely on the notion of lexicographic order, which is the order of execution of the statements in the source program. For instance, a data-dependence is defined to be an ordered pair of instruction instances that use or define the same data element, such that the first executes *before* the second. The lexicographic order is a purely syntactic characteristic that can be extracted from the source program. When the source program is explicitly parallel, the execution order becomes partial, because two distinct instruction instances can be part of concurrent activities. In this case the ordering is called the *Happens-before* relation. Paul Feautrier and Tomofumi Yuki have provided the first definition of *Happens-before* for explicitly parallel polyhedral programs, which covers the case of X10 programs using `finish` and `async` but without any clock involved. Being purely syntactic, their definition opens the way to the optimization of parallel X10 `finish-async` polyhedral programs. The use of clocks, however, introduces a major difficulty. Since clocks define phases of the program, one would like to use the "phase-number" of each instruction instance as an additional dimension, and include this dimension in further analysis. Phase-numbers have analytic forms (for the class of polyhedral programs), but they belong to the class of Ehrhart's quasi-polynomials, i.e., they are outside the polyhedral (affine) model.

We have formalized the class of programs under consideration, as well as all notions pertaining to the definition of the *Happens-before* relation, in Coq, a proof assistant developed at Inria. The formalization includes minimal structures to represent explicitly parallel polyhedral programs, including `finish` and `async`, loops, and simple statements. The definition of the *Happens-before* relation is that of an inductive predicate, parametrized by the computation of phase-numbers, which is left unspecified. To make the connection between the (static) *Happens-before* relation and the (dynamic) position of instruction instances in program traces, we use a single axiom. To reinforce our confidence in this arbitrary component, we also provide a second set of axioms, which we prove is equivalent to the first. The proof is based on an operational semantics, providing the relation between programs and their executions traces. We then prove that when *Happens-before* holds between two (static) instruction instances, then any trace of the program sees the corresponding dynamic instances ordered. We also prove the converse, which makes the definition of *Happens-before* sound and complete.

The Coq source files are kept in an Inria-forge project. Since this is our first effort in formal proofs, it currently amounts to about ten thousands lines of Coq source code. It is not yet clear whether we will publish the proof by itself, or publish an informal version of it as part of our colleagues' work on the use of *Happens-before*. In any case, our short-term plan is to extend the formalization and accompanying theorems and proofs to the case of mixed-programs, where some activities ignore the clock in scope.

7.2. Loop Nests and Integer Polyhedra

Participant: Alain Ketterlin.

The polyhedral model has been found adequate to model a large number of program analyses and transformations. It has now been used for decades in automatic parallelization, locality optimization, high-level code synthesis, and other applications. Thanks to the availability of high-quality software tools, the polyhedral model is now widely used. However, we feel that some of its most fundamental operations need more thorough attention, and possibly new theoretical developments. Even though the translation of loop nests into polyhedra (or unions thereof) obviously use integers only, many algorithms still use an underlying rational (or real) domain. For instance, Fourier-Motzkin variable elimination is defined on rational domains, and its modern incarnation (the Omega test), uses convoluted and costly techniques to compensate for the mishandling of integer variables. When used for projection (for instance during code generation, i.e., turning polyhedra into loop nests), these defects lead to sub-optimal results, with programs including more control than necessary. Overall, we feel that current techniques are inadequate to capture the precise behavior of integer variables.

We have started investigating new representations for inequalities over integer variables, using a notation called "periodic numbers". This notation was invented by Eugène Ehrhart in his classical results on the number of integer points inside integer polyhedra, and rediscovered and generalized by Philippe Clauss in his work on the use of counting for locality optimization and automatic parallelization. Periodic numbers capture all sorts of integer-specific behaviors: for instance, they are especially suitable to represent the seemingly chaotic structure of discrete line intersections, or the modular intersections of parallel hyper-planes. Periodic numbers also have algebraic properties that make them easy to manipulate and combine. We have defined a generalization of affine expressions where the constant term becomes a periodic number: it turns out that this family of expressions has interesting stability properties, that make them especially suitable for variable elimination. We have shown that most problems of Fourier-Motzkin variable elimination are related to the "looseness" of affine inequalities over integer variables, and that periodic numbers can correct this defect. The result is a new representation of inequalities, that makes reasoning with inequalities sound and complete.

An immediate application of our new representation is deciding whether a given integer polyhedron contains an integer point (or: whether a given set of affine constraints on integer variables is feasible). We have developed a straightforward version of Fourier-Motzkin elimination that is always exact. An interesting aspect of this work is that the algorithm is only a slight generalization of the original Fourier-Motzkin elimination, to cover the cases where inequalities have periodic components. We have also extended the basic algorithm to produce arbitrary projections of integer polyhedra. This improves over the Omega elimination strategy in that we are able to produce a provably disjoint union. These interesting properties derive directly from the use of periodic numbers.

Periodic numbers, and periodic linear inequalities, also have applications more directly related to the compilation of affine loop nests. For instance, we have developed a fully-general unswitching transformation. Unswitching a loop containing a conditional amounts to split the loop into one or more new loops such that the conditional has a constant truth value in all loop fragments, and can therefore be removed. The transformation is general in that the resulting program contains only affine loops and periodic linear conditionals. This means that the process can be repeated until obtaining a final version of the loop nest that is completely free of conditionals. We expect this "code generation" strategy, though naive, to remove enough "divergence" to increase existing and enable new applications of vectorization, leading to more efficient code. On the theory side, producing a conditional-free code scanning an arbitrary union of polyhedra has also direct consequences on various polyhedral operations: for instance, computing extrema becomes a trivial task, and linear optimization also falls under this umbrella. We hope to be able to explore these tracks in the near future.

We have developed software making use (and illustrating) our theoretical developments. We expect to share this software with select colleagues very soon, so as to be able to assess the scope of our techniques. Publication of these results is expected in the next year, time permitting. We also expect to extend our current software base to provide a range of integer polyhedra operations (images and pre-images, projection, and linear optimization, mostly). Finally, our middle-term goal is to investigate a formal modeling of the integer polyhedra operations. All algorithms have been kept as simple as possible, favoring elaborate abstractions over complex processing, with the goal of being able to formally specify the fundamental operations.

7.3. Splitting Polyhedra to Generate More Efficient Code

Participants: Harenome Ranaivoarivony-Razanajato, Vincent Loechner, Cédric Bastoul.

Code generation in the polyhedral model takes as input a union of Z-polyhedra and produces a code scanning all of them. Modern code generation tools are heavily relying on polyhedral operations to perform this task. However, these operations are typically provided by general-purpose polyhedral libraries that are not specifically designed to address the code generation problem. In particular, (unions of) polyhedra may be represented in various mathematically equivalent ways which may have different properties with respect to code generation. We investigated this problem and tried to find the best representation of polyhedra to generate an efficient code.

We demonstrated that this problem has been largely under-estimated, showing significant control overhead deviations when using different representations of the same polyhedra. Second, we proposed an improvement to the main algorithm of the state-of-the-art code generation tool CLooG. It generates code with less tests in the inner loops, and aims at reducing control overhead and at simplifying vectorization for the compiler, at the cost of a larger code size. It is based on a smart splitting of the union of polyhedra while recursing on the dimensions.

We implemented our algorithm in CLooG/PolyLib, and compared the performance and size of the generated code to the CLooG/isl version. Our results show that there can be important performance differences between the generated versions. In some cases, our new technique may significantly improve the quality of the generated code, but in some other cases, it may not be adequate compared to the existing solution. Finding other alternatives and chosing the best one remain open problems to be investigated in the future.

7.4. Code-Bones for Fast and Flexible Runtime Code Generation

Participants: Juan Manuel Martinez Caamaño, Artiom Baloian, Philippe Clauss.

We have developed a new runtime code generation technique for speculative loop optimization and parallelization. The main benefit of this technique, compared to previous approaches, is to enable advanced optimizing loop transformations at runtime with an acceptable time overhead. The loop transformations that may be applied are those handled by the polyhedral model. The proposed code generation strategy is based on the generation of *code-bones* at compile-time, which are parametrized code snippets either dedicated to speculation management or to computations of the original target program. These code bones are then instantiated and assembled at runtime to constitute the speculatively-optimized code, as soon as an optimizing polyhedral transformation has been determined. Their granularity threshold is sufficient to apply any polyhedral transformation, while still enabling fast runtime code generation. This approach has been implemented in the speculative loop parallelizing framework Apollo, and published at the conference Euro-Par 2016 where it has been selected as best paper [13]. An extended journal version is currently under review. This is also the main contribution of Juan Manuel Martinez Caamaño's PhD thesis which was defended in September 2016 [8].

7.5. Automatic Collapsing of Non-Rectangular Loops

Participants: Philippe Clauss, Ervin Altıntaş, Matthieu Kuhn.

Loop collapsing is a well-known loop transformation which combines some loops that are perfectly nested into one single loop. It allows to take advantage of the whole amount of parallelism exhibited by the collapsed loops, and provides a perfect load balancing of iterations among the parallel threads.

However, in the current implementations of this loop optimization, as the ones of the OpenMP language, automatic loop collapsing is limited to loops with constant loop bounds that define rectangular iteration spaces, although load imbalance is a particularly crucial issue with non-rectangular loops. The OpenMP language addresses load balance mostly through dynamic runtime scheduling of the parallel threads. Nevertheless, this runtime schedule introduces some unavoidable execution-time overhead, while preventing to exploit the entire parallelism of all the parallel loops.

We propose a technique to automatically collapse any perfectly nested loops defining non-rectangular iteration spaces, whose bounds are linear functions of the loop iterators. Such spaces may be triangular, tetrahedral, trapezoidal, rhomboidal or parallelepiped. Our solution is based on original mathematical results addressing the inversion of a multi-variate polynomial that defines a ranking of the integer points contained in a convex polyhedron.

We show on a set of non-rectangular loop nests that our technique allows to generate parallel OpenMP codes that outperform the original parallel loop nests, parallelized either by using options “static” or “dynamic” of the OpenMP-schedule clause. A conference paper presenting these results, co-authored by Philippe Clauss, Ervin Altıntaş (Master student) and Matthieu Kuhn (Inria Bordeaux Sud-Ouest, team HIEPACS), is currently under review.

7.6. Efficient Data Structures for a PIC Code on SIMD Architectures

Participants: Yann Barsamian, Éric Violard.

In collaboration with Sever Adrian Hirstoaga (mathematician researcher, member of Inria team TONUS), we have developed an efficient particle simulation code. The domain of application is plasma physics, the Particle-In-Cell code simulating 2d2v Vlasov-Poisson equation on Cartesian grid with periodic boundary conditions for Landau damping test-case. We first analyzed different strategies for improving its performance on single core and then we used a standard approach for parallelizing it on many cores using hybrid OpenMP/MPI implementation. The optimization of the sequential code is mainly based on (i) a structure of arrays for the particles, (ii) an efficient data structure for the electric field and the charge density, and (iii) an appropriate code for automatic vectorization of the charge accumulation and of the positions' update. The parallelization of the loops over the particles is performed in a simple way (without domain decomposition) by means of both distributed and share memory paradigms. Satisfactory strong and weak scaling up to 8,192 cores on GENCI's supercomputer Curie are obtained, bounded as expected by the overhead of MPI communications. A conference paper presenting this work is currently under review.

7.7. Interactive Code Restructuring

Participants: Cédric Bastoul, Oleksandr Zinenko, Stéphane Huot.

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 new research direction opened by CAMUS which strengthens the team's static parallelization and optimization issue. It is a joint work with two Inria teams specialized in interaction: EX-SITU at Inria Saclay (contact: Oleksandr Zinenko) and MJOLNIR at Inria Lille (contact: Stéphane Huot).

In 2016, we released the first version of our interactive tool, *Clint*, that maps direct manipulation of the visual representation to polyhedral program transformations with real-time semantics preservation feedback (https://ozinenko.com/clint). Oleksandr Zinenko also defended his thesis on the research and development on interactive code restructuring.

7.8. Automatic Generation of Adaptive Simulation Codes

Participants: Cédric Bastoul, Maxime Schmitt.

Compiler automatic optimization and parallelization techniques are well suited for some classes of simulation or signal processing applications, however they usually don't take into account neither domain-specific knowledge nor the possibility to change or to remove some computations to achieve "good enough" results. Quite differently, production simulation and signal processing codes have adaptive capabilities: they are designed to compute precise results only where it matters if the complete problem is not tractable or if the computation time must be short. In this research, we design a new way to provide adaptive capabilities to compute-intensive codes automatically, inspired by Adaptive Mesh Refinement a classical numerical analysis technique to achieve precise computation only in pertinent areas. It relies on domain-specific knowledge provided through special pragmas by the programmer in the input code and on polyhedral compilation techniques, to continuously regenerate at runtime a code that performs heavy computations only where it matters at every moment. A case study on a fluid simulation application shows that our strategy enables dramatic computation savings in the optimized portion of the application while maintaining good precision, with a minimal effort from the programmer.

This research direction started in 2015 and complements our other efforts on dynamic optimization. In 2016, we started a collaboration on this topic with Inria Nancy - Grand Est team TONUS, specialized on applied mathematics (contact: Philippe Helluy), to bring models and techniques from this field to compilers. This collaboration received the support from the excellence laboratory (LabEx) IRMIA through the funding of the thesis of Maxime Schmitt on this topic. A first paper on this new research direction has just been accepted to IMPACT 2017.

7.9. Polyhedral Compiler White-Boxing

Participants: Cédric Bastoul, Lénaïc Bagnères, Oleksandr Zinenko, Stéphane Huot.

While compilers offer a fair trade-off between productivity and executable performance in single-threaded execution, their optimizations remain fragile when addressing compute-intensive code for parallel architectures with deep memory hierarchies. Moreover, these optimizations operate as black boxes, impenetrable for the user, leaving them with no alternative to time-consuming and error-prone manual optimization in cases where an imprecise cost model or a weak analysis resulted in a bad optimization decision. To address this issue, we researched and designed a technique allowing to automatically translate an arbitrary polyhedral optimization, used internally by loop-level optimization frameworks of several modern compilers, into a sequence of comprehensible syntactic transformations as long as this optimization focuses on scheduling loop iterations. With our approach, we open the black box of the polyhedral frameworks enabling users to examine, refine, replay and even design complex optimizations semi-automatically in partnership with the compiler.

This research started in 2014 and we published our first solution in 2016. It has been conducted as a joint work between teams in compiler technologies (CAMUS and Inria Saclay's POSTALE team) and teams in interaction (EX-SITU at Inria Saclay and MJOLNIR at Inria Lille). The first paper on this has been accepted and presented in one of the top conferences on optimization techniques: CGO 2016 [10]. It is also discussed in Lénaïc Bagnèse and Oleksandr Zinenko theses. In 2016 we finally release the tool implementing this research (https://periscop.github.io/chlore).

7.10. Mapping Deviation

Participant: Cédric Bastoul.

Compilers can provide a major help by automating the optimization and parallelization work. However they are very fragile black-boxes. A compiler may take a bad optimization decision because of imprecise heuristics or may turn off an optimization because of imprecise analyses, without providing much control or feedback to the end user. To address this issue, we researched and introduced mapping deviation, a new compiler technique that aims at providing a useful feedback on the semantics of a given program restructuring. Starting from a transformation intuition a user or a compiler wants to apply, our algorithm studies its correctness and can suggest changes or conditions to make it possible rather than being limited to the classical go/no-go answer. This algorithm builds on state-of-the-art polyhedral representation of programs and provides a high flexibility. We present two example applications of this technique: improving semi-automatic optimization tools for programmers and automatically designing runtime tests to check the correctness of a transformation for compilers.

This is a mid-term research on the mathematical ground of polyhedral compilation, started back to 2012. We found a solution and published it in 2016 in one of the main conferences in compilation: Compiler Construction [11]. We plan to release the tool that implements this research during the coming year.

7.11. Combining Locking and Data Management Interfaces

Participants: Jens Gustedt, Mariem Saied, Daniel Salas.

Handling data consistency in parallel and distributed settings is a challenging task, in particular if we want to allow for an easy to handle asynchronism between tasks. Our publication [1] shows how to produce deadlock-free iterative programs that implement strong overlapping between communication, IO and computation.

An implementation (ORWL) of our ideas of combining control and data management in C has been undertaken, see 6.8 . In previous work it has demonstrated its efficiency for a large variety of platforms. In 2016, work on the ORWL model and library has gained vigor with the thesis of Mariem Saied (Inria & University of Strasbourg) and Daniel Salas (INSERM). We also now collaborate on that subject with the TADAAM project team from Inria Bordeaux, where a postdoc has been hired through Inria funding.

In 2016, a new domain specific language (DSL) has been completed that largely eases the implementation of applications with ORWL. In its first version it provides an interface for stencil codes, but extensions towards other types of applications are on their way. The approach allows to describe stencil codes quickly and efficient and leads to substantial speedups, see [14].

In addition, the framework has successfully been applied to encapsulate imaging applications that use certain pipeline patterns to describe dependencies between computational tasks, see [16]. Generally we have been able to use the knowledge of the communication structure of ORWL programs to map tasks to cores and thereby achieve interesting performance gains on multicore architectures, see [20].

In another work we have successfully applied ORWL to process Large Histopathology Images, see [15].

COMPSYS Team

7. New Results

7.1. Handling Polynomials for Program Analysis and Transformation

Participant: Paul Feautrier.

As is well known in natural language processing, the first step in translating a text from one language to another is to understand it. The situation is the same for formal languages. A language processor has to "understand" a program before translating or optimizing or verifying it. Such understanding takes the form of a *model*, usually a mathematical representation whose natural operations mimic the behavior of its program. The polyhedral model is such a representation. However, the set of programs it can represent is too restricted, and the hunt for more powerful models has been under way since the millennium.

An obvious ideas is to replace affine formulas by polynomials, and hence polyhedra by semi-algebraic sets. Polynomials are ubiquitous in HPC and embedded system programming. For instance, the 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. It may also happen that polynomials are absent from the source program, but are created either by an enabling analysis, as for OpenStream (see Section 7.2), or are imposed by complexity consideration. Lastly, polynomials may be native to the underlying algorithm, as when distances are computed by the usual Euclidean formula. 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 – was presented at (IMPACT'15) [14]. A tool to manipulate polynomials, polynomial constraints and objective functions, needed for the derivation of polynomial schedules, complements this work (see Section 6.2). It implements Farkas lemma and its generalization with Handelman & Schweighofer formulations, and is in constant development, including improvements on the objective functions, in particular to make schedule selection more stable, independently on the degree of the polynomial schedule.

7.2. Static Analysis of OpenStream Programs

Participants: Albert Cohen [Inria Parkas team], Alain Darte, Paul Feautrier.

In the context of the ManycoreLabs project, we started to study the applicability of polyhedral techniques to the parallel language OpenStream [19]. When applicable, polyhedral techniques are indeed invaluable for compile-time debugging and for generating efficient code well suited to a target architecture. OpenStream is a two-level language in which a control program directs the initialization of parallel task instances that communicate through *streams*, with possibly multiple writers and readers. It has a fairly complex semantics in its most general setting, but we restricted ourselves to the case where the control program is sequential, which is representative of the majority of the OpenStream applications.

In contrast to the language X10, which we studied in previous years, this restriction offers deterministic concurrency by construction, but deadlocks are still possible. We showed that, if the control program is polyhedral, one may statically compute, for each task instance, the read and write indices to each of its streams, and thus reason statically about the dependences among task instances (the only scheduling constraints in this polyhedral subset). If the control program has nested loops, communications use one-dimensional channels in a form of linearization, and these indices may be polynomials of arbitrary degree, thus requiring to extend to polynomials the standard polyhedral techniques for dependence analysis, scheduling, and deadlock detection. Modern SMT allow to solve polynomial problems, albeit with no guarantee of success; the approach previously developed by P. Feautrier [14], and recalled in Section 7.1 , offers an alternative solution.

The 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, we established two important results related to deadlocks: 1) a characterization of deadlocks in terms of dependence paths, which implies that streams can be safely bounded as soon as a schedule exists with such sizes, 2) the proof that deadlock detection is undecidable, even for polyhedral OpenStream. Details of this work have been published at the international workshop IMPACT'16 [1].

Some further developments are in progress for scheduling OpenStream programs using polynomial techniques (with a corresponding prototype scheduling tool, specific to OpenStream, see Section 6.3). In particular, we made some progress for parsing a simplified version of OpenStream, exhibiting the relevant structure, and on the properties and construction of schedules with bounded streams and bounded delays, and on the analysis of the "foot bath", i.e., the pool of tasks that are created (already requiring some resources) but not activated yet (because they need to wait for the termination of other tasks due to dataflow semantics). This work should have interesting connections with the way runtime systems of tasks are managed.

7.3. Liveness Analysis in Explicitly-Parallel Programs

Participants: Alain Darte, Alexandre Isoard, Tomofumi Yuki.

In the light of the parallel specifications encountered in our other work – kernel offloading with pipelined communications [10], automatic parallelization, analysis of X10 [22], [23] and of OpenStream (see Section 7.2), intra-array reuse (see Section 7.4) – we revisited scalar and array element-wise liveness analysis for programs with parallel specifications. In earlier work on memory allocation/contraction (register allocation or intra- and inter-array reuse in the polyhedral model), a notion of "time" or a total order among the iteration points was used to compute the liveness of values. In general, the execution of parallel programs is not a total order, and hence the notion of time is not applicable.

We first revised how conflicts are computed by using ideas from liveness analysis for register allocation, studying the structure of the corresponding conflict/interference graphs. Instead of considering the conflict between two pairs of live ranges, we only consider the conflict between a live range and a write. This simplifies the formulation from having four instances involved in the test down to three, and also improves the precision of the analysis in the general case. Then we extended the liveness analysis to work with partial orders so that it can be applied to many different parallel languages/specifications with different forms of parallelism. An important result is that the complement of the conflict graph with partial orders is directly connected to memory reuse, even in presence of races. However, programs with conditionals do not even have a partial order, and our next step will be to handle such cases with more accuracy. Details of this work have been published at the international workshop IMPACT'16 [3].

Some new developments are in progress to explore even further the properties of such liveness analysis and the construction of conflict sets, in the general case (with connections with the concept of trace monoid) or for some common situations such as series-parallel graphs, appearing in languages such as X10 or OpenMP.

7.4. Extended Lattice-Based Memory Allocation

Participants: Alain Darte, Alexandre Isoard, Tomofumi Yuki.

We extended lattice-based memory allocation [11], an earlier work on memory (array) reuse analysis. The main motivation is to handle in a better way the more general forms of specifications we see today, e.g., with loop tiling, pipelining, and other forms of parallelism available in explicitly parallel languages. Our extension has two complementary aspects. We showed how to handle more general specifications where conflicting constraints (those that describe the array indices that cannot share the same location) are specified as a (non-convex) union of polyhedra. Unlike convex specifications, this also requires to be able to choose suitable directions (or basis) of array reuse. For that, we extended two dual approaches, previously proposed for a fixed basis, into optimization schemes to select suitable basis. Our final approach relies on a combination of the two, also revealing their links with, on one hand, the construction of multi-dimensional schedules for parallelism and tiling (but with a fundamental difference that we identify) and, on the other hand, the construction of

universal reuse vectors (UOV), which was only used so far in a specific context, for schedule-independent mapping.

This algorithmic work, connected to our previous work on parametric tiling [10] and the liveness analysis results of Section 7.3 , is complemented by a set of prototype scripting tools, see Section 6.1 . Details of this work have been published at the 2016 International Conference on Compiler Construction [2].

7.5. Stencil Accelerators

Participants: Steven Derrien [University of Rennes 1, Inria/CAIRN], Sanjay Rajopadhye [Colorado State University], Tomofumi Yuki.

Stencil computations have been known to be an important class of programs for scientific calculations. Recently, various architectures (mostly targeting FPGAs) for stencils are being proposed as hardware accelerators with high throughput and/or high energy efficiency. There are many different challenges for such design: How to maximize compute-I/O ratio? How to partition the problem so that the data fits on the on-chip memory? How to efficiently pipeline? How to control the area usage? We seek to address these challenges by combining techniques from compilers and high-level synthesis tools.

One project in collaboration with the CAIRN team and Colorado State University targets stencils with regular dependence patterns. Although many architectures have been proposed for this type of stencils, most of them use a large number of small processing elements (PE) to achieve high throughput. We are exploring an alternative design that aims for a single, large, deeply-pipelined PE. The hypothesis is that the pipelined parallelism is more area-efficient compared to replicating small PEs. We have published a work-in-progress paper on this topic at IMPACT'16 [4].

7.6. Efficient Mapping of Irregular Memory Accesses on FPGA

Participants: Xinyu Niu [Imperial College London], Tomofumi Yuki.

In a collaboration with Imperial College, we looked at efficiently implementing dynamic dependences on FPGAs. The collaboration is in the context of the EURECA project [0] where the dynamic reconfigurability of modern FPGAs is used to efficiently handle dynamic access patterns. We worked on analyzing data dependent array accesses to identify regularities within irregular memory accesses to reduce the cost of a dynamic memory reconfiguration module.

One part of this work has been published at the 2016 International Conference on Field Programmable Logic and Applications [5].

7.7. PolyApps

Participant: Tomofumi Yuki.

Loop transformation frameworks using the polyhedral model have gained increased attention since the rise of the multi-core era. We now have several research tools that have demonstrated their power on important kernels found in scientific computations. However, there remains a large gap between the typical kernels used to evaluate these tools and the actual applications used by the scientists.

PolyApps is an effort to collect applications from other domains of science to better establish the link between the compiler tools and "real" applications. The applications are modified to bypass some of the front-end issues of research tools, while keeping the ability to produce the original output. The goal is to assess how the state-of-the-art automatic parallelizers perform on full applications, and to identify new opportunities that only arise in larger pieces of code.

We showed that, with a few enhancements, the current tools will be able to reach and/or exceed the performance of existing parallelizations of the applications. One of the most critical element missing in current tools is the ability to modify the memory mappings.

[0] http://www.doc.ic.ac.uk/~nx210/2015/09/01/eureca.html

CORSE Project-Team

6. New Results

6.1. Simplification and Run-time Resolution of Data Dependence Constraints for Loop Transformations

Participants: Diogo Nunes Sampaio, Alain Ketterlin [Inria CAMUS], Louis-Noël Pouchet [CSU, USA], Fabrice Rastello.

Loop optimizations such as tiling, thread-level parallelization or vectorization are essential transformations to improve performance. It is needed to compute dependence information at compile-time to assess their validity, but in many real situations, static dependence analysis fails to provide precise enough information. Part of the reason for this failure comes from the need to handle polynomial constraints in the dependence computation problem: such polynomial constraints can arise from linearized array accesses, typical in compilers IR such as LLVM-IR. In this scenario, the compiler will often be unable to apply aggressive transformations due to lack of conclusive static dependence analysis. This work tackles the problem of eliminating quantifiers in systems of inequalities using polynomial constraints. In particular, we design a quantifier elimination scheme on integer multivariate-polynomials, which can aid application of off-the-shelf polyhedral transformations on a larger class of programs, that holds polynomial memory access and affine loop bounds. We make a significant leap in accuracy compared to prior approaches, enabling to implement a hybrid optimizing compilation scheme. In this scheme, a test is evaluated at run-time to determine the legality of the program transformation chosen by the compiler, falling back to executing the original code if the test fails. This test integrates all may-dependences, involving polynomial inequalities, and is simplified by quantifier elimination at compile-time using our techniques. The preciseness of the presented scheme and the low run-time overhead of the test are key to make this approach realistic. We experimentally validate our technique on 25 benchmarks using complex loop transformations, achieving negligible overhead. Preciseness is assessed by the observed success of generated test in practical cases. We compare our variable elimination technique to other existing tools and demonstrate we achieve better precision when dealing with polynomial memory accesses.

This work is the fruit of the collaboration 8.4 with OSU.

6.2. A bounded memory allocator for software-defined global address spaces

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

This work is about the design of a memory allocator targeting manycore architectures with distributed memory. Among the family of Multi Processor System on Chip (MPSoC), these devices are composed of multiple nodes linked by an on-chip network; most nodes have multiple processors sharing a small local memory. While MPSoC typically excel on their performance-per-Watt ratio, they remain hard to program due to multilevel parallelism, explicit resource and memory management, and hardware constraints (limited memory, network topology).

Typical programming frameworks for MPSoC leave much target-specific work to the programmer: combining threads or node-local OpenMP, software caching, explicit message passing (and sometimes, routing), with non-standard interfaces. More abstract, automatic frameworks exist, but they target large-scale clusters and do not model the hardware constraints of MPSoC.

This memory allocator is one component of a larger runtime system, called Givy 5.3 , to support dynamic task graphs with automatic software caching and data-driven execution on MPSoC. To simplify the programmer's view of memory, both runtime and program data objects live in a Global Address Space (GAS). To avoid address collisions when objects are dynamically allocated, and to manage virtual memory mappings across nodes, a GAS-aware memory allocator is required. This work proposes such an allocator with the following properties: (1) it is free of inter-node synchronizations; (2) its node-local performance match that of state-of-the-art shared-memory allocators; (3) it provides node-local mechanisms to implement inter-node software caching within a GAS; (4) it is well suited for small memory systems (a few MB per node).

This work has been presented at the international conference ISMM 2016 [16].

6.3. On Fusing Recursive Traversals of K-d Trees

Participants: Samyam Rajbhandari [OSU, USA], Jinsung Kim [OSU, USA], Sriram Krishnamoorthy [PNNL, USA], Louis-Noel Pouchet [CSU, USA], Fabrice Rastello, Robert J. Harrison [Stony Brook, USA], P. Sadayappan [OSU, USA].

Loop fusion is a key program transformation for data locality optimization that is implemented in production compilers. But optimizing compilers for imperative languages currently cannot exploit fusion opportunities across a set of recursive tree traversal computations with producer-consumer relationships. In this work, we develop a compile-time approach to dependence characterization and program transformation to enable fusion across recursively specified traversals over k-d trees. We present the FuseT source-to-source code transformation framework to automatically generate fused composite recursive operators from an input program containing a sequence of primitive recursive operators. We use our framework to implement fused operators for MADNESS, Multiresolution Adaptive Numerical Environment for Scientific Simulation. We show that locality optimization through fusion can offer significant performance improvement.

This work is the fruit of the collaboration 8.4 with OSU. The specific work on FuseT has been presented to the international conference CC 2016 [32] and the more general work on the improvement of MADNESS at the ACM/IEEE international conference SC 2016 [20].

6.4. Effective Padding of Multidimensional Arrays to Avoid Cache Conflict Misses

Participants: Changwan Hong [OSU, USA], Wenlei Bao [OSU, USA], Albert Cohen [Inria PARKAS], Sriram Krishnamoorthy [PNNL, USA], Louis-Noel Pouchet [CSU, USA], Fabrice Rastello, J. Ramanujam [LSU, USA], P. Sadayappan [OSU, USA].

Caches are used to significantly improve performance. Even with high degrees of set associativity, the number of accessed data elements mapping to the same set in a cache can easily exceed the degree of associativity. This can cause conflict misses and lower performance, even if the working set is much smaller than cache capacity. Array padding (increasing the size of array dimensions) is a well-known optimization technique that can reduce conflict misses. In this work, we develop the first algorithms for optimal padding of arrays aimed at a set-associative cache for arbitrary tile sizes. In addition, we develop the first solution to padding for nested tiles and multi-level caches. Experimental results with multiple benchmarks demonstrate a significant performance improvement from padding.

This work is the fruit of the collaboration 8.4 with OSU. It has been presented at the ACM international conference PLDI 2016 [29].

6.5. 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 program outputs. Each technique suffers from limitations in the kind of transformations supported, space complexity, or the sensitivity to the testing dataset. In this work, we take a novel approach that addresses 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 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 fruit of the collaboration 8.4 with OSU and was presented at ACM POPL'16 [14].

6.6. Modularizing Crosscutting Concerns in Component-Based Systems

Participants: Antoine El-Hokayem, Yliès Falcone, Mohamad Jaber [American University of Beirut, Lebanon].

We define a method to modularize crosscutting concerns in the Behavior Interaction Priority (BIP) component-based framework. Our method is inspired from the Aspect Oriented Programming (AOP) paradigm which was initially conceived to support the separation of concerns during the development of monolithic systems. BIP has a formal operational semantics and makes a clear separation between architecture and behavior to allow for compositional and incremental design and analysis of systems. We thus distinguish local from global aspects. Local aspects model concerns at the component level and are used to refine the behavior of components. Global aspects model concerns at the architecture level, and hence refine communications (synchronization and data transfer) between components. We formalize global aspects as well as their integration into a BIP system through rigorous transformation primitives and overview local aspects. We present AOP-BIP, a tool for Aspect-Oriented Programming of BIP systems, and demonstrate its use to modularize logging, security, and fault-tolerance in a network protocol.

This work results of the collaboration with American University of Beirut (Lebanon) and was presented at SEFM 2016 [15].

6.7. Predictive runtime enforcement

Participants: Srinivas Pinisetty [Aalto University, Finland], Viorel Preoteasa [Aalto University, Finland], Stavros Tripakis [Aalto University, Finland], Thierry Jéron [Inria Rennes, France], Yliès Falcone, Hervé Marchand [Inria Rennes, France].

Runtime enforcement (RE) is a technique to ensure that the (untrustworthy) output of a black-box system satisfies some desired properties. In RE, the output of the running system, modeled as a stream of events, is fed into an enforcement monitor. The monitor ensures that the stream complies with a certain property, by delaying or modifying events if necessary. This work deals with predictive runtime enforcement, where the system is not entirely black-box, but we know something about its behavior. This a-priori knowledge about the system allows to output some events immediately, instead of delaying them until more events are observed, or even blocking them permanently. This in turn results in better enforcement policies. We also show that if we have no knowledge about the system, then the proposed enforcement mechanism reduces to a classical non-predictive RE framework. All our results are formalized and proved in the Isabelle theorem prover.

This work was presented at SAC-SVT 2016 [19].

6.8. Third International Competition on Runtime Verification

Participants: Giles Reger [University of Manchester, UK], Sylvain Hallé [The University of Québec at Chicoutimi, Canada], Yliès Falcone.

We report on the Third International Competition on Runtime Verification (CRV-2016). The competition was held as a satellite event of the 16th International Conference on Runtime Verification (RV'16). The competition consisted of two tracks: offline monitoring of traces and online monitoring of Java programs. The intention was to also include a track on online monitoring of C programs but there were too few participants to proceed with this track. This report describes the format of the competition, the participating teams, the submitted benchmarks and the results. We also describe our experiences with transforming trace formats from other tools into the standard format required by the competition and report on feedback gathered from current and past participants and use this to make suggestions for the future of the competition.

This work was presented at RV 2016 [13].

6.9. Monitoring Multi-threaded Component-Based Systems

Participants: Hosein Nazarpour [Verimag, France], Yliès Falcone, Saddek Bensalem [Verimag, France], Marius Bozga [Verimag, France], Jacques Combaz [Verimag, France].

This work addresses the monitoring of logic-independent linear-time user-provided properties on multi-threaded component-based systems. We consider intrinsically independent components that can be executed concurrently with a centralized coordination for multiparty interactions. In this context, the problem that arises is that a global state of the system is not available to the monitor. A naive solution to this problem would be to plug a monitor which would force the system to synchronize in order to obtain the sequence of global states at runtime. Such solution would defeat the whole purpose of having concurrent components. Instead, we reconstruct on-the-fly the global states by accumulating the partial states traversed by the system at runtime. We define formal transformations of components that preserve the semantics and the concurrency and, at the same time, allow to monitor global-state properties. Moreover, we present RVMT-BIP, a prototype tool implementing the transformations for monitoring multi-threaded systems described in the BIP (Behavior, Interaction, Priority) framework, an expressive framework for the formal construction of heterogeneous systems. Our experiments on several multi-threaded BIP systems show that RVMT-BIP induces a cheap runtime overhead.

This work was presented at iFM 2016 [18].

6.10. Decentralized Enforcement of Artifact Lifecycles

Participants: Sylvain Hallé [The University of Québec at Chicoutimi, Canada], Raphaël Khoury [The University of Québec at Chicoutimi, Canada], Antoine El-Hokayem, Yliès Falcone.

Artifact-centric workflows describe possible executions of a business process through constraints expressed from the point of view of the documents exchanged between principals. A sequence of manipulations is deemed valid as long as every document in the workflow follows its prescribed lifecycle at all steps of the process. So far, establishing that a given workflow complies with artifact lifecycles has mostly been done through static verification, or by assuming a centralized access to all artifacts where these constraints can be monitored and enforced. We propose an alternate method of enforcing document lifecycles that requires neither static verification nor single-point access. Rather, the document itself is designed to carry fragments of its history, protected from tampering using hashing and public-key encryption. Any principal involved in the process can verify at any time that a document's history complies with a given lifecycle. Moreover, the proposed system also enforces access permissions: not all actions are visible to all principals, and one can only modify and verify what one is allowed to observe.

This work was presented at EDOC 2016 [17].

6.11. Runtime enforcement of regular timed properties by suppressing and delaying events

Participants: Yliès Falcone, Thierry Jéron [Inria Rennes, France], Hervé Marchand [Inria Rennes, France], Srinivas Pinisetty [Aalto University, Finland].

Runtime enforcement is a verification/validation technique aiming at correcting possibly incorrect executions of a system of interest. In this work, we consider enforcement monitoring for systems where the physical time elapsing between actions matters. Executions are thus modelled as timed words (i.e., sequences of actions with dates). We consider runtime enforcement for timed specifications modelled as timed automata. Our enforcement mechanisms have the power of both delaying events to match timing constraints, and suppressing events when no delaying is appropriate, thus possibly allowing for longer executions. To ease their design and their correctness-proof, enforcement mechanisms are described at several levels: enforcement functions that specify the input–output behaviour in terms of transformations of timed words, constraints

that should be satisfied by such functions, enforcement monitors that describe the operational behaviour of enforcement functions, and enforcement algorithms that describe the implementation of enforcement monitors. The feasibility of enforcement monitoring for timed properties is validated by prototyping the synthesis of enforcement monitors from timed automata.

This work was published in the journal Science of Computer Programming [8].

6.12. Organising LTL monitors over distributed systems with a global clock

Participants: Christian Colombo [University of Malta, Malta], Yliès Falcone.

Users wanting to monitor distributed systems often prefer to abstract away the architecture of the system by directly specifying correctness properties on the global system behaviour. To support this abstraction, a compilation of the properties would not only involve the typical choice of monitoring algorithm, but also the organisation of submonitors across the component network. Existing approaches, considered in the context of LTL properties over distributed systems with a global clock, include the so-called orchestration and migration approaches. In the orchestration approach, a central monitor receives the events from all subsystems. In the migration approach, LTL formulae transfer themselves across subsystems to gather local information. We propose a third way of organising submonitors: choreography, where monitors are organised as a tree across the distributed system, and each child feeds intermediate results to its parent. We formalise choreography-based decentralised monitoring by showing how to synthesise a network from an LTL formula, and give a decentralised monitoring algorithm working on top of an LTL network. We prove the algorithm correct and implement it in a benchmark tool. We also report on an empirical investigation comparing these three approaches on several concerns of decentralised monitoring: the delay in reaching a verdict due to communication latency, the number and size of the messages exchanged, and the number of execution steps required to reach the verdict.

This work was published in the journal Formal Methods in System Design [6].

6.13. Decentralised LTL monitoring

Participants: Andreas Bauer [TU Munich, Software and Systems Engineering Munich, Germany], Yliès Falcone.

Users wanting to monitor distributed or component-based systems often perceive them as monolithic systems which, seen from the outside, exhibit a uniform behaviour as opposed to many components displaying many local behaviours that together constitute the system's global behaviour. This level of abstraction is often reasonable, hiding implementation details from users who may want to specify the system's global behaviour in terms of a linear-time temporal logic (LTL) formula. However, the problem that arises then is how such a specification can actually be monitored in a distributed system that has no central data collection point, where all the components' local behaviours are observable. In this case, the LTL specification needs to be decomposed into sub-formulae which, in turn, need to be distributed amongst the components' locally attached monitors, each of which sees only a distinct part of the global behaviour. The main contribution of this work is an algorithm for distributing and monitoring LTL formulae, such that satisfaction or violation of specifications can be detected by local monitors alone. We present an implementation and show that our algorithm introduces only a negligible delay in detecting satisfaction/violation of a specification. Moreover, our practical results show that the communication overhead introduced by the local monitors is generally lower than the number of messages that would need to be sent to a central data collection point. Furthermore, our experiments strengthen the argument that the algorithm performs well in a wide range of different application contexts, given by different system/communication topologies and/or system event distributions over time.

This work was published in the journal Formal Methods in System Design [4].

6.14. Using data dependencies to improve task-based scheduling strategies on NUMA architectures

Participants: Philippe Virouleau, François Broquedis, Thierry Gautier [Inria, AVALON], Fabrice Rastello.

The recent addition of data dependencies to the OpenMP 4.0 standard provides the application programmer with a more flexible way of synchronizing tasks. Using such an approach allows both the compiler and the runtime system to know exactly which data are read or written by a given task, and how these data will be used through the program lifetime. Data placement and task scheduling strategies have a significant impact on performances when considering NUMA architectures. While numerous studies focus on these topics, none of them has made extensive use of the information available through dependencies. One can use this information to modify the behavior of the application at several levels : during initialization to control data placement and during the application execution to dynamically control both the task placement and the tasks stealing strategy, depending on the topology. This work introduces several heuristics for these strategies, their implementations in the xkaapi OpenMP runtime system and the performances on linear algebra applications executed on a 192-core NUMA machine. Such approaches report noticeable performance improvement when considering both the architecture topology and the tasks data dependencies.

This work has been presented at the international conference EuroPar'2016 [22].

6.15. Description, Implementation and Evaluation of an Affinity Clause for Task Directives

Participants: Philippe Virouleau, Adrien Roussel [IFPEN], François Broquedis, Thierry Gautier [Inria, AVALON], Fabrice Rastello, Jean-Marc Gratien [IFPEN].

This work extends the affinity-based scheduling we proposed at the Europar 2016 conference to fit the philosophy of OpenMP programming. On this topic, OpenMP does not provide a lot of flexibility to the programmer yet, which lets the runtime system decide where a task should be executed. In this work, we propose our own interpretation of the new affinity clause for the task directive, which is being discussed by the OpenMP Architecture Review Board. This clause enables the programmer to give hints to the runtime about tasks placement during the program execution, which can be used to control the data mapping on the architecture. In our proposal, the programmer can express affinity between a task and the following resources: a thread, a NUMA node, and a data. We provide an implementation of this proposal in the Clang-3.8 compiler, and an implementation of the corresponding extensions in the xkaapi OpenMP runtime system.

This work has been presented at the international workshop on OpenMP IWOMP'2016 [23].

6.16. Design methodology for workload-aware loop scheduling strategies based on genetic algorithm and simulation

Participants: Pedro H. Penna [PUC Minas], Márcio Castro [UFSC], Henrique C. Freitas [PUC Minas], François Broquedis, Jean-François Méhaut.

In high-performance computing, the application's workload must be evenly balanced among threads to deliver cutting-edge performance and scalability. In OpenMP, the load balancing problem arises when scheduling loop iterations to threads. In this context, several scheduling strategies have been proposed, but they do not take into account the input workload of the application and thus turn out to be suboptimal. In this work, we introduce a design methodology to propose, study, and assess the performance of workload-aware loop scheduling strategies. In this methodology, a genetic algorithm is employed to explore the state space solution of the problem itself and to guide the design of new loop scheduling strategies, and a simulator is used to evaluate their performance. As a proof of concept, we show how the proposed methodology was used to propose and study a new workload-aware loop scheduling strategy named smart round-robin (SRR). We implemented this strategy into GNU Compiler Collection's OpenMP runtime. We carry out several experiments to validate the simulator and to evaluate the performance of SRR. Our experimental results show that SRR may deliver up to 37.89% and 14.10% better performance than OpenMP's dynamic loop scheduling strategy in the simulated environment and in a real-world application kernel, respectively.

This work is presented in the CCPE journal [9].

6.17. The Mont-Blanc prototype: An Alternative Approach for HPC Systems

Participants: Brice Videau, Kevin Pouget, Jean-François Méhaut.

The evolution of High-Performance Computing (HPC) systems is driven by the need of reducing time-to-solution and increasing the resolution of models and problems being solved by a particular program. Important milestones from the HPC system performance perspective were achieved using commodity technology. Examples are the ASCI Red and the Roadrunner supercomputers, which broke the 1 TFLOPS and 1 PFLOPS barriers, respectively. These systems showed how commodity technology could be used to take the next step in HPC system architecture.

Driven by a much larger market, commodity components evolve faster than their special-purpose counterparts, eventually achieving the same performance and eventually surpassing or replacing them. For this reason, RISC processors displaced vector processors, and x86 displaced RISC.

Nowadays commodity is in the embedded / mobile processor segment. Mobile processors develop fast, and are still not at a point of diminishing performance improvements from new designs. Furthermore, they progressively incorporate the capabilities required for HPC.

The embedded market size and endless customer requirements allow for constant investments into innovative designs, and rapid testing and adoption of new technologies. For example, LPDDR memory technology was first introduced in the mobile domain and has recently been proposed as a memory solution for energy proportional servers.

The Mont-Blanc project aims at providing an alternative HPC system solution based on the current commodity technology: mobile chips. As a demonstrator of such an approach, the project designed, built, and set-up a 1080-node HPC cluster made of Samsung Exynos 5250 SoCs. The Mont-Blanc project established the following goals: to design and deploy a sufficiently large HPC prototype system based on the current mobile commodity technology; to port and optimize the software stack, and enable its use for HPC; to port and optimize a set of HPC applications to be run at this HPC system.

Comparing the Mont-Blanc prototype to a contemporary supercomputer, MareNostrum III, reveals that a single-socket Mont-Blanc node is 9x slower than a dual-socket MareNostrum III node, while saving up to 40% of energy. MPI parallel applications show a 3.5x slowdown when running with the same number of MPI ranks on both machines, while consuming 9% less energy on the Mont-Blanc prototype on average. When targeting the same execution time, the Mont-Blanc prototype offers 12.5% space savings.

This work was funded by the European Commission with the Mont-Blanc projects 8.3.1.1 . This scientific result was presented at the SuperComputing Conference SC'2016 in Salt Lake City [31]. The paper was selected as a *best paper finalist*.

6.18. Control of Autonomid Parallelism on Software Transactional Memory

Participants: Naweiluo Zhou, Gwenaël Delaval [Univ. Grenoble Alpes, Associate Professor, Ctrl-A Inria team], Bogdan Robu [Univ. Grenoble Alpes, Associate Professor, Gipsa Laboratory], Eric Rutten [Inria, Rsearcher, Ctrl-A Inria team], Jean-François Méhaut.

Parallel programs need to manage the trade-off between the time spent in synchronization and computation. A high parallelism may decrease computing time while increase synchronization cost among threads. A way to improve program performance is to adjust parallelism to balance conflicts among threads. However, there is no universal rule to decide the best parallelism for a program from an offline view. Furthermore, an offline tuning is error-prone. Hence, it becomes necessary to adopt a dynamic tuning-configuration strategy to better manage a STM system. Software Transactional Memory (STM) has emerged as a promising technique, which bypasses locks, to address syn- chronization issues through transactions. Autonomic computing offers designers a framework of methods and techniques to build automated systems with well-mastered behaviours. Its key idea is to implement feedback control loops to design safe, efficient and predictable controllers, which enable monitoring and adjusting controlled systems dynamically while keeping overhead low. We propose to design feedback control loops to automate the choice of parallelism level at runtime to diminish program execution time.

This work is funded by the Persyval laboratory (LabEx) and the HPES team 8.1.2 . This scientific result is part of the Naweiluo Zhou's thesis. The thesis was defended in October 2016 [2]. This work was presented in the HPCS conference [25]. The paper was selected as *best paper finalist*. The Naweiluo Zhou's work is also presented at the ICAC conference.

6.19. Evaluating the SEE sensitivity of a 45nm SOI Multi-core Processor due to 14 MeV Neutrons

Participants: Pablo Ramos [Univ. Grenoble Alpes and ESPE Ecuador, PhD student TIMA Laboratory], Vanessa Vargas [Univ. Grenoble Alpes and ESPE Ecuador, PhD student TIMA Laboratory], Maud Baylac [CNRS, IN2P3, LSPSC Laboratory], Francesca Villa [CNRS, IN2P3, LSPSC Laboratory], Nacer-Eddine Zergainoh [Univ. Grenoble Alpes, Associate Professor, TIMA Laboratory], Jean-François Méhaut, Raoul Velazco [CNRS, Senior Scientist, TIMA Laboratory].

The aim of this work is to evaluate the SEE sensitivity of a multi-core processor having implemented ECC and parity in their cache memories. Two different application scenarios are studied. The first one configures the multi-core in Asymmetric Multi-Processing mode running a memory-bound application, whereas the second one uses the Symmetric Multi-Processsing mode running a CPU-bound application. The experiments were validated through radiation ground testing performed with 14 MeV neutrons on the Freescale P2041 multi-core manufactured in 45nm SOI technology. A deep analysis of the observed errors in cache memories was carried-out in order to reveal vulnerabilities in the cache protection mechanisms. Critical zones like tag addresses were affected during the experiments. In addition, the results show that the sensitivity strongly depends on the application and the multi-processsing mode used.

This work is part of the STIC Amsud EnergySFE project 8.4.3 . These results are published in the IEEE Transactions on Nuclear Science [10].

DREAMPAL Project-Team

6. New Results

6.1. A Language-Independent Proof System for Full Program Equivalence

Two programs are mutually equivalent if, for the same input, either they both diverge or they both terminate with the same result. 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 [11] we introduce a language-independent proof system for mutual equivalence, which is para-metric 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. The Collatz sequence is an interesting case study since it is not known wether the sequence terminates or not; nevertheless, our proof system shows that the two programs are mutually equivalent (even if we cannot establish termination or divergence of either one).

6.2. A Generic Framework for Symbolic Execution: a Coinductive Approach

In [12] we propose a language-independent symbolic execution framework. The approach is parameterised by a language definition, which consists of a signature for the lan-guage's syntax and execution infrastructure, a model interpreting the signature, and rewrite rules for the language's operational semantics. Then, symbolic execution amounts to computing symbolic paths using a derivative operation. We prove that the symbolic execution thus defined has the properties naturally expected from it, meaning that the feasible symbolic executions of a program and the concrete executions of the same program mutually simulate each other. We also show how a coinduction-based extension of symbolic execution can be used for the deductive verification of programs. We show how the proposed symbolic-execution approach, and the coinductive verification technique based on it, can be seamlessly implemented in language definition frameworks based on rewriting such as the K framework. A prototype implementation of our approach has been developed in K. We illustrate it on the symbolic analysis and deductive verification of nontrivial programs.

6.3. Circuit Merging versus Dynamic Partial Reconfiguration -The HoMade Implementation

One goal of reconfiguration is to save power and occupied resources. In [13] we compare two different kinds of reconfiguration available on field-programmable gate arrays (FPGA) and we discuss their pros and cons. The first method that we study is circuit merging. This type of reconfiguration methods consists in sharing common resources between different circuits. The second method that we explore is dynamic partial reconfiguration (DPR). It is specific to some FPGA, allowing well defined reconfigurable parts to be modified during runtime. We show that DPR, when available, has good and more predictable result in terms of occupied area. There is still a huge overhead in term of time and power consumption during the reconfiguration phase. Therefore we show that circuit merging remains an interesting solution on FPGA because it is not vendor specific and the reconfiguration time is around a clock cycle. Besides, good merging algorithms exist eventhough FPGA physical synthesis flow makes it hard to predict the real performance of the merged circuit during the optimization. We establish our comparison in the context of the HoMade processo

6.4. Language Definitions as Rewrite Theories

K is a formal framework for defining operational semantics of programming languages. The K-Maude compiler translates K language definitions to Maude rewrite theories. The compiler enables program execution by using the Maude rewrite engine with the compiled definitions, and program analysis by using various Maude analysis tools. K supports symbolic execution in Maude by means of an automatic transformation of language definitions. The transformed definition is called the symbolic extension of the original definnition. In [14] we investigate the theoretical relationship between K language definitions and their Maude translations, between symbolic extensions of K definitions and their Maude translations, and how the relationship between K definitions and their symbolic extensions is reflected on their respective representations in Maude. In particular, the results show how analysis performed with Maude tools can be formally lifted up to the original language definitions.

6.5. SCAC-Net: Reconfigurable Interconnection Network in SCAC Massively parallel SoC

Parallel communication plays a critical role in massively parallel systems, especially in distributed memory systems executing parallel programs on shared data. Therefore, integrating an interconnection network in these systems becomes essential to ensure data inter-nodes exchange. Choosing the most effective communication structure must meet certain criteria: speed, size and power consumption. Indeed, the communication phase should be as fast as possible to avoid compromising parallel computing, using small and low power consumption modules to facilitate the interconnection network extensibility in a scalable system. To meet these criteria and based on a module reuse methodology, we chose to integrate a reconfigurable SCAC-Net interconnection network to communicate data in SCAC Massively parallel SoC. In [15] we present the detailed hardware implementation and discusses the performance evaluation of the proposed reconfigurable SCAC-Net network.

6.6. Proving Reachability-Logic Formulas Incrementally

Reachability Logic (RL) is a formalism for defining the operational semantics of programming languages and for specifying program properties. As a program logic it can be seen as a language-independent alternative to Hoare Logics. Several verification techniques have been proposed for RL, all of which have a circular nature: the RL formula under proof can circularly be used as a hypothesis in the proof of another RL formula, or even in its own proof. This feature is essential for dealing with possibly unbounded repetitive behaviour (e.g., program loops). The downside of such approaches is that the verification of a set of RL formulas is monolithic, i.e., either all formulas in the set are proved valid, or nothing can be inferred about any of the formula's validity or invalidity. In [16] we propose a new, incremental method for proving a large class of RL formulas. The proposed method takes as input a given RL formula under proof (corresponding to a given program fragment), together with a (possibly empty) set of other valid RL formulas (e.g., already proved using our method), which specify sub-programs of the program fragment under verification. It then checks certain conditions are shown to be equivalent to the validity of the RL formula under proof. A newly proved formula can then be incrementally used in the proof of other RL formulas, corresponding to larger program fragments. The process is repeated until the whole program is proved. We illustrate our approach by verifying the nontrivial Knuth-Morris-Pratt string-matching program.

PACAP Project-Team

7. New Results

7.1. Compiler, vectorization, interpretation

Participants: Erven Rohou, Emmanuel Riou, Arjun Suresh, André Seznec, Nabil Hallou, Sylvain Collange, Rabab Bouziane, Arif Ali Ana-Pparakkal, Stefano Cherubin.

7.1.1. Improving sequential performance through memoization

Participants: Erven Rohou, Emmanuel Riou, 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 propose [20] 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.

Initial work has been published in ACM TACO 2015 [20] and accepted for presentation at the International Conference HiPEAC 2016 in Prague.

Further developments have been accepted for publication at the Compiler Construction Conference 2017 [49].

This research is described in the PhD thesis of Arjun Suresh [24].

7.1.2. Optimization in the Presence of NVRAM

Participants: Erven Rohou, Rabab Bouziane.

Energy-efficiency is one of the most challenging design issues in both embedded and high-performance computing domains. The aim is to reduce as much as possible the energy consumption of considered systems while providing them with the best computing performance. Finding an adequate solution to this problem certainly requires a cross-disciplinary approach capable of addressing the energy/performance trade-off at different system design levels.

We proposed [42] an empirical impact analysis of the integration of Spin Transfer Torque Magnetic Random Access Memory (STT-MRAM) technologies in multicore architectures when applying some existing compiler optimizations. For that purpose, we use three well-established architecture and NVM evaluation tools: NVSim, gem5 and McPAT. Our results show that the integration of STT-MRAM at cache memory levels enables a significant reduction of the energy consumption (up to 24.2 % and 31 % on the considered multicore and monocore platforms respectively) while preserving the performance improvement provided by typical code optimizations. We also identified how the choice of the clock frequency impacts the relative efficiency of the considered memory technologies.

This research is done in collaboration with Abdoulaye Gamatié at LIRMM (Montpellier) within the context the the ANR project CONTINUUM.

7.1.3. Hardware/Software JIT Compiler

Participant: Erven Rohou.

Dynamic Binary Translation (DBT) is often used in hardware/software co-design to take advantage of an architecture model while using binaries from another one. The co-development of the DBT engine and of the execution architecture leads to architecture with special support to these mechanisms. We proposed a hardware accelerated dynamic binary translation where the first steps of the DBT process are fully accelerated in hardware. Results shows that using our hardware accelerators leads to a speed-up of 8× and a cost in energy 18× lower, compared with an equivalent software approach.

An initial version of this work has been presented at Compas'16 [51]. The latest results have been accepted for publication at DATE 2017 [44].

This research is done in collaboration with Steven Derrien and Simon Rokicki from the CAIRN team.

7.1.4. Dynamic Parallelization of Binary Programs

Participants: Erven Rohou, Emmanuel Riou, Nabil Hallou.

We address runtime automatic parallelization of binary executables, assuming no previous knowledge on the executable code. The Padrone platform is used to identify candidate functions and loops. Then we disassemble the loops and convert them to the intermediate representation of the LLVM compiler. This allows us to leverage the power of the polyhedral model for auto-parallelizing loops. Once optimized, new native code is generated just-in-time in the address space of the target process.

Our approach enables user transparent auto-parallelization of legacy and/or commercial applications with auto-parallelization.

This work has been accepted for publication in the Springer journal IJPP: "Runtime Vectorization Transformations of Binary Code".

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

7.1.5. Dynamic Function Specialization

Participants: Erven Rohou, Arif Ali Ana-Pparakkal.

Compilers can do better optimization with the knowledge of run-time behaviour of the program. *Function Specialization* is an optimization technique in which different versions of a function are created according to the value of its arguments. It can be difficult to predict the exact value/behaviour of arguments during static compilation and so it is difficult for a static compiler to do efficient function specialization. In our *dynamic function specialization* technique, we capture the actual value of arguments during execution of the program and, when profitable, create specialized versions and include them at runtime.

This research is done within the context of the Nano 2017 PSAIC collaborative project.

7.1.6. Application Autotuning for Performance and Energy

Participants: Erven Rohou, Stefano Cherubin, Imane Lasri.

Due to the increasing complexity of both applications behaviors and underlying hardware, achieving reasonable (not to mention best) performance can hardly be done at compile time. Autotuning is an approach where a runtime manager is able to adapt the software to the runtime conditions. We have developed a framework and shown through a domain specific application initial exploration scenarios [32], [47].

We started characterizing applications – in particular the Parasuite benchmarks – and we will rely on split-compilation [2] embed hints and heuristics inside a binary program for dynamic adaptation and optimization.

This research is done within the context of the H2020 FET HPC collaborative project ANTAREX.

7.1.7. Customized Precision Computing

Participants: Erven Rohou, Stefano Cherubin, Imane Lasri.

Customized precision originates from the fact that many applications can tolerate some loss of quality during computation, as in the case of media processing (audio, video and image), data mining, machine learning, etc. Error-tolerating applications are increasingly common in the emerging field of real-time HPC. Thus, recent works have investigated this line of research in the HPC domain as a way to provide a breakthrough in power and performance for the Exascale era.

We aim at leveraging existing, HPC-oriented hardware architectures, while including in the precision tuning an adaptive selection of floating and fixed-point arithmetic. It is part of a wider effort to provide the programmers with an easy way to manage extra-functional properties of programs, including precision, power, and performance.

We explore tradeoffs between precision and time-to-solution, as well as precision and energy-to-solution.

This is done within the context of the ANTAREX project in collaboration with Stefano Cherubin, Cristina Silvano and Giovanni Agosta from Politecnico di Milano, and Olivier Sentieys from the CAIRN team.

7.1.8. SPMD Function Call Re-Vectorization

Participant: Sylvain Collange.

SPMD programming languages for SIMD hardware such as C for CUDA, OpenCL or ISPC have contributed to increase the programmability of SIMD accelerators and graphics processing units. However, SPMD languages still lack the flexibility offered by low-level SIMD programming on explicit vectors. To close this expressiveness gap while preserving the SPMD abstraction, we introduce the notion of Function Call Re-Vectorization (CREV) [38]. CREV allows changing the dimension of vectorization during the execution of an SPMD kernel, and exposes it as a nested parallel kernel call. CREV affords a programmability close to dynamic parallelism, a feature that allows the invocation of kernels from inside kernels, but at much lower cost. In this paper, we present a formal semantics of CREV, and an implementation of it on the ISPC compiler. To validate our idea, we have used CREV to implement some classic algorithms, including string matching, depth first search and Bellman-Ford, with minimum effort. These algorithms, once compiled by ISPC to Intel-based vector instructions, are as fast as state-of-the-art implementations, yet much simpler. As an example, our straightforward implementation of string matching beats the Knuth-Morris-Pratt algorithm by 12 %.

This work was done during the internship of Rubens Emilio in Rennes in collaboration with Sylvain Collange and Fernando Pereira (UFMG) as part of the Inria PROSPIEL Associate Team.

7.1.9. SPMD Function Call Fusion

Participant: Sylvain Collange.

The increasing popularity of Graphics Processing Units (GPUs) has brought renewed attention to old problems related to the Single Instruction, Multiple Data execution model. One of these problems is the reconvergence of divergent threads. A divergence happens at a conditional branch when different threads disagree on the path to follow upon reaching this split point. Divergences may impose a heavy burden on the performance of parallel programs.

We have proposed a compiler-level optimization to mitigate the performance loss due to branch divergence on GPUs [21]. This optimization consists in merging function call sites located at different paths that sprout from the same branch. We show that our optimization adds negligible overhead on the compiler. When not applicable, it does not slow down programs and it accelerates substantially those in which it is applicable. As an example, we have been able to speed up the well known SPLASH Fast Fourier Transform benchmark by 11 %.

This work is done in collaboration with Douglas do Couto Teixeira and Fernando Pereira from UFMG as part of the Inria PROSPIEL Associate Team.

7.1.10. SIMD programming in SPMD: application to multi-precision computations

Participant: Sylvain Collange.

GPUs are an important hardware development platform for problems where massive parallel computations are needed. Many of these problems require a higher precision than the standard double floating-point (FP) available. One common way of extending the precision is the multiple-component approach, in which real numbers are represented as the unevaluated sum of several standard machine precision FP numbers. This representation is called a FP expansion and it offers the simplicity of using directly available and highly optimized FP operations. We propose new data-parallel algorithms for adding and multiplying FP expansions specially designed for extended precision computations on GPUs [34]. These are generalized algorithms that can manipulate FP expansions of different sizes (from double-double up to a few tens of doubles) and ensure a certain worst case error bound on the results.

This work is done in collaboration with Mioara Joldes (CNRS/LAAS), Jean-Michel Muller (CNRS/LIP) and Valentina Popescu (ENS Lyon/LIP).

7.2. Processor Architecture

Participants: Pierre Michaud, Sylvain Collange, Erven Rohou, André Seznec, Arthur Perais, Sajith Kalathingal, Andrea Mondelli, Aswinkumar Sridharan, Biswabandan Panda, Fernando Endo, Kleovoulos Kalaitzidis.

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

7.2.1. Microarchitecture

7.2.1.1. Branch prediction

Participant: André Seznec.

IMLI-based predictors

The wormhole (WH) branch predictor was recently introduced to exploit branch outcome correlation in multidimensional loops. For some branches encapsulated in a multidimensional loop, their outcomes are correlated with those of the same branch in neighbor iterations, but in the previous outer loop iteration. In [18], we introduced practical predictor components to exploit this branch outcome correlation in multidimensional loops: the IMLI-based predictor components. The iteration index of the inner most loop in an application can be efficiently monitored at instruction fetch time using the Inner Most Loop Iteration (IMLI) counter. The outcomes of some branches are strongly correlated with the value of this IMLI counter. Our experiments show that augmenting a state-of-the-art global history predictor such as TAGE-SC-L [45] with IMLI-based components outperforms previous state-of-the-art academic predictors leveraging local and global history at much lower hardware complexity (i.e., smaller storage budget , smaller number of tables and simpler management of speculative states).

This study was accepted in the special issue Top Picks of the best papers in 2015 computer architecture conferences in IEEE Micro [30].

This research was done in collaboration with Joshua San Miguel and Jorge Albericio from University of Toronto

Championship Branch Prediction

The 5th Championship Branch Prediction was organized in Seoul in June 2016. The predictors submitted by the PACAP-team, respectively TAGE-SC-L and MTAGE-SC, for limited storage budgets and infinite storage budgets won the three tracks of the competition [46], [45]. These predictors are derived from our reference work [17].

7.2.1.2. 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. From 2013 to 2016, we have progressively revived the interest in value prediction. At a first step, we showed that all predictors are amenable to very high accuracy at the cost of some loss on prediction coverage [12]. Furthermore, we proposed EOLE [13]. EOLE leverages Value Prediction to *Early Execute* simple instructions whose operands are ready in parallel with Rename and to *Late Execute* to simple predicted instructions just before Commit. EOLE allows to reduce the out-of-order issue-width by 33% without impeding performance.

An extension of the initial EOLE paper [13] was published in ACM TOCS [27].

7.2.1.3. Physical register sharing

Participants: Arthur Perais, André Seznec.

Sharing a physical register between several instructions is needed to implement several microarchitectural optimizations. However, register sharing requires modifications to the register reclaiming process: Committing a single instruction does not guarantee that the physical register allocated to the previous mapping of its architectural destination register is free-able anymore. Consequently, a form of register reference counting must be implemented. While such mechanisms (e.g., dependency matrix, per register counters) have been described in the literature, we argue that they either require too much storage, or that they lengthen branch misprediction recovery by requiring sequential rollback. As an alternative, we present the Inflight Shared Register Buffer (ISRB), a new structure for register reference counting [41]. The ISRB has low storage overhead and lends itself to checkpoint-based recovery schemes, therefore allowing fast recovery on pipeline flushes. We illustrate our scheme with Move Elimination (short-circuiting moves) and an implementation of Speculative Memory Bypassing (short-circuiting store-load pairs) that makes use of a TAGE-like predictor to identify memory dependencies. We show that the whole potential of these two mechanisms can be achieved with a small register tracking structure.

7.2.1.4. Register Sharing for Equality Prediction

Participants: Arthur Perais, Fernando Endo, André Seznec.

Recently, Value Prediction (VP) has been gaining renewed traction in the research community. VP speculates on the result of instructions to increase Instruction Level Parallelism (ILP). In most embodiments, VP requires large tables to track predictions for many static instructions. However, in many cases, it is possible to detect that the result of an instruction is produced by an older inflight instruction, but not to predict the result itself. Consequently it is possible to rely on predicting register equality and handle speculation through the renamer. To do so, we propose to use Distance Prediction [40], a technique that was previously used to perform Speculative Memory Bypassing (short-circuiting def-store-load-use chains). Distance Prediction attempts to determine how many instructions separate the instruction of interest and the most recent older instruction that produced the same result. With this information, the physical register identifier of the older instruction can be retrieved from the ROB and provided to the renamer. The implementation of Distance Prediction necessitates a hardware mechanism to handle the sharing of physical registers as the ISRB [41].

7.2.1.5. Storage-Free Memory Dependency Prediction

Participants: Arthur Perais, André Seznec.

Memory Dependency Prediction (MDP) is paramount to good out-of-order performance, but decidedly not trivial as a all instances of a given static load may not necessarily depend on all instances of a given static store. As a result, for a given load, MDP should predict the exact store instruction the load depends on, and not only whether it depends on an inflight store or not, i.e., ideally, prediction should not be binary. However, we first argue that given the high degree of sophistication of modern branch predictors, the fact that a given dynamic load depends on an inflight store can be captured using the binary prediction capabilities of the branch predictor, providing coarse MDP at zero storage overhead. Second, by leveraging hysteresis counters, we show that the precise producer store can in fact be identified. This embodiment of MDP yields performance levels that are on par with state-of-the-art, and requires less than 70 additional bits of storage over a baseline without MDP at all [28].

7.2.1.6. Compressed Caches

Participants: André Seznec, Biswabandan Panda.

The YACC compressed cache

Cache memories play a critical role in bridging the latency, bandwidth, and energy gaps between cores and off-chip memory. However, caches frequently consume a significant fraction of a multicore chip's area, and thus account for a significant fraction of its cost. Compression has the potential to improve the effective capacity of a cache, providing the performance and energy benefits of a larger cache while using less area. The design of a compressed cache must address two important issues: i) a low-latency, low-overhead compression algorithm that can represent a fixed-size cache block using fewer bits and ii) a cache organization that can efficiently store the resulting variable-size compressed blocks. This paper focuses on the latter issue. We propose YACC (Yet Another Compressed Cache), a new compressed cache design that targets improving effective cache capacity with a simple design [29]. YACC uses super-blocks to reduce tag overheads, while packing variable-size compressed blocks to reduce internal fragmentation. YACC achieves the benefits of two state-of-the art compressed caches, Decoupled Compressed Cache (DCC) [61] and Skewed Compressed Cache (SCC) [15], with a more practical and simpler design. YACC's cache layout is similar to conventional caches, with a largely unmodified tag array and unmodified data array.

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

The DISH compression scheme

The effectiveness of a compressed cache depends on three features: i) the compression scheme, ii) the compaction scheme, and iii) the cache layout of the compressed cache. Both SCC [15] and YACC [29] use compression techniques to compress individual cache blocks, and then a compaction technique to compact multiple contiguous compressed blocks into a single data entry. The primary attribute used by these techniques for compaction is the compression factor of the cache blocks, and in this process, they waste cache space. We propose dictionary sharing (DISH), a dictionary based cache compression scheme that reduces this wastage [39]. DISH compresses a cache block by keeping in mind that the block is a potential candidate for the compaction process. DISH encodes a cache block with a dictionary that stores the distinct 4-byte chunks of a cache block and the dictionary is shared among multiple neighboring cache blocks. The simple encoding scheme of DISH also provides a single cycle decompression latency and it does not change the cache layout of compressed caches. Compressed cache layouts that use DISH outperforms the compression schemes, such as BDI and CPACK+Z, in terms of compression ratio, system performance , and energy efficiency.

7.2.1.7. 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.

In [8], 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 study published in ACM TACO in 2015 [8] and was presented at the HIPEAC 2016 conference.

7.2.1.8. Hardware data prefetching

Participant: Pierre Michaud.

Hardware prefetching is an important feature of modern high-performance processors. When an application's working set is too large to fit in on-chip caches, disabling hardware prefetchers may result in severe performance reduction. We propose a new hardware data prefetcher, the Best-Offset (BO) prefetcher. The BO prefetcher is an offset prefetcher using a new method for selecting the best prefetch offset taking into account prefetch timeliness. The hardware required for implementing the BO prefetcher is very simple. A version of the BO prefetcher won the 2015 Data Prefetching Championship. A comprehensive study of the BO prefetcher was presented at the HPCA 2016 conference [37].

7.2.1.9. Exploiting loops for lower energy consumption

Participants: Andrea Mondelli, Pierre Michaud, André Seznec.

Recent superscalar processors use a loop buffer to decrease the energy consumption in the front-end. The energy savings comes from the branch predictor, instruction cache and instruction decoder being idle when micro-ops are delivered to the back-end from the loop buffer. We explored the possibility to exploit loop behaviors for decreasing energy consumption further, in the back-end, without impacting performance. We proposed two independent optimizations requiring little extra hardware. The first optimization detects and removes from the execution redundant micro-ops producing the same result in every loop iteration. The second optimization focuses on loop loads and detects situations where a loop load needs accessing only the data cache, or only the store queue, not both.

7.2.2. Microarchitecture Performance Modeling

7.2.2.1. Optimal cache replacement

Participant: Pierre Michaud.

A cache replacement policy is an algorithm, implemented in hardware, selecting a block to evict to make room for an incoming block. This research topic has been revitalized recently, as level-2 and level-3 caches were integrated on chip. A cache replacement policy cannot be optimal in general unless it has the knowledge of future references. Unfortunately, practical replacement policies do not have this knowledge. Still, optimal replacement is an important benchmark for understanding replacement policies. Moreover, some new replacement policies proposed recently are directly inspired from algorithms for determining hits and misses under optimal replacement. Hence it is important to improve our understanding of optimal replacement.

The OPT policy, which evicts the block referenced furthest in the future, was proved optimal by Mattson et al. [57]. However, their proof is long and somewhat complicated. In collaboration with some researchers from Inha University, we found a shorter and more intuitive proof of optimality for OPT [6].

An intriguing aspect of optimal replacement, seldom mentioned in the literature, is the fact that Belady's MIN algorithm determines OPT hits and misses without the knowledge of future references [54]. Starting from this fact, we searched and found a new algorithm, different from MIN, for determining OPT hits and misses. This algorithm provides new insights about optimal replacement. We show that traces of OPT stack distances have a distinctive structure. In particular, we prove that OPT miss curves are always convex. We show that, like an LRU cache, an OPT cache cannot experience more misses as the reuse distance of references is decreased. Consequently, accessing data circularly is the worst access pattern for OPT, like it is for LRU. We discovered an equivalence between an OPT cache of associativity N with bypassing allowed and an OPT cache of associativity N+1 with bypassing disabled. A paper deriving these results was accepted in ACM TACO and will be presented at the HiPEAC 2017 conference [25].

7.2.2.2. Adaptive Intelligent Memory Systems

Participants: André Seznec, Aswinkumar Sridharan.

Multi-core processors employ shared Last Level Caches (LLC). This trend will continue in the future with large multi-core processors (16 cores and beyond) as well. At the same time, the associativity of this LLC tends to remain in the order of sixteen. Consequently, with large multicore processors, the number of cores that share the LLC becomes larger than the associativity of the cache itself. LLC management policies have been extensively studied for small scale multi-cores (4 to 8 cores) and associativity degree in the 16 range. However, the impact of LLC management on large multi-cores is essentially unknown, in particular when the associativity degree is smaller than the number of cores.

In [48], we introduce Adaptive Discrete and deprioritized Application PrioriTization (ADAPT), an LLC management policy addressing the large multi-cores where the LLC associativity degree is smaller than the number of cores. ADAPT builds on the use of the Footprint-number metric. Footprint-number is defined as the number of unique accesses (block addresses) that an application generates to a cache set in an interval of time. We propose a monitoring mechanism that dynamically samples cache sets to estimate the Footprint-number of applications and classifies them into discrete (distinct and more than two) priority buckets. The cache replacement policy leverages this classification and assigns priorities to cache lines of applications during cache replacement operations. Footprint-number is computed periodically to account the dynamic changes in applications behavior. We further find that de- prioritizing certain applications during cache replacement is beneficial to the overall performance. We evaluate our proposal on 16, 20 and 24-core multi-programmed workloads and discuss other aspects in detail.

[48] got the best paper award at the IPDPS 2016 conference.

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

Participants: Sylvain Collange, André Seznec, Sajith Kalathingal.

Threads of Single-Program Multiple-Data (SPMD) applications often exhibit very similar control flows, i.e. they execute the same instructions on different data. In [36] we propose the Dynamic Inter-Thread Vectorization Architecture (DITVA) to leverage this implicit data-level parallelism in SPMD applications by assembling dynamic vector instructions at runtime. DITVA extends an in-order SMT processor with SIMD units with an inter-thread vectorization execution mode. In this mode, multiple scalar threads running in lockstep share a single instruction stream and their respective instruction instances are aggregated into SIMD instructions. To balance thread-and data-level parallelism, threads are statically grouped into fixed-size independently scheduled warps. DITVA leverages existing SIMD units and maintains binary compatibility with existing CPU architectures. Our evaluation on the SPMD applications from the PARSEC and Rodinia OpenMP benchmarks shows that a 4-warp $\times$ 4-lane 4-issue DITVA architecture with a realistic bank-interleaved cache achieves 1.55$\times$ higher performance than a 4-thread 4-issue SMT architecture with AVX instructions while fetching and issuing 51 % fewer instructions, achieving an overall 24 % energy reduction.

Our paper [36] received the Best Paper Award of the SBAC-PAD conference.

7.2.2.4. Generalizing the SIMT execution model to general-purpose instruction sets

Participant: Sylvain Collange.

The *Single Instruction, Multiple Threads* (SIMT) execution model as implemented in NVIDIA Graphics Processing Units (GPUs) associates a multi-thread programming model with an SIMD execution model [59]. It combines the simplicity of scalar code from the programmer's and compiler's perspective with the efficiency of SIMD execution units at the hardware level. However, current SIMT architectures demand specific instruction sets. In particular, they need specific branch instructions to manage thread divergence and convergence. Thus, SIMT GPUs have remained incompatible with traditional general-purpose CPU instruction sets.

We designed Simty, an SIMT processor proof of concept that lifts the instruction set incompatibility between CPUs and GPUs [50]. Simty is a massively multi-threaded processor core that dynamically assembles SIMD instructions from scalar multi-thread code. It runs the RISC-V (RV32-I) instruction set. Unlike existing SIMD or SIMT processors like GPUs, Simty takes binaries compiled for general-purpose processors without any instruction set extension or compiler changes. Simty is described in synthesizable RTL. A FPGA prototype validates its scaling up to 2048 threads per core with 32-wide SIMD units.

7.3. WCET estimation and optimization

Participants: Isabelle Puaut, Damien Hardy, Viet Anh Nguyen, Benjamin Rouxel, Sébastien Martinez, Erven Rohou.

7.3.1. WCET estimation for many core processors

Participants: Viet Anh Nguyen, Damien Hardy, Sébastien Martinez, Isabelle Puaut, Benjamin Rouxel.

7.3.1.1. Optimization of WCETs by considering the effects of local caches

The overall goal of this research is to defined WCET estimation methods for parallel applications running on many-core architectures, such as the Kalray MPPA machine.

Some approaches to reach this goal have been proposed, but they assume the mapping of parallel applications on cores already done. Unfortunately, on architectures with caches, task mapping requires a priori known WCETs for tasks, which in turn requires knowing task mapping (i.e., co-located tasks, co-running tasks) to have tight WCET bounds. Therefore, scheduling parallel applications and estimating their WCET introduce a chicken and egg situation.

We address this issue by developing both optimal and heuristic techniques for solving the scheduling problem, whose objective is to minimize the WCET of a parallel application. Our proposed static partitioned non-preemptive mapping strategies address the effect of local caches to tighten the estimated WCET of the parallel application. Experimental results obtained on real and synthetic parallel applications show that co-locating tasks that reuse code and data improves the WCET.

This research is part of the PIA Capacités project.

7.3.1.2. Accounting for shared resource contentions to minimize WCETs

Accurate WCET analysis for multi-cores is known to be challenging, because of concurrent accesses to shared resources, such as communication through busses or Networks on Chips (NoC). Since it is impossible in general to guarantee the absence of resource conflicts during execution, current WCET techniques either produce pessimistic WCET estimates or constrain the execution to enforce the absence of conflicts, at the price of a significant hardware under-utilization. In addition, the large majority of existing works consider that the platform workload consists of independent tasks. As parallel programming is the most promising solution to improve performance, we envision that within only a few years from now, real-time workloads will evolve toward parallel programs. The WCET behavior of such programs is challenging to analyze because they consist of *dependent* tasks interacting through complex synchronization/communication mechanisms.

In this work, we propose techniques that account for interferences to access shared ressources, in order to minimize the WCET of parallel applications. An optimal and a heuristic method are proposed to map and schedule tasks on multi-cores. These methods take the structure of applications (synchronizations/communications) into consideration to tightly identify shared resource interferences and consequently tighten WCET estimates.

This work is performed in cooperation with Steven Derrien, Angeliki Kritikakou and Imen Fassi from the CAIRN research group and is part of the ARGO H2020 project.

7.3.2. Cache-Persistence-Aware Response-Time Analysis for Fixed-Priority Preemptive Systems

Participants: Damien Hardy, Isabelle Puaut.

A task can be preempted by several jobs of higher priority tasks during its execution. Assuming the worst-case memory demand for each of these jobs leads to pessimistic worst-case response time (WCRT) estimations. Indeed, there is a big chance that a large portion of the instructions and data associated with the preempting task τ_j are still available in the cache when τ_j releases its next jobs. Accounting for this observation allows the pessimism of WCRT analysis to be significantly reduced, which is not considered by existing work.

The four main contributions of this work are: 1) The concept of persistent cache blocks is introduced in the context of WCRT analysis, which allows re-use of cache blocks to be captured, 2) A cache-persistence-aware WCRT analysis for fixed-priority preemptive systems exploiting the PCBs to reduce the WCRT bound, 3) A multi-set extension of the analysis that further improves the WCRT bound and 4) An evaluation showing that our cache-persistence-aware WCRT analysis results in up to 10 % higher schedulability than state-of-the-art approaches.

This work [43] appeared at ECRTS 2016 and was selected as an outstanding paper in this conference.

This work was performed in cooperation with Syed Aftab Rashid, Geoffrey Nelissen, Benny Akesson and Eduardo Tovar from ISEP (Polytechnic Institute of Porto), Portugal.

7.4. Fault Tolerance

7.4.1. WCET estimation for architectures with faulty caches

Participants: Damien Hardy, Isabelle Puaut.

Fine-grained disabling and reconfiguration of hardware elements (functional units, cache blocks) will become economically necessary to recover from permanent failures, whose rate is expected to increase dramatically in the near future. This fine-grained disabling will lead to degraded performance as compared to a fault-free execution.

Until recently, all static worst-case execution time (WCET) estimation methods were assuming fault-free processors, resulting in unsafe estimates in the presence of faults. The first static WCET estimation technique dealing with the presence of permanent faults in instruction caches was proposed in [4]. This study probabilistically quantified the impact of permanent faults on WCET estimates. It demonstrated that the probabilistic WCET (pWCET) estimates of tasks increase rapidly with the probability of faults as compared to fault-free WCET estimates.

New results show that very simple reliability mechanisms allow mitigating the impact of faulty cache blocks on pWCETs. Two mechanisms, that make part of the cache resilient to faults are analyzed. Experiments show that the gain in pWCET for these two mechanisms are on average 48 % and 40 % as compared to an architecture with no reliability mechanism.

This work [35] appeared at DATE 2016 (best paper award for the embedded systems track).

This is joint work with Yannakis Sazeides from University of Cyprus.

TASC Project-Team

7. New Results

7.1. Discrete Convexity

We introduce a propagator for pairs of Sum constraints, where the expressions in the sums respect a form of convexity. This propagator is parametric and can be instantiated for various concrete pairs, including Deviation, Spread, and the conjunction of Linear($\leq$) and Among. We show that despite its generality, our propagator (see Figure 1) is competitive in theory and practice with state-of-the-art propagators. (see AI journal paper).

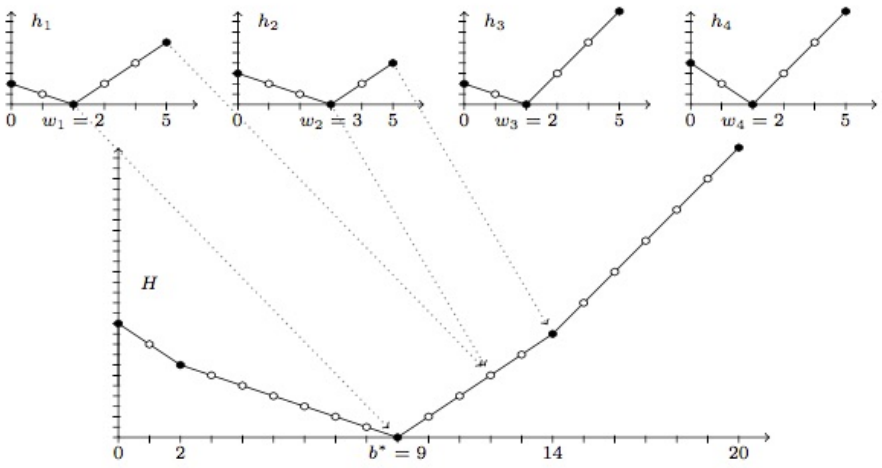

Figure 1. Illustration of the filtering wrt h function

7.2. Transducers

We describe a large family of constraints for structural time series by means of function composition. These constraints are on aggregations of features of patterns that occur in a time series, such as the number of its peaks, or the range of its steepest ascent. The patterns and features are usually linked to physical properties of the time series generator, which are important to capture in a constraint model of the system, i.e. a conjunction of constraints that produces similar time series. We formalise the patterns using finite transducers, whose output alphabet corresponds to semantic values that precisely describe the steps for identifying the occurrences of a pattern. Based on that description, we automatically synthesise automata with accumulators, as well as constraint checkers. The description scheme not only unifies the structure of the existing 30 time-series constraints in the Global Constraint Catalogue, but also leads to over 600 new constraints, with more than 100,000 lines of synthesised code. (see Constraint journal paper)

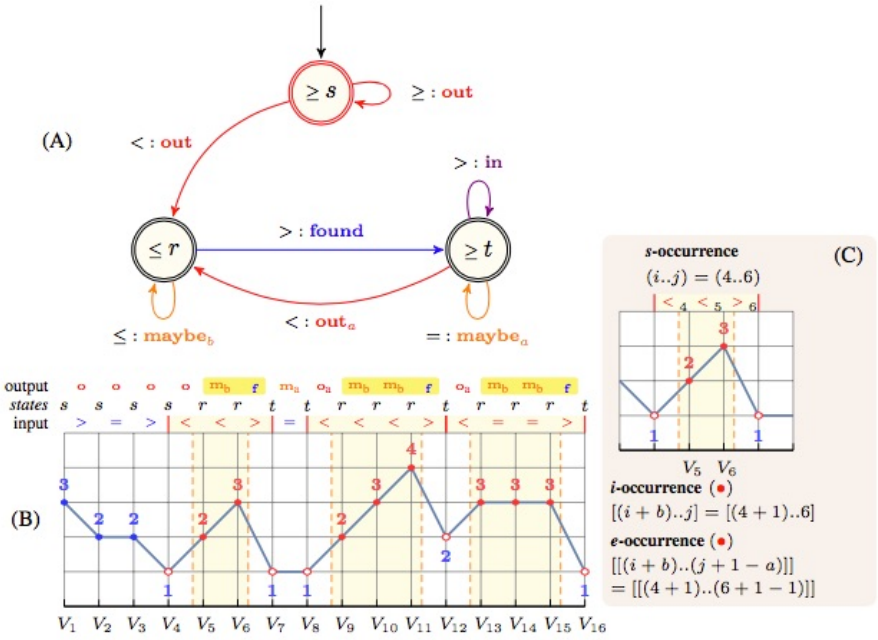

Figure 2. Transducer for the peak pattern and its execution on a sequence

7.3. Compositional Glue Matrix and Bound for Time-Series Constraints

Integer time series are often subject to constraints on the aggregation of the integer features of all occurrences of some pattern within the series. For example, the number of inflexions may be constrained, or the sum of the peak maxima, or the minimum of the peak widths. It is currently unknown how to maintain domain consistency efficiently on such constraints. We propose parametric ways of systematically deriving glue constraints (see Figures 3 and 4 for the parametric and concrete glue constraints), which are a particular kind of implied constraints, as well as aggregation bounds (see Figure 5) that can be added to the decomposition of time-series constraints. We evaluate the beneficial propagation impact of the derived implied constraints and bounds, both alone and together. (see CP conference paper)

	s	r	t
s	$\phi_g(\overrightarrow{C}, \overleftarrow{C})$	$\phi_g(\overrightarrow{C}, \overleftarrow{C})$	$\phi_g(\overrightarrow{C}, \overleftarrow{C})$
r	$\phi_g(\overrightarrow{C}, \overleftarrow{C})$	$\phi_f(\overrightarrow{D}, \overleftarrow{D}, \delta_f^i)$	$\phi_f(\overleftarrow{C}, \overrightarrow{D}, \overleftarrow{D}, \delta_f^i)$
t	$\phi_g(\overrightarrow{C}, \overleftarrow{C})$	$\phi_f(\overrightarrow{C}, \overrightarrow{D}, \overleftarrow{D}, \delta_f^i)$	$\phi_g(\overrightarrow{C}, \overleftarrow{C})$

Figure 3. Parametrised glue matrix for the peak pattern expressed in term of parametrised functions depending on the states pairs between the prefix and the suffix of a sequence

7.4. Reformulation of time-series constraint in MIP

	s	r	t
s	$\overrightarrow{C}+\overleftarrow{C}$	$\overrightarrow{C}+\overleftarrow{C}$	$\overrightarrow{C}+\overleftarrow{C}$
r	$\overrightarrow{C}+\overleftarrow{C}$	1	1
t	$\overrightarrow{C}+\overleftarrow{C}$	1	$\overrightarrow{C}+\overleftarrow{C}$

Figure 4. Concrete glue matrix for the number peak constraint expressed in term of concrete functions depending on the states pairs between the prefix and the suffix of a sequence

$\mathtt{sv} \leq 3 \vee \mathtt{rv} \leq 1 \Rightarrow \mathtt{VALUE} = 0$
$\mathtt{VALUE} \geq 0$
$\mathtt{rv} = 2 \Rightarrow \mathtt{VALUE} \leq \lfloor \mathtt{sv}/4 \rfloor$①
$\mathtt{rv} \geq 3 \Rightarrow \mathtt{VALUE} \leq \mathtt{max}(0, \lfloor (\mathtt{sv}-1)/3 \rfloor)$②
required(VARIABLES, var)
where
$\mathtt{sv} = |\mathtt{VARIABLES}|$
$\mathtt{rv} =$range(VARIABLES.var)

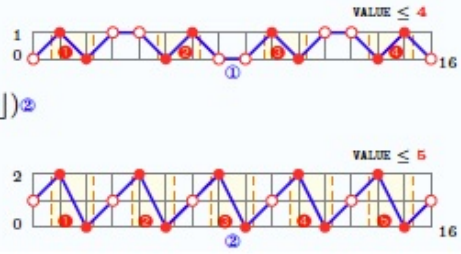

Figure 5. Upper bound on the number of zigzag depending on the domain range being equal to 2 or greater than or equal to 3

A checker for a constraint on a variable sequence can often be compactly specified by an automaton, possibly with accumulators, that consumes the sequence of values taken by the variables; such an automaton can also be used to decompose its specified constraint into a conjunction of logical constraints. The inference achieved by this decomposition in a CP solver can be boosted by automatically generated implied constraints on the accumulators, provided the latter are updated in the automaton transitions by linear expressions. Automata with non-linear accumulator updates can be automatically synthesised for a large family of time-series constraints. In this paper, we describe and evaluate extensions to those techniques. First, we improve the automaton synthesis to generate automata with fewer accumulators. Second, we decompose a constraint specified by an automaton with accumulators into a conjunction of linear inequalities, for use by a MIP solver. Third, we generalise the implied constraint generation to cover the entire family of time-series constraints. The newly synthesised automata for time-series constraints outperform the old ones, for both the CP and MIP decompositions, and the generated implied constraints boost the inference, again for both the CP and MIP decompositions. We evaluate CP and MIP solvers on a prototypical application modelled using time-series constraints. (see CPAIOR conference paper)

7.5. Scheduling Constraint for Video Summarisation

Given a sequence of tasks T subject to precedence constraints between adjacent tasks, and given a set of fixed intervals I, the TaskIntersection (T,I,o,inter) constraint restricts the overall intersection of the tasks of T with the fixed intervals of I to be greater than or equal or less than or equal to a given limit inter. We provide a bound(Z)-consistent cost filtering algorithm wrt the starts and the ends of the tasks for the TaskIntersection constraint and evaluate the constraint on the video summarisation problem. (see CPAIOR conference paper)

7.6. A Model Seeker for Learning Constraints Models from Positive Samples

We describe a system which generates finite domain constraint models from positive example solutions (e.g. see Figure 6 giving a season schedule of the Bundesliga), for highly structured problems. The system is based on the global constraint catalog, providing the library of constraints that can be used in modeling, and the Constraint Seeker tool, which finds a ranked list of matching constraints given one or more sample call patterns (e.g. see Figure 7 giving the model learned for the input data of Figure 6). We have tested the modeler with 230 examples, ranging from 4 to 6,500 variables, using between 1 and 7,000 samples. These examples come from a variety of domains, including puzzles, sports-scheduling, packing and placement, and design theory. When comparing against manually specified canonical models for the examples, we achieve a hit rate of 50 percent, processing the complete benchmark set in less than one hour on a laptop. Surprisingly, in many cases the system finds usable candidate lists even when working with a single, positive example. (see Book chapter of Data Mining and Constraint Programming)

7.7. Global Constraint Catalog Volume II: Time-Series Constraints

First this report presents a restricted set of 22 finite transducers used to synthesise structural time-series constraints described by means of a multi-layered function composition scheme. Second it provides the corresponding synthesised catalogue of structural time-series constraints where each of the 626 constraints is explicitly described in terms of automata with accumulators, see Figure 8 for the syntesised automaton of the sum surf peak constraint. (arXiv 1609.08925)

7.8. Probabilistic Model for Binary CSP

This work introduces a probabilistic-based model for binary CSP that provides a fine grained analysis of its internal structure. Assuming that a domain modification could occur in the CSP, it shows how to express, in a predictive way, the probability that a domain value becomes inconsistent, then it express the expectation of the number of arc-inconsistent values in each domain of the constraint network. Thus, it express the expectation of the number of arc-inconsistent values for the whole constraint network. Next, it provides bounds for each of these three probabilistic indicators. Finally, a polytime algorithm, which propagates the probabilistic information, is presented. (see arXiv 1606.03894 or [19])

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
8	1	14	11	4	7	2	15	12	13	6	9	10	3	18	5	16	17
3	14	17	2	13	6	5	12	9	16	11	18	1	4	15	8	7	10
...																	
18	17	2	1	4	3	6	5	10	9	16	15	14	13	12	11	8	7
13	12	11	14	17	16	15	2	9	6	1	8	7	4	5	18	3	10
...																	

Figure 6. Input data corresponding to a flat sample (a sequence of integer values) giving a one year season schedule of the Bundesliga

-	Sequence Generator	Projection	Constraint Conjunction
1	scheme(612,34,18,34,1)	id	alldifferent*18
2	scheme(612,34,18,2,2)	id	alldifferent*153
3	scheme(612,34,18,1,18)	id	alldifferent*34
4	scheme(612,34,18,1,18)	absolute_value	symmetric_alldifferent([1..18])*34
5	scheme(612,34,18,17,1)	absolute_value	alldifferent*36
6	repart(612,34,18,34,9)	id	sum_ctr(0)*306
7	repart(612,34,18,34,9)	id	twin*1
8	repart(612,34,18,34,9)	id	elements([i,-i])*1
9	first(9,[1,3,5,7,9,11,13,15,17])	id	strictly_increasing*1
10	vector(612)	id	global_cardinality([-18.. -1-17,0-0,1..18-17])*1
11	repart(612,34,18,34,9)	id	sum_powers5_ctr(0)*306
12	repart(612,34,18,34,9)	id	sum_cubes_ctr(0)*306
13	repart(612,34,18,34,3)	sign	global_cardinality([-1-3,0-0,1-3])*102
14	scheme(612,34,18,34,1)	sign	global_cardinality([-1-17,0-0,1-17])*18
15	repart(612,34,18,17,9)	sign	global_cardinality([-1-2,0-0,1-2])*153
16	repart(612,34,18,2,9)	sign	global_cardinality([-1-17,0-0,1-17])*18
17	scheme(612,34,18,1,18)	sign	global_cardinality([-1-9,0-0,1-9])*34
18	repart(612,34,18,34,9)	sign	sum_ctr(0)*306
19	repart(612,34,18,34,9)	sign	twin*1
20	repart(612,34,18,34,9)	absolute_value	twin*1
21	repart(612,34,18,34,9)	sign	elements([i,-i])*1
22	scheme(612,34,18,34,1)	sign	among_seq(3,[-1])*18
23	repart(612,34,18,34,9)	absolute_value	elements([i,i])*1
24	first(9,[1,3,5,7,9,11,13,15,17])	absolute_value	strictly_increasing*1
25	first(6,[1,4,7,10,13,16])	absolute_value	strictly_increasing*1
26	scheme(612,34,18,34,1)	absolute_value	nvalue(17)*18

Figure 7. Model, i.e. conjunction of global constraints, learned from the single flat sample

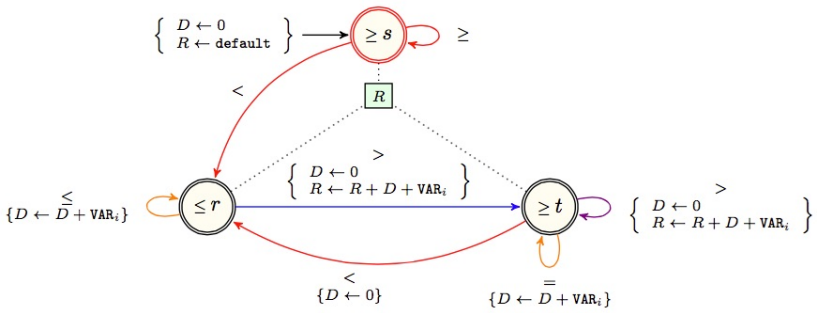

Figure 8. Synthesised automaton with accumulator of the sum surf peak constraint obtained from the transducer of the peak pattern

7.9. Estimating parallel runtimes for randomized algorithms in constraint solving

We present a detailed analysis of the scalability and parallelisation of Local Search algorithms for constraint-based and SAT (Boolean satisfiability) solvers. We propose a framework to estimate the parallel performance of a given algorithm by analyzing the runtime behavior of its sequential version. Indeed, by approximating the runtime distribution of the sequential process with statistical methods, the runtime behavior of the parallel process can be predicted by a model based on order statistics. We apply this approach to study the parallel performance of a constraint-based Local Search solver (Adaptive Search), two SAT Local Search solvers (namely Sparrow and CCASAT), and a propagation-based constraint solver (Gecode, with a random labeling heuristic). We compare the performance predicted by our model to actual parallel implementations of those methods using up to 384 processes. We show that the model is accurate and predicts performance close to the empirical data. Moreover, as we study different types of problems, we observe that the experimented solvers exhibit different behaviors and that their runtime distributions can be approximated by two types of distributions: exponential (shifted and non-shifted) and lognormal. Our results show that the proposed framework estimates the runtime of the parallel algorithm with an average discrepancy of 21 percent w.r.t. the empirical data across all the experiments with the maximum allowed number of processors for each technique. (see Journal of Heuristics)

7.10. ghost: A Combinatorial Optimization Framework for Real-Time Problems

We presents GHOST, a combinatorial optimization framework that a real-time strategy (RTS) AI developer can use to model and solve any problem encoded as a constraint satisfaction/optimization problem (CSP/COP). We show a way to model three different problems as a CSP/COP, using instances from the RTS game StarCraft as test beds. Each problem belongs to a specific level of abstraction (the target selection as reactive control problem, the wall-in as a tactics problem, and the build order planning as a strategy problem). In our experiments, GHOST shows good results computed within some tens of milliseconds. We also show that GHOST outperforms state-of-the-art constraint solvers, matching them on the resources allocation problem, a common combinatorial optimization problem. (see IEEE Transactions on Computational Intelligence and AI in games journal)

7.11. TorchCraft: a Library for Machine Learning Research on Real-Time Strategy Games

We present TorchCraft, a library that enables deep learning research on Real-Time Strategy (RTS) games such as StarCraft: Brood War, by making it easier to control these games from a machine learning framework, here Torch. This white paper argues for using RTS games as a benchmark for AI research, and describes the design and components of TorchCraft. (see arXiv 1611.00625)

7.12. POSL: A Parallel-Oriented metaheuristic-based Solver Language

For a couple of years, all processors in modern machines are multi-core. Massively parallel architectures, so far reserved for super-computers, become now available to a broad public through hardware like the Xeon Phi or GPU cards. This architecture strategy has been commonly adopted by processor manufacturers, al- lowing them to stick with Moore's law. However, this new architecture implies new ways to design and implement algorithms to exploit its full potential. This is in particular true for constraint-based solvers dealing with combinatorial optimiza- tion problems. Here we propose a Parallel-Oriented Solver Language (POSL, pronounced "puzzle"), a new framework to build interconnected meta-heuristic based solvers working in parallel. The novelty of this approach lies in looking at solver as a set of components with specific goals, written in a parallel-oriented language based on operators. A major feature in POSL is the possibility to share not only information, but also behaviors, allowing solver modifications during runtime. Our framework has been designed to easily build constraint-based solvers and reduce the developing effort in the context of parallel architecture. POSL's main advan- tage is to allow solver designers to quickly test different heuristics and parallel com- munication strategies to solve combinatorial optimization problems, usually time- consuming and very complex technically, requiring a lot of engineering.

7.13. Towards Automated Strategies in Satisfiability Modulo Theory

SMT solvers include many heuristic components in order to ease the theorem proving process for different logics and problems. Handling these heuristics is a non-trivial task requiring specific knowledge of many theories that even a SMT solver developer may be unaware of. This is the first barrier to break in order to allow end-users to control heuristics aspects of any SMT solver and to successfully build a strategy for their own purposes. We present a first attempt for generating an automatic selection of heuristics in order to improve SMT solver efficiency and to allow end-users to take better advantage of solvers when unknown problems are faced. Evidence of improvement is shown and the basis for future works with evolutionary and/or learning-based algorithms are raised (see Genetic Programming conference paper).

7.14. Using CP for the Urban Transit Crew Rescheduling Problem

Scheduling urban and trans-urban transportation is an important issue for industrial societies. The Urban Transit Crew Scheduling Problem is one of the most important optimization problem related to this issue. It mainly relies on scheduling bus drivers workday respecting both collective agreements (see Figure 9 for an example of regulation rule) and the bus schedule needs. If this problem has been intensively studied from a tactical point of view, its operational aspect has been neglected while the problem becomes more and more complex and more and more prone to disruptions. In this way, this paper presents how the constraint programming technologies are able to recover the tactical plans at the operational level in order to efficiently help in answering regulation needs after disruptions (see CP conference paper).

7.15. Traveling salesman and the tree: the importance of search in CP

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 paper 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. (see Constraints journal)

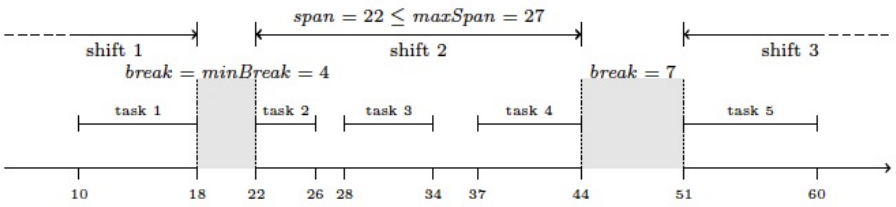

Figure 9. Illustration of typical regulation rule in the Labcom project; Shift 2 of an employee is composed of tasks 2, 3, and 4. It is between shifts 1 and 3 of the same employee. A break of duration 4 is scheduled between task 1 of shift 1 and task 2 of shift 2, because the gap between these tasks is at least a break of 4. Similarly, a break of duration 7 is scheduled after task 4 of shift 2 and task 5 of shift 3. No other breaks can be scheduled between the tasks of shift 2 because of the minimum break duration. The span of shift 2 is 22 and does not exceed 27: it is composed of tasks 2, 3, and 4, as well as of the two gaps between these tasks.

7.16. Event Selection Rules to Compute Explanations

Explanations have been introduced in the previous century. Their interest in reducing the search space is no longer questioned. Yet, their efficient implementation into CSP solver is still a challenge. In this paper, we introduce ESeR, an Event Selection Rules algorithm that filters events generated during propagation. This dynamic selection enables an efficient computation of explanations for intelligent backtracking algorithms. We show the effectiveness of our approach on the instances of the last three MiniZinc challenges. (see arXiv 1608.08015 or [20])

7.17. Towards energy-proportional Clouds partially powered by renewable energy

With the emergence of the Future Internet and the dawning of new IT models such as cloud computing, the usage of data centers (DC), and consequently their power consumption, increase dramatically. Besides the ecological impact, the energy consumption is a predominant criterion for DC providers since it determines the daily cost of their infrastructure. As a consequence, power management becomes one of the main challenges for DC infrastructures and more generally for large-scale-distributed systems. In this paper, we present the EpoCloud prototype, from hardware to middleware layers. This prototype aims at optimizing the energy consumption of mono-site Cloud DCs connected to the regular electrical grid and to renewable-energy sources (see Journal of Computing).

AOSTE Project-Team

6. New Results

6.1. CCSL as a Logical Clock Calculus Algebra: expressiveness and analysis techniques

Participants: Robert de Simone, Julien Deantoni, Frédéric Mallet, Dongdong An.

CCSL is a simple, half-declarative and half-imperative language describing relations and constraints between sequences of events considered as Logical Clocks. The usage of CCSL for specification of embedded systems is powerful in that it defers the precise setting of physical timing until later implementation design phases (which may vary according to circumstances), see 3.2 .

Early this year we established the universal recursive expressivity of CCSL, by encoding the dynamics of Petri Nets with inhibitor arcs in our framework (still unpublished). Those results were presented by Robert de Simone in a keynote talk at Memocode 2016. This result prompts the use of non-automatic methods for establishing actual schedules as solutions of CCSL specifications seen as schedulability constraints. Steps in that direction were made in [37].

We also considered the extension of CCSL towards stochastic modeling of potential input clocks as my emerge from the Cyber-Physical world (mixing probabilistic modeling of external events with discrete transformations by discrete cyber digital controlers). This work was initiated in [28], and should be further extended in the ongoing PhD thesis of Dongdong An.

Finally, we have also investigated to decide on specific schedules (e.g. periodic schedules) valid for a subset of CCSL. We have established a sufficient static condition for the existence of such a periodic schedule as well as a practical implementation to build such a solution [39] based on a SMT solver.

6.2. Industrial design flow for Embedded System Engineering

Participants: Julien Deantoni, Frédéric Mallet, Marie Agnes Peraldi Frati, Robert de Simone, Hui Zhao, Ales Mishchenko.

As part of the PIA LEOC Clarity collaborative project we considered the inytroduction of formal methods into a high-level model-based design environment for embedded systems, named CAPELLA (https://polarsys.org/capella/). CAPELLA is part of the Polarsys Eclipse project. It originates from Thales, and is currently being deployed in real operational divisions in a number of companies.

Our activities consisted in demonstrating how the theoretical models of Logical Time and derives Models of Computation could be used to give precise semantics and provide simulation benefits, when applied to the modeling paradigms used in CAPELLA and advanced in Clarity. In particular we focused on the connection between timing/performance properties and other kinds of non-functional properties, including model variability.

This year we focused on two mains tasks:

First, we clarified and extended the notion of Modes and States in the Capella system engineering language. Specifically, a specific diagram has been introduced to deal with the system modes. The notion of mode is then used to specify different configurations of the system, mainly in terms of the active functions, their data dependencies, their deployment on the logical and physical architecture as well as the scenario to be verified in this specific mode. In consequence, the behavioural semantics of the mode diagram strongly interacts with the behavioral semantics of the other diagrams. The execution semantics was given by promoting our contributions in GEMOC and BCOoL (see 6.3).

Second, Capella proposes a consistent multi-view approach accross different engineering domains. At some step in the refinement process, these different views are extracted to a domain specific tool (like Simulink for instance). It is then required 1) to verify that the manipulation done in the domaine specific tool respect the original semantics expected by the architect, and 2) to understand the impact of the decisions made in domain specific tools on the interaction with the other views. To do so we provided a generic aproach to confront the race to the behavioral semantics we formally defined in Capella. We are curretly working on a theoretical approach to improve the overal performance of such approach.

While BCOoL and Gemoc only considers discrete models, the PhD thesis of Hui Zhao, which started in March 2016, explores a possible extension that specifically targets Cyber-Physical Systems where we different timed models combined, including both discrete and dense timed models. In this thesis, we also explore the impact of such an heterogeneous modeling framework to guarantee security and safety properties of the combined models. This is done in collaboration with Ludovic Apvrille (who is co-advisor of the thesis) from Telecom ParisTech.

6.3. Coordination of heterogeneous Models of Computation as Domain-Specific Languages

Participants: Matias Vara Larsen, Julien Deantoni, Frédéric Mallet.

Our work this on coordination of heterogeneous languages produced two major results. The first one is the development of BCOoL (Behavioral Coordination Operator Language. BCOoL is a language dedicated to the specification of coordination patterns between heterogeneous languages. It comes with a tool chain allowing the generation of the coordination given a BCOoL operator and specific models. Our second result is the development of an heterogeneous execution engine, integrated to Gemoc studio, to run conjointly different models. Both works re extensively reported in Matias Vara Larsen PhD thesis [19].

6.4. SoC multiview (meta)modeling for performance, power, and thermal aspects

Participants: Amani Khecharem, Robert de Simone, Emilien Kofman, Julien Deantoni.

In the framework of the ANR HOPE project we progressed the definition of multiview metamodels for the design of Systems-on-Chip) (SoC systems integrating performance, power and thermal aspects. The main concern was to stress regularity and commonalitbetween those views, each developed on "domains" defined as partitions of the original block diagram (clock domains, voltage domains, floorplans,...), and with finite state machine controlers setting the levels of these domains; links between distinct views are originally provided by laws of physics, but then usually identified with discrete allowed values(such as OPP, Operating Performance Points, providing the available frequency-voltage levels for processor clocks).

The corresponding methodology, named MuArch, was reported as Ameni Khacharem PhD document [16].

6.5. MoCs and novel architectures

Participants: Amine Oueslati, Robert de Simone, Albert Savary, Emilien Kofman.

In the context of the FUI Clistine project we considered the links betwen formal Models of Computation and parallel programming models (MPI mainly). The objective is to figure to what level an abstraction of MPI processes as concuurent communicating processes can help for the AAA design process being applied to theselection of adequate MPI communications. This topic reflects the ongoing PhD thesis of Amine Oueslati, and the engineering work of Albert Savary in the first semester.

6.6. Solving AAA constraints analytically

Participants: Emilien Kofman, Dumitru Potop Butucaru, Robert de Simone, Amine Oueslati.

We experimented on the use of SMT solvers to compute efficient mappings (bothe schedules and placement allocations) for concurrent embedded applications onto specific embedded architectures of big.LITTLE features (where alocation and migration of tasks can follow concern for low-power consumption). In fact, the work consisted greatly in a study of how the various models could be encoded to scale up, allowing the solvesr to provide results in reasonalble time. The results have been presented [41], [31], and will soon appear as E. Kofman PhD thesis.

6.7. Coupling SystemC and FMI for co-simulation of Cyber-Physical Systems

Participants: Stefano Centomo, Julien Deantoni, Robert de Simone.

In collaboration with Professor Davide Quaglia, from the University of Verona, we are studying the proper joint modeling of interactions between different domains involved in a cyber-physical system (CPS), and specifically between the cyber and physical parts. In our first work, realized in the context of Stefano Centomo master internship, we investigated how an event based hardware description language can be used in an emerging industry standard for co-simulation (FMI/FMU developed orginally in a Modelica framework). Preliminary results were published [26], and we hope to start a PhD as follow-up of these results.

6.8. Behavioural Semantics if Open pNets

Participants: Eric Madelaine, Ludovic Henrio, Siqi Li, Min Zhang.

We have extended our preliminary work on Parameterised Networks of Automata (pNets), by looking at the behavioural semantics and at bisimulation equivalences for open pNet systems. These can be used to encode operators of various process algebras, construct of distributed or reactive system programming languages, or even parallel algorithmic skeletons, and generic distributed algorithms. As a first step, we studied the properties of a strong bisimulation equivalence based on logical hypotheses about the behaviour of process variables in the open systems. This has been published in [22], [33] and an extended version as an Inria research report [43]. We are now implementing algorithms for computing the symbolic behavioural semantics of open pNets, and checking strong bisimulation, using a SAT engine for reasoning on the hypotheses.

In order to understand better this behavioural semantics, we also have defined another version with a denotational flavour, namely using a “Universal Theory of Processes (UTP)” style. There we express the communication actions of pNets using traces of interaction events, and we were able to prove axiomatic properties of some simple (open) pNets. This was published in [32]. In the long term, it could be interesting to study the relations between the FH-bisimulation and the UTP semantics, relating both behavioural, denotational and algebraic semantics of pNets.

6.9. Behavioural semantics for GCM components

Participants: Ludovic Henrio, Oleksandra Kulankhina, Eric Madelaine.

With Ludovic Henrio (Comred/I3S) and Rabea Ameur-Boulifa (Labsoc/Telecom-Paristech), we have pursued our research on the Behavioural semantics, in terms of pNets, of the core concepts of Grid Component Model (GCM). The results are currently submitted for publication as a journal paper, under revision.

6.10. Performance analysis and optimisation of an HPC scientific application

Participants: Luis Agustin Nieto, Sid Touati.

In the context of the internatinal Internship of Luis Agustin Nieto we conducted a large-scale experiment of source code optimization for HPC application. This work is meant to identify potential approaches that may be automatized in the future. The current use case was an application named CONVIV. CONVIV is a computer code implementing the VMFCI Method to solve the stationary Schrödinger equation for a set of distinguishable degrees of freedom (https://svn.oca.eu/trac/conviv). It is used in Chemistry for computing the energy levels of molecules.

This application is very computer-intensive (many hours of computation on a high performance grid computer). We have been given its source code (fortran with OpenMP), and we have been asked to analyse its performance and to optimise its execution time.

We did an extensive set of experiments for this application on many computers, and mainly on the `cicada.unice.fr` shared grid computer used for scientific parallel computing at UNS). We varied many parameters in our experiments:

- The number of threads was 2, 4, 6, 8, 16 threads. We also analysed the sequential code version.
- The thread affinity strategies for scheduling were: none (linux scheduler), scatter, compact.
- We repeated each experience 35 times to analyse performance stability.
- We used 2 compilers (gfortran, ifort) with -O3.
- We did a precise performance profiling using the Intel Vtune tool.

During our experiments we observed that, even with all the parameters above kept fixed, repeating the executions 35 times shows grat variability between best and worst execution times (more than double in some cases). The critical-path functions remained the same for each configuration choice, including in particular specific matrix computation functions.

After investigation and experiments, we succeed in getting a spectacular performance improvement by applying the following optimisations:

- Replace one of the matrix computation function by an MKL one (highly optimised and tuned function done by Intel).
- Use the compact thread scheduling strategy (OpenMP parameter).
- By using gfortran compiler with -O3, we reduced the execution time from 18400 seconds to 820 seconds (speedup=22).
- By using the ifort compiler with -03, we reduced the execution time from 21000 seconds to 620 seconds (speedup=33).

6.11. Formal translation validation of multi-processor real-time schedules

Participants: Keryan Didier, Dumitru Potop-Butucaru.

This research direction is mainly represented by the PhD thesis of Keryan Didier, and takes place in the framework of the ITEA3 ASSUME project. The technical focus of the ASSUME project is on formal compiler verification and on correct real-time implementation for parallel applications. The objective of this PhD thesis is to formally prove the correctness of (part of) the automatic code generation technology of Lopht, considering the respect of non-functional requirements, and in particular real-time requirements such as release dates, deadlines and periods.

During this first year of work we have:

1. Simplified the allocation and scheduling algorithms of Lopht to facilitate proof while still being able to handle the industrial use case. The resulting algorithms consider all the aspects pertaining to functional specification and non-functional requirements, but make simplifying assumptions on the execution platform (by not taking into account memory access interferences during parallel execution).
2. Developed a formally proved translation validation tool to determine the correctness of schedules produced by the algorithms at point (1). The tool is developed and proved in Coq. Coq code extraction is used to produce OCaML code that integrates in the allocation and scheduling flow.
3. Evaluated the tool on a large-scale industrial use case from Airbus (6000 Scade nodes). We demonstrated the tool to our project partners and during the ASSUME project evaluation. This evaluation showed that our scheduling and formally proved validation tools scale up to the size of large applications.

The main limitation of the current work is that it does not take into account the interferences due to concurrent memory accesses. This gives the main research direction for the next year.

We are currently writing a paper on this subject.

6.12. Lopht back-end for TTEthernet-based distributed systems

Participants: Raul Gorcitz, Dumitru Potop-Butucaru.

The global objective of this activity is a large-scale, ongoing effort to assess the possibility of automatically synthesizing full real-time implementations, including the so-called "bus frame" (the network configuration) on complex industrial platforms and for complex functional and non-functional specifications. We worked this year in the context of the post-doctoral position of Raul Gorcitz, funded by the ITEA3 ASSUME project, but also in the framework of our collaboration with CNES and Airbus DS.

The chosen platform was an industry-level evaluation platform using several Single-Board Computers (SBCs) running the VxWorks 653 OS, and connected through a Time-Triggered Ethernet (TTE) network. This platform was provided by CNES, as typical target for embedded applications. TTE is a standardized commmercial communication network, on top of a switched Ethernet basis, ommmercialized by TTTech. TTE adds support for realtime and fault tolerant communications, allows multiple communications of mixed criticalities to share a single physical medium. This is ensured by means of dedicated hardware using a set of configuration files describing the system architecture and behavior. These configurations are synthesized by the proprietary TTEplan tool starting from a global network description file.

The main scientific difficulty was the formal modeling of the behavior of the TTE network, followed by the extension of scheduling algorithms to consider such a network. While preliminary results were obtained and published last year, we completed and demonstrated this work to our industrial partners, and we are currently writing a second paper on the subject.

6.13. Uniprocessor Real-Time Scheduling

Participants: Mehdi Mezouak, Yves Sorel, Walid Talaboulma.

In the context of the master internship of Mehdi Mezouak, we thoroughly tested the offline time triggered scheduler implemented on an ARM Cortex M4 last year. We remind that this scheduler, intended for safety critical applications, uses a scheduling table containing the instants when the scheduler will be called through interruptions triggered by a timer. This table is generated by a uniprocessor offline schedulability analysis which accounts accurately for the scheduler cost itself, and for the cost of all preemptions the data dependent tasks are subjected to. This approach allows accounting for preemptions induced by the cost of other preemptions. We implemented a time measurement system on a LPC4080 microcontroller board of NXP which includes the ARM Cortex M4 and several timers, to determine on the one hand the actual cost of the scheduler and the cost of one preemption, and on the other hand start, resume and completion times of every task of the task sets. For the ARM Cortex M4 with a 120Mhz clock we obtained 142 cycles (2.3 μs) for the scheduler cost and 54 cycles (0.9 μs) for the cost of one preemption. We used these values for schedulability analyses we applied to various task sets. We improved the graphical tools proposed last year to draw the timing diagrams obtained during the schedulability analysis and during the real-time execution of the task set in order to compare them. For example, thanks to these measurement system and tools, we showed that this scheduler, based on a non periodic timer rather than the usual periodic one, allows the periodic execution of tasks without any jitter.

6.14. Multiprocessor Real-Time Scheduling

Participants: Mehdi Mezouak, Salah Eddine Saidi, Yves Sorel.

Always in the context of the master internship of Mehdi Mezouak, we studied the extension to multiprocessor of our offline time triggered scheduler. Since we chose the partitioned multiprocessor scheduling approach rather than the global one which is not suited to safety critical applications due to the prohibitive cost of task migrations, the uniprocessor schedulability analysis is easily extended. Indeed, the main modification consists, for every processor, in accounting for the cost of inter-processor communications and synchronizations due to data dependences when a producer task is allocated to a processor which is different from the one the corresponding consumer task is allocated to. Therefore, new scheduler calls are added to the scheduling table corresponding to instants when awaited data are available, i.e. produced and then transfered. Of course, there are as many scheduling tables, and thus schedulers, as there are processors, and these scheduling tables are supposed to share a unique global time. The implementation of this global time raises a complex problem since it is not possible to dispatch a unique physical clock to all the processors. Among various solutions, we chose to use a physical clock rather than a logical one like in the Lamport's timestamp approach since we are interested in safety critical real-time. In addition, we chose the Berkeley's algorithm based on a master-slave approach where the clock server is maintained by one of the processor of the multiprocessor. This algorithm is more robust to failures than other algorithms based on an external clock server. Finally, using the measurement system mentioned previously, we measured accurately the cost of inter-processor communications according to the number of transfered data, in the case of an ethernet network that we experimented last year to connect several LPC4080 microcontroller boards.
During the second year of the PhD thesis of Salah Eddine Saidi, we continued to study the parallelization on multi-core of FMI-based co-simulation of numerical models, that is increasingly used for the design of Cyber-Physical Systems. Such model developed according to the FMI standard is defined by a number of C functions, called "operations", for computing its variables (inputs, outputs, state) and data dependences between these variables. Each model has an associated integration step and exchanges data with the other models according to its communication step which can be larger or equal to its integration step. These models are represented by a dataflow graph of operations [35] that is compliant with the conditioned repetitive dataflow model of our AAA methodology for functional specification. Our work mainly focused on two aspects. First, we proposed a graph transformation algorithm in order to allow handling multi-rate co-simulation, i.e. where connected models have different communication steps. This algorithm is based on the concept of graph unfolding similarly to the unrolling algorithm of our AAA methodology. The new graph is represented over the hyper-step which is equal to the least common multiple of the communication steps of all the models. Each operation is repeated in the graph according to the ratio between the hyper-step and its communication step. Then, rather than adding edges connecting all the repetitions of dependent operations, specific rules are used to define the repetitions that have to be connected by edges. These rules ensure correct data exchange between the operations as requested in the context of simulation. Second, some FMI functions called to compute model variables may not be "thread-safe", i.e. they cannot be executed in parallel as they may share some resource (e.g. variables). Consequently, if two or more operations belonging to the same model are executed on different cores, a mechanism that ensures these operations are executed in strictly disjoint time intervals must be set up. We proposed an acyclic orientation heuristic to solve this problem. This heuristic adds non directed edges between the operations that belong to the same model, and then assigns directions to these edges with the aim of minimizing the critical path of the resulting graph and subject to the constraint that no cycle is generated in the graph.

6.15. Probabilistic Solutions for Hard Real-Time Systems

Participants: Adriana Gogonel, Dorin Maxim, Antoine Bertout, Tomasz Kloda, Irina Asavoae, Mihail Asavoae, Cristian Maxim, Walid Talaboulma, Slim Ben-Amor, Robert Davis, Liliana Cucu.

The probabilistic solutions for hard real-time systems are built under the hypothesis that worst case values and worst case execution scenarios have extremely low probability of appearance. While continuing the estimation of bounds for the worst case execution times of a program [34], [25], we have proposed the first utilisation of probabilistic description for mixed-criticality systems [42]. Our result is exploiting the heavy tails of the execution times of a program to propose efficient scheduling solutions. Moreover since the feasibility intervals [21] for a probabilistic real-time system is not formally identified, we have formulated the first feasibility reasoning for such systems [47] under fixed-priority assignment policies [20]. Another important problem

for probabilistic real-time systems concerns the feasibility in presence of precedence constraints, often used by our industry partners. The introduction of precedence constraints requires the comparison of probabilistic arrivals and we showed that existing measures are not correct in this context and we proposed and proved correct new measures [24].

CONVECS Project-Team

6. New Results

6.1. New Formal Languages and their Implementations

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.

6.1.1. Translation from LNT to LOTOS

Participants: Hubert Garavel, Frédéric Lang, Wendelin Serwe.

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.

In 2016, the long-term effort to enhance the LNT language and its tools has been pursued. LNT has been enriched with a new statement "use X" that suppresses compiler warnings when a variable X is assigned but not used. The syntax of LNT expressions has been modified so that field selections ("E.X"), field updates ("E1.X = E2"), and array accesses ("E1 [E2]") can now be freely combined without extra parentheses. LNT programs can now import predefined libraries, and two such libraries (BIT.lnt and OCTET.lnt) have been introduced.

A move towards "safer" LNT exceptions has started. The syntax for exceptions in function declarations has been modified and the semantics of LNT has shifted from "unchecked" to "checked" exceptions: exception parameters, if any, must be explicitly mentioned when a function is called. Such exception parameters can now be passed using either the named style or the positional style.

A few static-semantics constraints have been relaxed; for instance, it is no longer required that actual gate parameters be different when calling a process. Various bugs have been fixed. Several error/warning messages have been made more precise, and the format of LNT error/warning messages has been aligned on that of GCC. Finally, the LNT2LOTOS Reference Manual has been updated and enhanced.

6.1.2. Translation from LOTOS NT to C

Participants: Hubert Garavel, Sai Srikar Kasi, Wendelin Serwe.

The TRAIAN compiler is used to build many compilers and translators of the CADP toolbox. This compiler itself is built using the FNC-2 compiler generator that, unfortunately, is no longer available. For this reason, TRAIAN only exists in 32-bit version, and sometimes hits the 3-4 GByte RAM limit when dealing with complex compilers such as LNT2LOTOS.

In 2016 we addressed this issue, in several steps. As a first step, we released a stable version 2.8 of TRAIAN. Then, we gathered all programs written in LOTOS NT, the input language of TRAIAN, and organized them in non-regression test bases. We entirely scrutinized the source code of TRAIAN, which consists in a large collection of attribute grammars, deleting all parts of code corresponding to those features of the LOTOS NT language that were either not fully implemented or seldom used in practice. This reduced the source code of TRAIAN by 40% and divided by two the size of TRAIAN executables. A few other bugs have been fixed and the reference manual of TRAIAN was entirely revised and updated.

In parallel, we undertook a complete rewrite of TRAIAN to get rid of the FNC-2 deprecated attribute grammar tool. We developed lexical and syntactic descriptions of the input language using the SYNTAX compiler-generation system developed at Inria Paris. The syntax tree of LOTOS NT and the library of predefined LOTOS NT types and functions are now themselves defined in LOTOS NT, as we plan to follow a bootstraping approach, using the current version of TRAIAN to build the next one. To this aim, a large fraction of the TRAIAN attribute grammars has been rewritten in LOTOS NT.

6.1.3. 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 2016, following the writing of the new CADP manual page for LOTOS, the common front-end of CAESAR and CAESAR.ADT was carefully inspected, which led to various bug fixes regarding type signatures, error messages for overloaded functions, renaming/actualization of sorts and operations, equations for renamed operations, C-language reserved keywords, implementation of numeral sorts, and iterators over these sorts. Another bug was fixed for the "`-external`" option of CAESAR and a new "`-numeral`" option was introduced. Also, the C identifiers automatically generated by CAESAR.ADT for sorts, operations, tester and selector macros have been simplified; as the new conventions are not backward compatible, migration tools were developed to ease transitioning the existing LOTOS and C files.

6.1.4. NUPN

Participant: Hubert Garavel.

Nested-Unit Petri Nets (NUPNs) is an upward-compatible extension of P/T nets, which are enriched with structural information on their concurrent structure. Such additional information can easily be produced when NUPNs are generated from higher-level specifications (e.g., process calculi); quite often, such information allows logarithmic reductions in the number of bits required to represent states, thus enabling verification tools to perform better. The principles of NUPNs are exposed in [33] and its PNML representation is described here [0].

In 2016, the NUPN principles have been presented in an invited talk at D-CON, the German national conference on concurrency theory. The collection of NUPN models used for experimentation has been enlarged and reorganized; it now contains more than $10,000$ models. A new beta-version of the VLPN (*Very Large Petri Nets*) benchmark suite, which contains 350 large models has been produced. Also, new prototype tools have been developed that try to convert P/T nets into NUPNs, which requires to automatically infer the concurrent structure of flat, unstructured nets.

The CAESAR.BDD tool that analyzes NUPN models and serves to prepare the yearly Model Checking Contest [0] has been enhanced with two new options "`-initial-places`" and "`-initial-tokens`". It now properly handles the case where the initial marking contains more than 2^{31} tokens. The output of the "`-mcc`" option has been made more precise when the NUPN under study is conservative or sub-conservative.

6.1.5. Translation from BPMN to LNT

Participants: Gwen Salaün, Ajay Muroor-Nadumane.

Evolution has become a central concern in software development and in particular in business processes, which support the modeling and the implementation of software as workflows of local and inter-process activities. We advocate that business process evolution can be formally analyzed in order to compare different versions of processes, identify precisely the differences between them, and ensure the desired consistency.

In collaboration with Pascal Poizat (LIP6, Paris), we worked on checking the evolution of BPMN processes. To promote its adoption by business process designers, we have implemented it in a tool, VBPMN, that can be used through a Web application. We have defined different kinds of atomic evolutions that can be combined and formally verified. We have defined a BPMN to LNT model transformation, which, using the LTS operational semantics of LNT, enables us to automate our approach using existing LTS model checking and equivalence checking tools, such as those provided by CADP. We have applied our approach to many examples for evaluation purposes. These results have been published in an international conference [23].

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

6.1.6. Translation from GRL to LNT

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

In the context of the Bluesky project (see § 8.2.2.1), 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. A translator named GRL2LNT has been developed, allowing an LNT program to be generated from a GRL specification automatically. Additionally, an OPEN/CAESAR-compliant compiler named GRL.OPEN (based on GRL2LNT and LNT.OPEN) enables the on-the-fly exploration of the LTS underlying a GRL specification using CADP.

In 2016, a new version 3.3 of the GRL2LNT translator has been released, with an improved LNT code generation exploiting the "`use`" construct newly added to LNT. Also, a non-regression test base containing hundreds of GRL specifications has been set up. This also contributes to the non-regression testing of the compilation chain for LNT by providing new LNT descriptions generated automatically by GRL2LNT.

An overview paper about GRL and its translation to LNT was published in an international journal [14]. The complete definition of GRL and its applications to GALS systems are available in F. Jebali's PhD thesis [44].

6.1.7. Translation of Term Rewrite Systems

Participants: Hubert Garavel, Lina Marsso, Mohammad-Ali Tabikh.

In 2016, we pursued the development undertaken in 2015 of a software platform for systematically comparing the performance of rewrite engines and pattern-matching implementations in algebraic specification and functional programming languages. Our platform reuses the benchmarks of the three Rewrite Engine Competitions (2006, 2009, and 2010). Such benchmarks are term-rewrite systems expressed in a simple formalism named REC, for which we developed automated translators that convert REC benchmarks into various languages.

In 2016, we corrected a number of benchmarks and added many new ones, to reach a total of 85 benchmarks in December 2016. Among these new benchmarks, one can mention a formalization of arithmetic operations on signed integers, a collection of (8-bit, 16-bit, and 32-bit) binary adders and multipliers, and a complete model of the MAA ("Message Authenticator Algorithm"), a Message Authentication Code used for financial transactions (ISO 8731-2) between 1987 and 2002.

The existing translators (for Haskell, LOTOS, Maude, mCRL, OCAML, Opal, Rascal, Scala, SML-NJ, and Tom) have been enhanced and new translators (for AProVE, Clean, LNT, MLTON, Stratego/XT) have been developed. Tools for automatically extracting and synthesizing performance statistics have also been developed.

6.2. Parallel and Distributed Verification

6.2.1. Distributed State Space Manipulation

Participants: Hubert Garavel, Hugues Evrard, Wendelin Serwe.

For distributed verification, CADP provides the PBG format, which implements the theoretical concept of *Partitioned LTS* [38] and provides a unified access to an LTS distributed over a set of remote machines.

In 2016, the code of the CAESAR_NETWORK_1 library, which is a building block for the distributed verification tools of CADP, has been carefully scrutinized and split into logically-independent modules. Nine problems have been detected and solved, among which a flaw in the distributed termination algorithm: the entire network could freeze if a worker process crashed too early, before the opening of TCP sockets. From now on, a better distributed termination algorithm is used, which supports the coexistence of several networks, ensures that all connections are closed before terminating, and produces more informative traces indicating which worker has triggered termination. Also, the improved CAESAR_NETWORK_1 library checks that all workers operate in directories that are pairwise distinct, mutually disjoint, and different from the working directory of the coordinator process.

6.2.2. Distributed Code Generation for LNT

Participants: Hugues Evrard, Frédéric Lang, Wendelin Serwe.

Rigorous development and prototyping of a distributed algorithm using 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. 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 2016, we improved the user documentation of the DLC distribution, and added support for structured data types, enabling experiments of DLC on the LNT model of the CAESAR_SOLVE_2 library (see § 6.2.3). An overview paper about DLC has been accepted in an international journal [12].

6.2.3. Distributed Resolution of Boolean Equation Systems

Participant: Wendelin Serwe.

The BES_SOLVE tool of CADP enables to solve BESs (*Boolean Equation Systems*) using the various resolution algorithms provided by the CAESAR_SOLVE library (see 5.1), including a distributed on-the-fly resolution algorithm described in pseudo-code in [45].

In 2016, we modeled the pseudo-code of the distributed resolution algorithm in LNT (about $1,000$ lines). For a set of BES examples (encoded as LNT data types and functions), we experimented the generation of the LTS corresponding to the distributed resolution algorithm applied to each BES. We also experimented with the DLC tool [21] to generate a prototype distributed implementation of the resolution algorithm from its LNT specification. These experiments uncovered some errors in the original pseudo-code.

We also simplified the C implementation included in the BES_SOLVE tool to closer match the corrected LNT model, mainly by removing additional synchronization messages. We started to evaluate the simplified implementation using our non-regression test base (more than $15,000$ BESs), with promising results.

6.2.4. Stability of Communicating Systems

Participant: Gwen Salaün.

Analyzing systems communicating 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 imposing an upper bound for the size of communication buffers.

In 2016, our focus was on systems that are likely to be unbounded and therefore result in infinite systems. We did not want to restrict the system by imposing any arbitrary bound. We introduced a notion of stability and proved that once the system is stable for a specific buffer bound, it remains stable whatever larger bounds are chosen for buffers. This enables one to check certain properties on the system for that bound and to ensure that the system will preserve them for arbitrarily larger buffer bounds. We also proved that computing this bound is undecidable but we showed how we succeed in computing these bounds for many examples using heuristics and equivalence checking. These results have been published in an international conference [18].

In collaboration with Carlos Canal (University of Málaga, Spain), we have also shown how the stability approach can be used for composition and adaptation of component-based software. This led to a publication in an international conference [20].

6.2.5. Debugging of Concurrent Systems

Participants: Gianluca Barbon, Gwen Salaün.

Model checking is an established technique for automatically verifying that a model satisfies a given temporal property. When the model violates the property, the model checker returns a counterexample, which is a sequence of actions leading to a state where the property is not satisfied. Understanding this counterexample for debugging the specification is a complicated task for several reasons: (i) the counterexample can contain hundreds of actions, (ii) the debugging task is mostly achieved manually, and (iii) the counterexample does not give any clue on the state of the system (e.g., parallelism or data expressions) when the error occurs.

In 2016, we proposed a new approach that improves the usability of model checking by simplifying the comprehension of counterexamples. Our solution aims at keeping only actions in counterexamples that are relevant for debugging purposes. To do so, we first extract in the model all the counterexamples. Second, we define an analysis algorithm that identifies actions that makes the behaviour skip from incorrect to correct behaviours, making these actions relevant from a debugging perspective. Our approach is fully automated by a tool that we implemented and applied on real-world case studies from various application areas for evaluation purposes. A paper presenting these results has been accepted at an international conference.

6.3. Timed, Probabilistic, and Stochastic Extensions

6.3.1. On-the-fly Model Checking for Extended Regular Probabilistic Operators

Participant: Radu Mateescu.

In the context of the SENSATION project (see § 8.3.1.1), we study the specification and verification of quantitative properties of concurrent systems, which requires expressive and user-friendly property languages combining temporal, data-handling, and quantitative aspects.

In 2016, in collaboration with José Ignacio Requeno (Univ. Zaragoza, Spain), we aimed at facilitating the quantitative analysis of systems modeled as PTSs (Probabilistic Transition Systems) labeled by actions containing data values and probabilities. We proposed a new regular probabilistic operator that computes the probability measure of a path specified by a generalized regular formula involving arbitrary computations on data values. This operator, which subsumes the Until operators of PCTL (Probabilistic Computation Tree Logic) [41] and their action-based counterparts, can provide useful quantitative information about paths having certain (e.g., peak) cost values. We integrated the regular probabilistic operator into MCL and we devised an associated on-the-fly model checking method, based on a combined local resolution of linear and Boolean equation systems. We implemented the method in a prototype extension of the EVALUATOR model checker and experimented it on realistic PTSs modeling concurrent systems. This work led to a publication [22].

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

6.4.1. Compositional Verification

Participant: 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 2016, the *n among m* parallel composition operator "`par`" of the EXP language has been extended. Before the extension, the set of m processes among which any subset of size n could be synchronized on a given action was necessarily the set of all parallel processes composed by the "`par`" operator. From now on, by a slight extension of the syntax, this set of m processes can be defined as a subset of the parallel processes. Also, while n had to be strictly greater than 1, it can now also have value 0 (meaning that none of the m processes can perform the corresponding action) or 1 (meaning that each process can perform the corresponding action on its own, without synchronization). A bug in EXP.OPEN has been fixed and better messages are now emitted to warn the user about dubious usage of the "`par`" operator.

The SVL language has been extended to include the extended "`par`" operator. Two bugs have also been corrected.

6.4.2. Other Component Developments

Participants: Hubert Garavel, Frédéric Lang, Radu Mateescu, Wendelin Serwe.

Sustained effort was made to improve the documentation of the CADP toolbox. Various corrections have been brought to the CADP manual pages. A 27-page manual page explaining how the LOTOS language is implemented has been written, and the manual pages of the CAESAR and CAESAR.ADT compilers have been also updated. To make documentation more readable, the EVALUATOR3, and EVALUATOR4 manual pages have been splitted each in two parts, so as to better distinguish between the languages (namely, MCL3 and MCL) and their model checkers. The CADP distribution has been made leaner by keeping only the essential papers, and the "publications" and "tutorial" pages of the CADP Web site have been enriched and reorganized.

The conventions for string notations to represent "raw" values (i.e., values whose type is not a predefined one, but a custom type defined by the user) have been improved, together with the associated conversion algorithms for reading/writing raw values from/to strings. The BCG_WRITE manual page has been updated to more accurately describe how label fields of type "raw" are parsed. The behaviour of the functions `bcg_character_scan()`, `bcg_string_scan()`, `bcg_real_scan()`, and `bcg_raw_scan()` has been carefully revised, and all the BCG libraries and tools (especially BCG_IO) have been modified to follow the new conventions and emit better diagnostics when label fields contain incorrect notations of raw values. Also, BCG_IO has been enhanced so that very long execution sequences can be converted into SEQ or LOTOS files without causing stack overflow.

Finally, enhancements and bug fixes have been brought to other CADP components, including CADP_MEMORY, EUCALYPTUS, INSTALLATOR, OCIS, RFL, TST, and XTL. The style files for the various editors supported by CADP have been updated to take into account the recent features added to LNT. The predefined MCL libraries of the EVALUATOR model checker have been modified to generate more explanatory diagnostics for the inevitability operators.

Although CADP is mostly used on Linux, specific effort has been made to target other execution platforms. Concerning macOS: CADP now supports the recent versions 10.10 ("Yosemite"), 10.11 ("El Capitan"), and 10.12 ("Sierra"). Concerning Windows: changes have been brought to support Windows 10 and the 64-bit version of Cygwin (previously, only the 32-bit version was supported). Other adaptations were required to handle the recent versions of Cygwin packages, MinGW C compiler, and Mintty shell, as well as the case where Cygwin is not installed in "`C:\`", but in either "`C:\Cygwin`" or "`C:\Cygwin64`".

6.5. Real-Life Applications and Case Studies

6.5.1. Reconfiguration and Resilience of Distributed Cloud Applications

Participants: Umar Ozeer, Gwen Salaün.

In the context of a collaboration with Orange Labs, an Ensimag student (Bakr Derrazi) supervised by Xavier Etchevers and Gwen Salaün, has made his internship from February 2016 to July 2016 at Orange Labs. As a result, we have proposed a first solution and prototype for detecting and repairing failures of data-centric applications distributed in the cloud. A PhD thesis (Umar Ozeer) has started on this subject in November 2016.

6.5.2. Activity Detection in a Smart Home

Participants: Frédéric Lang, Radu Mateescu.

In collaboration with Paula-Andrea Lago-Martinez and Claudia Roncancio (SIGMA team, LIG) and with Nicolas Bonnefond (PERVASIVE INTERACTION team, Inria and LIG), we study how formal methods can help to analyze logs of events obtained from the many sensors and actuators installed in the Amiqual4Home smart home.

In 2016, we considered using the MCL temporal logic to detect the start and end of activities in a log, such as cooking or taking a shower. We applied our tools on a log containing about $140,000$ events that had been generated over 10 days of living in the smart home. This preliminary study has shown that the MCL temporal logic is sufficiently rich to enable an easy specification of the searched activities, notably thanks to its multiple extensions such as macro definitions, parameterized fixed point operators, and data handling mechanisms. The particularly long length of the analyzed logs also enabled us to improve some of the CADP tools, so that they better scale up. This work led to an article submitted to an international conference.

6.5.3. Other Case Studies

Participant: Hubert Garavel.

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 2016, the effort to maintain and enhance these demos has been pursued. The demo 12 (Message Authentication Algorithm) and demo 31 (SCSI-2 bus arbitration protocol) have been manually translated from LOTOS to LNT. Additionally, demo 12 has been deeply revised by simplifying its LOTOS, LNT, and C code, by taking advantage of the imperative-programming features of LNT, and by enriching the LNT specification with the test cases contained in the original MAA description. This allowed to detect and correct a mistake in the C code implementing function `HIGH_MUL()`. Other CADP demos (namely demos 05, 16, and 36) have also been simplified and/or enhanced in various ways.

HYCOMES Project-Team

7. New Results

7.1. Structural Analysis of Multi-Mode DAEs

Differential Algebraic Equation (DAE) systems constitute the mathematical model supporting physical modeling languages such as Modelica or Simscape. Unlike Ordinary Differential Equations, or ODEs, they exhibit subtle issues because of their implicit *latent equations* and related *differentiation index*. Multi-mode DAE (mDAE) systems are much harder to deal with, not only because of their mode-dependent dynamics, but essentially because of the events and resets occurring at mode transitions. Unfortunately, the large literature devoted to the numerical analysis of DAEs do not cover the multi-mode case. It typically says nothing about mode changes. This lack of foundations cause numerous difficulties to the existing modeling tools. Some models are well handled, others are not, with no clear boundary between the two classes. In [11], we develop a comprehensive mathematical approach to the *structural analysis* of mDAE systems which properly extends the usual analysis of DAE systems. We define a constructive semantics based on nonstandard analysis and show how to produce execution schemes in a systematic way. This work has been accepted for presentation at the HSCC 2017 conference [19] in April 2017.

7.2. Decoupling Abstractions

In [10], we investigated decoupling abstractions, by which we seek to simulate (i.e. abstract) a given system of ordinary differential equations (ODEs) by another system that features completely independent (i.e. uncoupled) sub-systems, which can be considered as separate systems in their own right. Beyond a purely mathematical interest as a tool for the qualitative analysis of ODEs, decoupling can be applied to verification problems arising in the fields of control and hybrid systems. Existing verification technology often scales poorly with dimension. Thus, reducing a verification problem to a number of independent verification problems for systems of smaller dimension may enable one to prove properties that are otherwise seen as too difficult. We show an interesting correspondence between Darboux polynomials and decoupling simulating abstractions of systems of polynomial ODEs and give a constructive procedure for automatically computing the latter.

7.3. Formal Verification of the ACAS X System

The *Next-Generation Airborne Collision Avoidance System* (ACAS X) is intended to be installed on all large aircraft to give advice to pilots and prevent mid-air collisions with other aircraft. It is currently being developed by the Federal Aviation Administration (FAA). In [6], we determine the geometric configurations under which the advice given by ACAS X is safe under a precise set of assumptions and formally verify these configurations using hybrid systems theorem proving techniques. We consider subsequent advisories and show how to adapt our formal verification to take them into account. We examine the current version of the real ACAS X system and discuss some cases where our safety theorem conflicts with the actual advisory given by that version, demonstrating how formal hybrid systems proving approaches are helping to ensure the safety of ACAS X. Our approach is general and could also be used to identify unsafe advice issued by other collision avoidance systems or confirm their safety.

7.4. Chattering-Free Simulation

Chattering is a fundamental phenomenon that is unique to hybrid systems, due to the complex interaction between discrete dynamics (in the form of discrete transitions) and continuous dynamics (in the form of time). In practice, simulating chattering hybrid systems is challenging in that simulation effectively halts near the chattering time point, as an infinite number of discrete transitions would need to be simulated. In [7],

formal conditions are provided for when the simulated models of hybrid systems display chattering behavior, and methods are proposed for avoiding chattering "on the fly" in runtime. We utilize dynamical behavior analysis to derive conditions for detecting chattering without enumeration of modes. We also present a new iterative algorithm to allow for solutions to be carried past the chattering point, and we show by a prototypical implementation how to generate the equivalent chattering-free dynamics internally by the simulator in the main simulation loop.

MUTANT Project-Team

6. New Results

6.1. Embedding Audio Processing

Participants: Jean-Louis Giavitto, Pierre Donat-Bouillud.

Audio processing has been integrated in the Antescofo language. This experimental extension aims at providing sample-accurate control and dynamic audio graphs directly in Antescofo. Currently, FAUST (through a native embedding of the in-core compiler) and a few specific signal processors (notably FFT) can be defined. The tight integration enable specification of multiple-timed signal processing in conjunction with control programs. One example of this integration is the use of symbolic curve specification to specify variations of control parameters at sample rate, a task whose correctness in real-time is not at the scope of competing systems. Our approach has proven to provide such mechanisms at a lower computational cost; for example a factor of two in the *remaking* of Boulez' piece *Antheme 2* compared to the original version with the audio effects managed in Max. We will further pursue such optimizations while extending sample accuracy, by developing a type-system to preserve block computations in case of preemptive audio processing [41].

The reduced footprint enable the embedding of an *Antescofo* engine with internal audio processing on Raspberry PI and UDOO nano-computers (early results are reported in [26]).

6.2. Representation of Rhythm and Quantization

Participants: Florent Jacquemard, Adrien Ycart, Pierre Donat-Bouillud.

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 structures, which were originally used in natural language processing. We are particularly interested in two well established formalisms with solid theoretical foundations: weighted automata for trees and dags and term rewriting.

Our main contribution in that context is the development of a new framework for rhythm transcription, the problem of the generation, from a sequence of timestamped notes, *e.g.* a file in MIDI format, of a score in traditional music notation) – see Section 5.2 . 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. In collaboration with Jean Bresson (Ircam) and Slawek Staworko (LINKS), we are developing an approach based on algorithms for the enumeration of large sets of weighted trees (tree series), representing possible solutions to a problem of transcription. The implementation work is performed by Adrien Ycart, under a research engineer contract with Ircam. This work has been presented in [22], [23].

Moreover, in collaboration with Prof. Masahiko Sakai (Nagoya University), we are working on symbolic processing of music notation, based on the above models. We proposed a structural theory (equational system on rhythm trees) defining equivalence on rhythm notations [42], [51], and use this approach, for instance, to generate, by transformation, different possible notations of the same rhythm, with the ability to select either alternative notation in accordance with certain constraints, *e.g.* in the context of transcription.

Related results on the property of confluence of term rewriting systems were presented in [19] (invited talk), and other work on data tree processing, in collaboration with Luc Segoufin and Jeremie Dimino, have published in [16].

6.3. Model-based Testing an Interactive Music System

Participants: Clément Poncelet, Florent Jacquemard, Pierre Donat-Bouillud.

We have been pursuing in 2016 our applications of model-based timed testing techniques to the interactive music system Antescofo, in the context of the Phd of Clément Poncelet and in relation with the developments presented in Section 5.3 .

Several formal methods have been developed for automatic conformance testing of critical embedded software, with the execution of a real implementation under test (IUT, or black-box) in a testing framework, where carefully selected inputs are sent to the IUT and then the outputs are observed and analyzed. 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 case of IMS presents important originalities compared to other applications of MBT to realtime systems. On the one hand, the time model of IMS comprises 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 raises several new problems for the generation of test suites and their execution. On the other hand, we can reasonably assume that a given mixed score of Antescofo specifies completely the expected timed behavior of the IMS, and compile automatically the given score into a formal model of the IUT's expected behavior, using an intermediate representation. This give a fully automatic test method, which is in contrast with other approaches which generally require experts to write the specification manually.

We have developed online and offline approaches to MBT for Antescofo. The offline approach relies on tools of the Uppaal suite [53], [50], using a translation of our models into timed automata. The online approach is based on a virtual machine executing the models of score in Intermediate Representation (IR).

To this respect, the transformation of Antescofo's mixed scores (in DSL) into IR, described in Section 5.3 , can be seen as the premise of a compiled approach for Antescofo.

These results have been published this year in the Journal of New Music Research [14], the journal Science of Computer Programming [18], and in the PhD of Clément Poncelet, defended in November 2016.

PARKAS Project-Team

6. New Results

6.1. Verified compilation of Lustre

Participants: Timothy Bourke, Lélio Brun, Marc Pouzet.

Synchronous dataflow languages and their compilers are increasingly used to develop safety-critical applications, like fly-by-wire controllers in aircraft and monitoring software for power plants. A striking example is the SCADE Suite tool of ANSYS/Esterel Technologies which is DO-178B/C qualified for the aerospace and defense industries. This tool allows engineers to develop and validate systems at the level of abstract block diagrams that are automatically compiled into executable code.

Formal modelling and verification in an interactive theorem prover can potentially complement the industrial certification of such tools to give very precise definitions of language features and increased confidence in their correct compilation; ideally, right down to the binary code that actually executes.

This year we integrated elements of the CompCert verified C compiler into our Lustre compiler. In particular, we modularized the syntax and semantics of our source Lustre language and intermediate Obc language to be independent of the underlying types and operators of the host language. All previous proofs are independent of the choice of host language. We integrated CompCert by instantiating the types and operators with those of the Clight language and by adding a function that compiles an Obc program into Clight. The key challenge in this compilation pass is to move from a model where program variables are stored in a tree structure where distinctness is manifest to a model where variables are stored in nested structures in a single memory block with concomitant problems of aliasing, alignment, and memory size. We addressed this challenge by extending a CompCert library for expressing separation assertions and applying it to express our recursive predicates.

A similar approach was taken to address the encoding of multiple return values (permitted in Obc but not in Clight). We made various practical improvements to our compiler and proofs including the addition of a verified parser, the addition of an elaboration pass with type and clock checking, and pretty-printers for intermediate languages. It is now possible to compile scheduled and normalized Lustre programs to assembly code with a proof correction that relates the generated transition function to the dataflow semantics of the source program.

The initial part of this work, reported last year, has been published [20].

In collaboration with Pierre-Évariste Dagand (CNRS), Lionel Reig (Collège de France), and Xavier Leroy (Inria, GALLIUM team).

6.2. Fence Optimisations for Multicore Architectures

Participants: Robin Morisset, Francesco Zappa Nardelli.

We have pursued our investigation of sound optimisations for modern multicore architectures. Last year we focused on optimisations that can be expressed inside the semantics of the C11/C++11 programming language; we thus moved to optimisations that can be expressed only at the harware level. In particular we have shown how partial redundancy elimination (PRE) can be instantiated to perform *provably correct* fence elimination for multi-threaded programs running on top of the x86, ARM and IBM Power relaxed memory models. We have implemented our algorithm in the x86, ARM and Power backends of the `LLVM` compiler infrastructure. The optimisation does not induce an observable overhead at compile-time and can result in up-to 10% speedup on some benchmarks.

This work has been published in CC 2017 [18]. The implementation of the optimisations will be submitted for inclusion in the LLVM compiler suite.

6.3. Compiling synchronous languages for multi-processor implementations

Participants: Timothy Bourke, Albert Cohen, Guillaume Iooss, Marc Pouzet.

Working together with industrial partners in the context of the ASSUME project, we have been working to treat a large-scale and complete case study of an industrial application. This has involved studying the original sources and adapting the Heptagon Lustre compiler. Three main extensions have been developed this year: a mechanism to calculate and exploit module interdependencies; an extension to the type system to allow operator overloading via ad hoc polymorphism; and modifications to the parser to accept the provided source code. We have also worked on a means to generate dependency graphs from the provided nonfunctional specifications.

Our current work centers on understanding how to formalize the peculiarities of this class of application and the target architecture in our framework, and on generating Lustre code from the non-functional specifications. The ultimate aim is to generate correct multi-processor task-parallel real-time code for an embedded target and to integrate with both the Heptagon and Vélus compilers.

In collaboration (this year) with Dumitru Potop-Butucaru (Inria, AOSTE team), Keryan Didier (Inria, AOSTE team), Jean Souyris (Airbus), and Adrien Gauffriau (Airbus).

POSET Team

7. New Results

7.1. Alpha release of the T-calculus

One of the main achievement of the PoSET project in 2016 is the alpha release of the T-calculus [15] that not only implements the tiled front-end programing interface that was proposed earlier [10], [8], but also an original mide-end programing interface for implementing interactive behavior and the related categorical combinators that allows for effectively running these high level constructs.

7.2. A new collaboration with Bernard Serpette

A new collaboration with Bernard Serpette also aim at developing a formal models for the T-calculus semantics [27], [25]. Though at its birth, such an approach eventually reveals rather deep connection with Matsikoudis and Lee works on causal functions semantics [33], openning new perspectives towards higher-order timed programing.

SPADES Project-Team

6. New Results

6.1. Components and contracts

Participants: Alain Girault, Christophe Prévot, Sophie Quinton, Jean-Bernard Stefani.

6.1.1. Contracts for the negotiation of embedded software updates

We address the issue of change after deployment in safety-critical embedded system applications in collaboration with Thales and also in the context of the CCC project (http://ccc-project.org/).

The goal of CCC is to substitute lab-based verification with in-field formal analysis to determine whether an update may be safely applied. This is challenging because it requires an automated process able to handle multiple viewpoints such as functional correctness, timing, etc. For this purpose, we propose an original methodology for contract-based negotiation of software updates. The use of contracts allows us to cleanly split the verification effort between the lab and the field. In addition, we show how to rely on existing viewpoint-specific methods for update negotiation. We have validated our approach on a concrete example inspired by the automotive domain in collaboration with our German partners from TU Braunschweig [19].

In collaboration with Thales we mostly focus on timing aspects with the objective to anticipate at design time future software evolutions and identify potential schedulability bottlenecks. This year we have presented an approach to quantify the flexibility of a system with respect to timing. In particular we have shown that it is possible under certain conditions to identify the task that will directly induce the limitations on a possible software update. If performed at design time, such a result can be used to adjust the system design by giving more slack to the limiting task [21].

6.1.2. Location graphs

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 recently introduced the location graph model [79], where components are understood as graphs of locations hosting higher-order processes, and where component structures can be arbitrary graphs.

We have continued the development of location graphs, revisiting the underlying structural model (hypergraphs instead of graphs), and simplifying its operational semantics while preserving the model expressivity. Towards the development of a behavioral theory of location graphs, we have defined different notions of bisimilarity for location graphs and shown them to be congruences, although a fully fledged co-inductive characterization of contextual equivalence for location graphs is still in the works. This work has not yet been published.

6.2. Real-Time multicore programming

Participants: Pascal Fradet, Alain Girault, Gregor Goessler, Xavier Nicollin, Sophie Quinton.

6.2.1. Time predictable programming languages

Time predictability (PRET) is a topic that emerged in 2007 as a solution to the ever increasing unpredictability of today's embedded processors, which results from features such as multi-level caches or deep pipelines [52]. For many real-time systems, it is mandatory to compute a strict bound on the program's execution time. Yet, in general, computing a tight bound is extremely difficult [82]. The rationale of PRET is to simplify both the programming language and the execution platform to allow more precise execution times to be easily computed [34].

Following our past results on the PRET-C programming language [32], we have proposed a time predictable synchronous programming language for multicores, called FOREC. It extends C with a small set of ESTEREL-like synchronous primitives to express concurrency, interaction with the environment, looping, and a synchronization barrier [83] (like the `pause` statement in ESTEREL). 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. We provide several deterministic combine policies for shared variables, in a way similar as concurrent revisions [45]. Thanks to this, it benefits from a deterministic semantics. FOREC is compiled into threads that are then statically scheduled for a target multicore chip. Our WCET analysis takes into account the access to the shared TDMA bus and the necessary administration for the shared variables. We achieve a very precise WCET (the over-approximation being less than 2%) thanks to a reachable space exploration of the threads' states [15]. We have published a research report presenting the complete semantics and the compiler [27], and submitted it to a journal.

Furthermore, we have extended the PRET-C compiler [32] in order to make it energy aware. To achieve this, we use dynamic voltage and frequency scaling (DFVS) and we insert DVFS control points in the control flow graph of the PRET-C program. The difficulty is twofold: first the control flow graph is concurrent, and second resulting optimization problem is in the 2D space (time,energy). Thanks to a novel ILP formulation and to a bicriteria heuristic, we are able to address the two objectives jointly and to compute, for each PRET-C program, the Pareto front of the non-dominated solutions in the 2D space (time, energy) [20].

This is a collaboration with Eugene Yip from Bamberg University, and with Partha Roop and Jiajie Wang from the University of Auckland.

6.2.2. Modular distribution of synchronous programs

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

We are working on defining a *fully-conservative* extension of a synchronous data-flow programming language (the HEPTAGON language, inspired from LUCID SYNCHRONE [46]). The extension, by means of *annotations* adds *abstract location parameters* to functions, and *communications* of values between locations. At deployment, every abstract location is assigned an actual one; this yields an executable for each actual computing resource. Compared to the PhD of Gwenaël Delaval [50], [51], the goal here is to achieve *modular* distribution even in the presence of non-static clocks, *i.e.*, clocks defined according to the value of inputs.

By *fully-conservative*, we have three aims in mind:

1. A non-annotated (*i.e.*, centralized) program will be compiled exactly as before;
2. An annotated program eventually deployed onto only one computing location will behave exactly as its centralized couterpart;
3. The input-output semantics of a distributed program is the same as its centralized counterpart.

By *modular*, we mean that we want to compile each function of the program into a single function capable of running on any computing location. At deployment, the program of each location may be optimized (by simple Boolean-constant-propagation, dead-code and unused-variable elimination), yielding different optimized code for each computing location.

We have formalized the type-system for inferring the location of each variable and computation. In the presence of local clocks, added information is computed from the existing clock-calculus and the location-calculus, to infer necessary communication of clocks between location. All pending theorical and technical issues have been answered, and the new compiler is being implemented, with new algorithms for deployment (and code optimization), achieving the three aims detailed above.

6.2.3. 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 [55], which extends the standard dataflow model by allowing rates to be parametric, and the *Boolean Parametric Data Flow* (BPDF) MoC [38], [37] which combines integer parameters (to express dynamic rates) and boolean parameters (to express the activation and deactivation of communication channels). In the past years, several other parametric dataflow MoCs have been presented. All these models aim at providing an interesting trade-off between analyzability and expressiveness. They offer a controlled form of dynamism under the form of parameters (*e.g.*, parametric rates), along with run-time parameter configuration.

We have written a survey which provides a comprehensive description of the existing parametric dataflow MoCs (constructs, constraints, properties, static analyses) and compares them using a common example [11]. The main objectives are to help designers of streaming applications to choose the most suitable model for their needs and to pave the way for the design of new parametric MoCs.

We have also studied *symbolic* analyses of data-flow graphs [24], [16], [17], [12]. Symbolic analyses express the system performance as a function of parameters (*i.e.*, input and output rates, execution times). Such functions can be quickly evaluated for each different configuration or checked *w.r.t.* different quality-of-service requirements. These analyses are useful for parametric MoCs, partially specified graphs, and even for completely static SDF graphs. We provide symbolic analyses for computing the maximal throughput of acyclic synchronous dataflow graphs, the minimum required buffers for which as soon as possible (asap) scheduling achieves this throughput, and finally the corresponding input-output latency of the graph. We first investigate these problems for a single parametric edge. The results are then extended to general acyclic graphs using linear approximation techniques. We assess the proposed analyses experimentally on both synthetic and real benchmarks.

6.2.4. Synthesis of switching controllers using approximately bisimilar multiscale abstractions

The use of discrete abstractions for continuous dynamics has become standard in hybrid systems design (see *e.g.*, [80] and the references therein). The main advantage of this approach is that it offers the possibility to leverage controller synthesis techniques developed in the areas of supervisory control of discrete-event systems [75]. The first attempts to compute discrete abstractions for hybrid systems were based on traditional systems behavioral relationships such as simulation or bisimulation, initially proposed for discrete systems most notably in the area of formal methods. These notions require inclusion or equivalence of observed behaviors which is often too restrictive when dealing with systems observed over metric spaces. For such systems, a more natural abstraction requirement is to ask for closeness of observed behaviors. This leads to the notions of approximate simulation and bisimulation introduced in [56].

These approaches are based on sampling of time and space where the sampling parameters must satisfy some relation in order to obtain abstractions of a prescribed precision. In particular, the smaller the time sampling parameter, the finer the lattice used for approximating the state-space; this may result in abstractions with a very large number of states when the sampling period is small. However, there are a number of applications where sampling has to be fast; though this is generally necessary only on a small part of the state-space. We have been exploring two approaches to overcome this state-space explosion [5].

We are currently investigating an approach using mode sequences of given length as symbolic states for our abstractions. By using mode sequences of variable length we are able to adapt the granularity of our abstraction to the dynamics of the system, so as to automatically trade off precision against controllability of the abstract states.

6.2.5. Schedulability of weakly-hard real-time systems

We focus on the problem of computing tight deadline miss models for real-time systems, which bound the number of potential deadline misses in a given sequence of activations of a task. In practical applications, such guarantees are often sufficient because many systems are in fact not hard real-time [4].

Our major contribution this year is the extension of our method for computing deadline miss models, called Typical Worst-Case Analysis (TWCA), to systems with task dependencies. This allows us to provide bounds on deadline misses for systems which until now could not be analyzed [18].

In parallel, we have developed an extension of sensitivity analysis for budgeting in the design of weakly-hard real-time systems. During design, it often happens that some parts of a task set are fully specified while other parameters, e.g. regarding recovery or monitoring tasks, will be available only much later. In such cases, sensitivity analysis can help anticipate how these missing parameters can influence the behavior of the whole system so that a resource budget can be allocated to them. We have developed an extension of sensitivity analysis for deriving task budgets for systems with hard and weakly-hard requirements. This approach has been validated on synthetic test cases and a realistic case study given by our partner Thales. This work will be submitted soon.

Finally, in collaboration with TU Braunschweig and Daimler we have investigated the use of TWCA in conjunction with the Logical Execution Time paradigm [68] according to which data are read and written at predefined time instants. In particular, we have extended TWCA to different deadline miss handling strategies. This work has not been published yet.

6.3. Language Based Fault-Tolerance

Participants: Pascal Fradet, Alain Girault, Yoann Geoffroy, Gregor Goessler, Jean-Bernard Stefani, Martin Vassor, Athena Abdi.

6.3.1. Fault Ascription in Concurrent 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". In computer science, almost all definitions of logical causality — including the landmark definition of [63] and its derivatives — rely 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 [7].

In his PhD thesis, Yoann Geoffroy proposed a generalization of our fault ascription technique to systems composed of black-box and white-box components. For the latter a faithful behavioral model is given but no specification. The approach leverages results from game theory and discrete controller synthesis to define several notions of causality.

We are currently working on an instantiation of our general semantic framework for fault ascription in [60] to acyclic models of computation, in order to compare our approach with the standard definition of *actual causality* proposed by Halpern and Pearl.

6.3.2. Tradeoff exploration between energy consumption and execution time

We have continued our work on multi-criteria scheduling, in two directions. First, in the context of dynamic applications that are launched and terminated on an embedded homogeneous 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 subsets 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 dynamic 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 be minimized. For this, we *(i)* allocate a sufficient number of cores to each active application, *(ii)* allocate the unassigned cores to the applications yielding the largest gain in energy, and *(iii)* 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 visited the team in September 2015.

Second, in the context of a static application (again represented a DAG of tasks) running on an homogeneous multi-core chip, we have worked on the static scheduling minimizing the WCRT of the application under the multiple constraints that the reliability, the power consumption, and the temperature remain below some given thresholds. There are multiple difficulties: *(i)* the reliability is not an invariant measure w.r.t. time, which makes it impossible to use backtrack-free scheduling algorithms such as list scheduling [33]; to overcome this, we adopt instead the Global System Failure Rate (GSFR) as a measure of the system's reliability, which is invariant with time [57]; *(ii)* keeping the power consumption under a given threshold requires to lower the voltage and frequency, but this has a negative impact both on the WCRT and on the GSFR; keeping the GSFR below a given threshold requires to replicate the tasks on multiple cores, but this has a negative impact both on the WCRT, on the power consumption, and on the temperature; *(iii)* keeping the temperature below a given threshold is even more difficult because the temperature continues to increase even after the activity stops, so each scheduling decision must be assessed not based on the current state of the chip (*i.e.*, the temperature of each core) but on the state of the chip at the end of the candidate task, and cooling slacks must be inserted. We have proposed a multi-criteria scheduling heuristics to address these challenges. It produces a static schedule of the given application graph and the given architecture description, such that the GSFR, power, and temperature thresholds are satisfied, and such that the execution time is minimized. We then combine our heuristic with a variant of the ε-constraint method [62] in order to produce, for a given application graph and a given architecture description, its entire Pareto front in the 4D space (exec. time, GSFR, power, temp.). This is a joint work with Athena Abdi and Hamid Zarandi from Amirkabir U., Iran, who have visited the team in 2016.

6.3.3. 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 cycles"*.

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 [25]. 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. As our experiments show, if the SEU fault-model is replaced with the stricter fault-model of SET, it has a minor impact on the number of removed voters. On the other hand, BDD-based modeling of SET effects represents a more complex task than the modeling of an SEU as a bit-flip. We propose solutions for this task and explain the nature of encountered problems. We discuss scalability issues arising from formal verification with approximations and assess their efficiency and precision.

6.3.4. Concurrent flexible reversibility

Reversible concurrent models of computation provide natively what appears to be very fine-grained checkpoint and recovery capabilities. We have made this intuition clear by formally comparing a distributed algorithm for checkpointing and recovery based on causal information, and the distributed backtracking algorithm that lies at the heart of our reversible higher-order pi-calculus. We have shown that (a variant of) the reversible higher-order calculus with explicit rollback can faithfully encode a distributed causal checkpoint and recovery algorithm. The reverse is also true but under precise conditions, which restrict the ability to rollback a computation to an identified checkpoint. This work has currently not been published.

TEA Project-Team

7. New Results

7.1. Toward a distribution of ADFG

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

The ADFG tool is being developed in the context of the ADT "Opama" in order to serve both as scheduler synthesis tool from AADL specifications and ordinary tasksets. ADFG has been partly rewritten in order to target more users : it is now freely available online and comes with a complete documentation. These improvements imply that ADFG does not anymore provide Safety Critical Java application generation; its main purpose of scheduler synthesis is now available from an Eclipse plugin, as a command-line interface, and also in Polychrony (as part of the AADL to Signal translation process). Moreover ADFG accepts and exports several file formats with related scheduling tools: SDF3, Yartiss and soon Cheddar.

The Eclipse interface has changed significantly with a dialog window and a console to present the results (as shown in the figure 4). Also the graphical data-flow graph editor is still present but has been simplified. An other big change (not seen by the end-user) is the internal use of the free LpSolve linear programming software instead of CPLEX. Finally, it will soon be possible to use this software not only as a scheduling synthesizer but also as a scheduling checker (with timing properties given by the user).

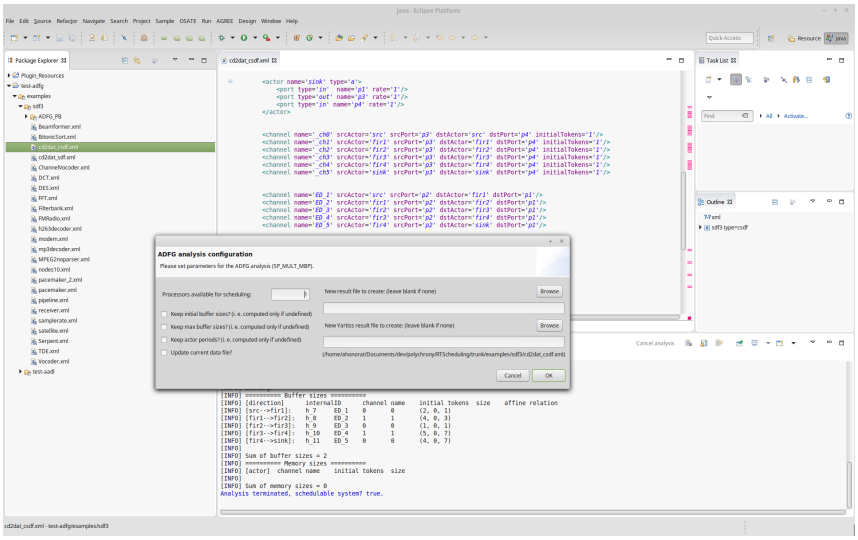

Figure 4. ADFG under Eclipse

7.2. Modular verification of cyber-physical systems using contract theory

Participants: Jean-Pierre Talpin, Benoit Boyer, David Mentre, Simon Lunel.

The primary goal of our project, in collaboration with Mitsubishi Electronics Research Centre Europe (MERCE), is to ensure correctness-by-design in realistic cyber-physical systems, i.e., systems that mix software and hardware in a physical environment, e.g., Mitsubishi factory automation lines or water-plant factory. To achieve that, we develop a verification methodology based on contract reasoning.

We have first performed a state of the art of the research and the work of A. Platzer with the Differential Dynamic Logic ($d\mathcal{L}$) retained our attention [0]. This a formalism built on the Dynamic Logic of V. Pratt augmented with the possibility of expressing Ordinary Differential Equations (ODEs). ODEs are the usual way to model physical behaviors in physics and $d\mathcal{L}$ permits to accurately model cyber-physical systems. But this logic can also express properties on real arithmetic and there is proof system associated, under the form of a sequent calculus, which let us a mean to prove specifications. To finish, it is very natural to use contract to specify systems since it was the primary goal of the work of V. Pratt. To conclude, $d\mathcal{L}$ is particularly fit to our purpose.

We have some preliminary results about a design-by-composition methodology: we have defined a syntactic composition operator in $d\mathcal{L}$, which enjoys associativity and commutativity. We have then characterized the conditions under which we can derive automatically a proof of the contract of our composition. To exemplified our ideas, we are currently studying a simplified water-tank system, which will serve as a basis for more realistic case studies. We plan to provide refinement and abstraction mechanisms to ultimately allow a mix between vertical and horizontal design.

7.3. Runtime verification and trace analysis

Participants: Vania Joloboff, Daian Yue, Frédéric Mallet.

When engineers design a new cyber physical system, there are well known requirements that can be translated as system properties that must be verified. These properties can be expressed in some formalism and when the model has been designed, the properties can be checked at the model level, using model checking techniques or other model verification techniques.

This requires that the properties are well specified at the time the virtual prototype is assembled. However it is also the case that many intrinsic properties are actually unforeseen when the virtual prototype is assembled, for example that some hardware buffer overflow should not remain unnoticed by the software. In most cases, during system design the simulation fails: the engineers then must investigate the cause of the failure.

A widely used technique for that consists in storing all of the trace data of simulation sessions into trace files, which are analyzed later with specialized trace analyzer tools. Such trace files have become huge, possibly hundred of Gigabytes as all data are stored into the trace files, and have become intractable by human manual handling.

In order to better identify the reason for such failures and capture the missing properties that the system should verify we have started to work on a new run time verification approach based on trace analysis. Approaches like PSL requires that the properties to verify are known before hand. Our approach is attempting for the engineers to experiment various property verification of failing simulations without re-building the virtual prototype. We have established a technique that makes it possible to investigate properties either statically working from a trace file or dynamically by introducing a dynamic verification component into the virtual prototype, or actually the real system.

The Trace Runtime Analysis Platform (TRAP) provides a model-based framework and implements the corresponding tool chain to support runtime analysis and verification of traces generated by virtual prototypes or cyber-physical systems. The main goal is to make it easy for engineers to define system properties that should be satisfied and verify them at system runtime (or from a recorded session). The property verification tools proposed do not require a detailed knowledge of the system implementation, do not require any modification or recompilation of the system to investigate different properties, and do not require the engineers to be familiar with temporal logic. TRAP proposes Domain Specific Languages (DSL's) integrated within the Eclipse Modeling Framework to express the properties. The DSL tool-chain uses the concept of Logical Clock defined by CCSL and takes advantage of CCSL clock algebra as the underlying formal support. The DSL's compilers eventually generate C++ code to verify the properties at run time, making usage of dynamically loaded code.

[0] *Differential Dynamic Logic for Hybrid Systems*, André Platzer, http://symbolaris.com/logic/dL.html

This year we have investigated and implemented this approach, using Eclipse EMF. The STML and TPSL compilers are implemented in Java and generate C++ code. Results of this work have been published at the FDL'16 conference referenced on IEEE Explore. [17]

7.4. Polychronous automata and formal validation of AADL models

Participants: Loïc Besnard, Thierry Gautier, Alexandre Honorat, Clément Guy, Jean-Pierre Talpin.

We have defined a model of *polychronous automata* based on clock relations [7]. A specificity of this model is that an automaton is submitted to clock constraints: these finite-state automata define transition systems to express explicit reactions together with properties, in the form of Boolean formulas over logical time, to constrain their behavior. This allows one to specify a wide range of control-related configurations, either reactive, or restrictive with respect to their control environment. A semantic model is defined for these polychronous automata, that relies on a Boolean algebra of clocks. Polychronous automata integrate smoothly with data-flow equations in the polychronous model of computation.

This polychronous MoC has been used previously as semantic model for systems described in the core AADL standard. The core AADL is extended with annexes, such as the Behavior Annex, which allows to specify more precisely architectural behaviors. The translation from AADL specifications into the polychronous model should take into account these behavior specifications, which are based on description of automata.

For that purpose, the AADL state transition systems are translated as Signal automata (a slight extension of the Signal language has been defined to support the model of polychronous automata). States are declared as Signal labels. Transitions are expressed using a call to a specific Signal process `Automaton\_Transition` which takes as parameters the labels of the source and destination states, and the condition expression corresponding to the AADL guard of the transition. The transition processes implicitly declare the equations that are required to compute the firing instants of the transitions. These processes, viewed as macros, are replaced during Signal compilation with a set of Signal equations handling current state and transition firing.

Once the AADL model of a system transformed into a Signal program, one can analyze the program using the Polychrony framework in order to check if timing, scheduling and logical requirements over the whole system are met.

We have implemented the translation and experimented it using a concrete case study, which is the AADL modeling of an Adaptive Cruise Control (ACC) system, a highly safety-critical system embedded in recent cars.

7.5. Formal Semantics of Behavior Specifications in the Architecture Analysis and Design Language Standard

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

In system design, an architecture specification or model serves, among other purposes, as a repository to share knowledge about the system being designed. Such a repository enables automatic generation of analytical models for different aspects relevant to system design (timing, reliability, security, etc.). The Architecture Analysis and Design Language (AADL) is a standard proposed by SAE to express architecture specifications and share knowledge between the different stakeholders about the system being designed. To support unambiguous reasoning, formal verification, high-fidelity simulation of architecture specifications in a model-based AADL design work-flow, we have defined a formal semantics for the behavior specification of the AADL. Since it began being discussed in the AADL standard committee, our formal semantics evolved from a synchronous model of computation and communication to a semantic framework for time and concurrency in the standard: asynchronous, synchronous or timed, to serve as a reference for model checking, code generation or simulation tools uses with the standard [14]. These semantics are simple, relying on the structure of automata present in the standard already, yet provide tagged, trace semantics framework to establish formal relations between (synchronous, asynchronous, timed) usages or interpretations of behavior.

We define the model of computation and communication of a behavior specification by the synchronous, timed or asynchronous traces of automata with variables. These constrained automata are derived from *polychronous automata* defined within the polychronous model of computation and communication [7].

States of a behavior annex transition system can be either observable from the outside (`initial`, `final` or `complete` states), that is states in which the execution of the component is paused or stopped and its outputs are available; or non observable execution states, that is internal states. We thus define two kinds of steps in the transition system: *small steps*, that is non-observable steps from or to an internal state; and *big steps*, that is observable steps from a *complete* state to another, through a number of small steps). The semantics of the AADL considers the observable states of the automaton. The set of states S_A of automaton A (used to interpret the behavior annex) thus only contains states corresponding to these observable states and the set of transitions T_A big-step transitions from an observable state to another (by opposition with small-step transitions from or to an execution state). The action language of the behavior annex defines actions performed during transitions. Actions associated with transitions are action blocks that are built from basic actions and a minimal set of control structures (sequences, sets, conditionals and loops). Typically, a behavior action sequence is represented by concatenating the transition systems of its elements; a behavior action set is represented by composing the transition systems of its elements.

For our semantics, we considered a significant subset of the behavioral specification annex of the AADL. This annex allows one to attach a behavior specification to any components of a system modeled using the AADL, and can be then analyzed for different purposes which could be, for example, the verification of logical, timing or scheduling requirements.

7.6. Integration of Polychrony with QGen

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

The FUI project P gave birth to the GGen qualifiable model compiler, developed by Adacore. The tool accepts a discrete subset of Simulink expressed in a language called P and produces C or Ada code.

Our contribution was about providing a semantic bridge between Polychrony and QGen [15]. Our objective was to use Polychrony to compute fined-grained static scheduling of computations and communications for P models based on architectural properties. This work was twofold. First, we defined an alternative unambiguous static block scheduler for QGen, which can compute both partial and total orders based on user preferences. The purpose of this sequencer is to allow QGen to inter-operate with external sequencing tools while providing guarantees about the compatibility of external block execution orders with respect to both QGen's compilation scheme and user expectations. On the other hand, we developed a transformation function from the P language, more precisely, from the System Model subset of P, to the Signal meta-model, SSME. This work is based on a high-level API designed on top of SSME and can be used to transform a subset of Simulink to Signal. We validated our approach with the test suite used by QGen which is composed of over two-hundred small-sized Simulink models. We tested both block sequencing and model transformations. We ran the conversion tool and the set of models used by QGen for its regression tests and successfully converted medium to large models. The P language is capable of representing a useful subset of Simulink. That is why it is an interesting tool to help interpreting Simulink models and possibly architectural properties as executable Signal programs. The programs currently produced with our transformation tool can be compiled by Polychrony and reorganized as clusters of smaller processes.

7.7. Code generation for poly-endochronous processes

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

The synchronous modeling paradigm provides strong correctness guarantees for embedded system design while requiring minimal environmental assumptions. In most related frameworks, global execution correctness is achieved by ensuring the insensitivity of (logical) time in the program from (real) time in the environment. Tis property, called endochrony, can be statically checked, making it fast to ensure design correctness. Unfortunately, it is not preserved by composition, which makes it difficult to exploit with component-based design concepts in mind. It has been shown that compositionality can be achieved by weakening the objective of endochrony: a weakly endochronous system is a deterministic system that can perform independent computations and communications in any order as long as this does not alter its global state. Moreover, the non-blocking composition of weakly endochronous processes is isochronous, which means that the synchronous and asynchronous compositions of weakly endochronous processes accept the same behaviors. Unfortunately, testing weak endochrony needs state-space exploration, which is very costly in the general case. Then, a particular case of weak endochrony, called polyendochrony, was defined, which allows static checking thanks to the existing clock calculus. The clock hierarchy of a polyendochronous system may have several trees, with synchronization relations between clocks placed in different trees, but the clock expressions of the clock system must be such that there is no clock expression (especially, no root clock expression) defined by symmetric difference: root clocks cannot refer to absence. In other words, the clock system must be in disjunctive form [10].

We have now implemented code generation for poly-endochronous systems in Polychrony. This generation reuses techniques of distributed code generation, with rendez-vous management for synchronization constraints on clocks which are not placed in the same tree of clocks. The approach has been validated on several use cases running in parallel with time to time synchronization.

ANTIQUE Project-Team

7. New Results

7.1. Memory Abstraction

7.1.1. Abstraction of arrays based on non contiguous partitions

Participants: Jiangchao Liu, Xavier Rival [correspondant].

Abstract interpretation, Memory abstraction, Array abstract domains. In [2], 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.

7.2. Rule-based modeling

7.2.1. Reachability analysis via orthogonal sets of patterns

Participants: Kim Quyên Ly, Jérôme Feret [correspondant].

Rule-based modeling languages, as Kappa, allow for the description of very detailed mechanistic models. Yet, as the rules become more and more numerous, there is a need for formal methods to enhance the level of confidence in the models that are described with these languages. We develop abstract interpretation tools to capture invariants about the biochemical structure of bio-molecular species that may occur in a given model. In previous works, we have focused on the relationships between the states of the sites that belong to a same instance of a protein. This comes down to detect for a specific set of patterns, which ones may be reachable during the execution of the model. This paper [6], we generalize this approach to a broader family of abstract domains, that we call orthogonal sets of patterns. More precisely, an orthogonal set of patterns is obtained by refining recursively the information about some patterns containing a given protein, so as to partition of the set of occurrences of this protein in any mixture.

7.2.2. Local traces: an over-approximation of the behaviour of the proteins in rule-based models

Participants: Kim Quyên Ly, Jérôme Feret [correspondant].

Thanks to rule-based modelling languages, we can assemble large sets of mechanistic protein-protein interactions within integrated models. Our goal would be to understand how the behaviour of these systems emerges from these low-level interactions. Yet this is a quite long term challenge and it is desirable to offer intermediary levels of abstraction, so as to get a better understanding of the models and to increase our confidence within our mechanistic assumptions. In this paper [5], we propose an abstract interpretation of the behaviour of each protein, in isolation. Given a model written in Kappa, this abstraction computes for each kind of protein a transition system that describes which conformations this protein can take and how a protein can pass from one conformation to another one. Then, we use simplicial complexes to abstract away the interleaving order of the transformations between conformations that commute. As a result, we get a compact summary of the potential behaviour of each protein of the model.

7.3. Formal Derivation of Qualitative Dynamical Models from Biochemical Networks

Participants: Wassim Abou-Jaoudé, Denis Thieffry, Jérôme Feret [correspondant].

As technological advances allow a better identification of cellular networks, more and more molecular data are produced allowing the construction of detailed molecular interaction maps. One strategy to get insights into the dynamical properties of such systems is to derive compact dynamical models from these maps, in order to ease the analysis of their dynamics. Starting from a case study, we present in [1] a methodology for the derivation of qualitative dynamical models from biochemical networks. Properties are formalised using abstract interpretation. We first abstract states and traces by quotienting the number of instances of chemical species by intervals. Since this abstraction is too coarse to reproduce the properties of interest, we refine it by introducing additional constraints. The resulting abstraction is able to identify the dynamical properties of interest in our case study.

7.4. Taking Static Analysis to the Next Level: Proving the Absence of Run-Time Errors and Data Races with Astrée

Participants: Antoine Miné, Laurent Mauborgne, Xavier Rival, Jérôme Feret [correspondant], Patrick Cousot, Daniel Kästner, Stephan Wilhelm, Christian Ferdinand.

In [9], we present an extension of Astrée to concurrent C software. Astrée is a sound static analyzer for run-time errors previously limited to sequential C software. Our extension employs a scalable abstraction which covers all possible thread interleavings, and soundly reports all run-time errors and data races: when the analyzer does not report any alarm, the program is proven free from those classes of errors. We show how this extension is able to support a variety of operating systems (such as POSIX threads, ARINC 653, OSEK/AUTOSAR) and report on experimental results obtained on concurrent software from different domains, including large industrial software.

7.5. Stochastic mechanics of graph rewriting

Participants: Nicolas Behr, Vincent Danos, Ilias Garnier [correspondant].

We propose an algebraic approach to stochastic graph-rewriting which extends the classical construction of the Heisenberg-Weyl algebra and its canonical representation on the Fock space. Rules are seen as particular elements of an algebra of “diagrams": the diagram algebra D. Diagrams can be thought of as formal computational traces represented in partial time. They can be evaluated to normal diagrams (each corresponding to a rule) and generate an associative unital non-commutative algebra of rules: the rule algebra R. Evaluation becomes a morphism of unital associative algebras which maps general diagrams in D to normal ones in R. In this algebraic reformulation, usual distinctions between graph observables (real-valued maps on the set of graphs defined by counting subgraphs) and rules disappear. Instead, natural algebraic substructures of R arise: formal observables are seen as rules with equal left and right hand sides and form a commutative subalgebra, the ones counting subgraphs forming a sub-subalgebra of identity rules. Actual graph-rewriting is recovered as a canonical representation of the rule algebra as linear operators over the vector space generated by (isomorphism classes of) finite graphs. The construction of the representation is in close analogy with and subsumes the classical (multi-type bosonic) Fock space representation of the Heisenberg-Weyl algebra.

This shift of point of view, away from its canonical representation to the rule algebra itself, has unexpected consequences. We find that natural variants of the evaluation morphism map give rise to concepts of graph transformations hitherto not considered. These will be described in a separate paper [2]. In this extended abstract we limit ourselves to the simplest concept of double-pushout rewriting (DPO). We establish “jump-closure", i.e. that the sub-space of representations of formal graph observables is closed under the action of any rule set. It follows that for any rule set, one can derive a formal and self-consistent Kolmogorov backward equation for (representations of) formal observables.

This result and the fallowing ones, co-authored by Vincent Danos, were published in peer-reviewed international conferences and journals. Although the papers are on HAL, they are not imported in the bibtex file so we can’t cite them properly.

7.6. Giry and the machine

Participants: Fredrik Dahlqvist, Vincent Danos, Ilias Garnier [correspondant].

We present a general method – the Machine – to analyse and characterise in finitary terms natural transformations between well-known functors in the category Pol of Polish spaces. The method relies on a detailed analysis of the structure of Pol and a small set of categorical conditions on the domain and codomain functors. We apply the Machine to transformations from the Giry and positive measures functors to combinations of the Vietoris, multiset, Giry and positive measures functors. The multiset functor is shown to be defined in Pol and its properties established. We also show that for some combinations of these functors, there cannot exist more than one natural transformation between the functors, in particular the Giry monad has no natural transformations to itself apart from the identity. Finally we show how the Dirichlet and Poisson processes can be constructed with the Machine.

7.7. Robustly Parameterised Higher-Order Probabilistic Models

Participants: Fredrik Dahlqvist, Vincent Danos, Ilias Garnier [correspondant].

We present a method for constructing robustly parameterised families of higher-order probabilistic models. Parameter spaces and models are represented by certain classes of functors in the category of Polish spaces. Maps from parameter spaces to models (parameterisations) are continuous and natural transformations between such functors. Naturality ensures that parameterised models are invariant by change of granularity – i.e. that parameterisations are intrinsic. Continuity ensures that models are robust with respect to their parameterisation. Our method allows one to build models from a set of basic functors among which the Giry probabilistic functor, spaces of cadlag trajectories (in continuous and discrete time), multisets and compact powersets. These functors can be combined by guarded composition, product and coproduct. Parameter spaces range over the polynomial closure of Giry-like functors. Thus we obtain a class of robust parameterised models which includes the Dirichlet process, various point processes (random sequences with values in Polish spaces) and other classical objects of probability theory. By extending techniques developed in prior work, we show how to reduce the questions of existence, uniqueness, naturality, and continuity of a parameterised model to combinatorial questions only involving finite spaces.

7.8. Bayesian inversion by ω-complete cone duality

Participants: Fredrik Dahlqvist, Vincent Danos, Ilias Garnier [correspondant], Ohad Kammar.

The process of inverting Markov kernels relates to the important subject of Bayesian modelling and learning. In fact, Bayesian update is exactly kernel inversion. In this paper, we investigate how and when Markov kernels (aka stochastic relations, or probabilistic mappings, or simply kernels) can be inverted. We address the question both directly on the category of measurable spaces, and indirectly by interpreting kernels as Markov operators: For the direct option, we introduce a typed version of the category of Markov kernels and use the so-called 'disintegration of measures'. Here, one has to specialise to measurable spaces borne from a simple class of topological spaces -e.g. Polish spaces (other choices are possible). Our method and result greatly simplify a recent development in Ref. [4]. For the operator option, we use a cone version of the category of Markov operators (kernels seen as predicate transformers). That is to say, our linear operators are not just continuous, but are required to satisfy the stronger condition of being ω-chain-continuous.1 Prior work shows that one obtains an adjunction in the form of a pair of contravariant and inverse functors between the categories of L1- and L^∞-cones [3]. Inversion, seen through the operator prism, is just adjunction.2 No topological assumption is needed. We show that both categories (Markov kernels and ω-chain-continuous Markov operators) are related by a family of contravariant functors Tp for $1 \leq p \leq \infty$. The Tp's are Kleisli extensions of (duals of) conditional expectation functors introduced in Ref. [3]. With this bridge in place, we can prove that both notions of inversion agree when both defined: if f is a kernel, and f† its direct inverse, then $T\infty(f)† = T1(f†)$.

7.9. Continuous-time Markov chains as transformers of unbounded observables

Participants: Vincent Danos, Ilias Garnier [correspondant], Tobias Heindel, Jakob Simonsen.

We provide broad sufficient conditions for the com-putability of time-dependent averages of stochastic processes of the form f (Xt) where Xt is a continuous-time Markov chain (CTMC), and f is a real-valued function (aka an observable). We consider chains with values in a countable state space S, and possibly unbounded f s. Observables are seen as generalised predicates on S and chains are interpreted as transformers of such generalised predicates, mapping each observable f to a new observable Ptf defined as (Ptf)(x) = Ex(f (Xt)), which represents the mean value of f at time time t as a function of the initial state x. We obtain three results. First, the well-definedness of this operator interpretation is obtained for a large class of chains and observables by restricting Pt to judiciously chosen rescalings of the basic Banach space C0(S) of S-indexed sequences which vanish at infinity. We prove, under appropriate assumptions, that the restricted family Pt forms a strongly continuous operator semigroup (equivalently the time evolution map t $\rightarrow$ Pt is continuous w.r.t. the usual topology on bounded operators). The computability of the time evolution map follows by generic arguments of constructive analysis. A key point here is that the assumptions are flexible enough to accommodate unbounded observables, and we give explicit examples of such using stochastic Petri nets and stochastic string rewriting. Thirdly, we show that if the rate matrix (aka the q-matrix) of the CTMC is locally algebraic on a subspace containing f , the time evolution of projections t $\rightarrow$ (Ptf)(x) is PTIME computable for each x. These results provide a functional analytic alternative to Monte Carlo simulation as test bed for mean-field approximations, moment closure, and similar techniques that are fast, but lack absolute error guarantees.

7.10. Communities in socio-cognitive networks.

Participants: Vincent Danos, Ricardo Honorato-Zimmer [correspondant].

We investigate a recent network model which combines social and cognitive features. Each node in the social network holds a (possibly different) cognitive network that represent its beliefs. In this internal cognitive network a node denotes a concept and a link indicates whether the two linked concepts are taken to be of a similar or opposite nature. We show how these networks naturally organise into communities and use this to develop a method that detects communities in social networks. How they organise depends on the social structure and the ratio between the cognitive and social forces driving the propagation of beliefs.

7.11. Synchronous Balanced Analysis

Participants: Andreea Beica [correspondant], Vincent Danos.

When modeling Chemical Reaction Networks, a commonly used mathematical formalism is that of Petri Nets, with the usual interleaving execution semantics. We aim to substitute to a Chemical Reaction Network, especially a “growth” one (i.e., for which an exponential stationary phase exists), a piecewise synchronous approximation of the dynamics: a resource-allocation-centered Petri Net with maximal-step execution semantics. In the case of unimolecular chemical reactions, we prove the correctness of our method and show that it can be used either as an approximation of the dynamics, or as a method of constraining the reaction rate constants (an alternative to flux balance analysis, using an emergent formally defined notion of “growth rate” as the objective function), or a technique of refuting models.

7.12. Pointless learning

Participants: Florence Clerc, Fredrik Dahlqvist, Vincent Danos, Ilias Garnier [correspondant].

Bayesian inversion is at the heart of probabilistic programming and more generally machine learning. Understanding inversion is made difficult by the pointful (kernel-centric) point of view usually taken in the literature. We develop in a pointless (kernel-free) approach to inver- sion. While doing so, we revisit some foundational objects of probability theory, unravel their category-theoretical underpinnings and show how pointless Bayesian inversion sits naturally at the centre of this construction.

7.13. Survival of the fattest.

Participants: Andreea Beica [correspondant], Vincent Danos, Guillaume Terradot, Andrea Weisse.

Cells derive resources from their environments and use them to fuel the biosynthetic processes that determine cell growth. Depending on how responsive the biosynthetic processes are to the availability of intracellular resources, cells can build up different levels of resource storage. Here we use a recent mathematical model of the coarse-grained mechanisms that drive cellular growth to investigate the effects of cellular resource storage on growth. We show that, on the one hand, there is a cost associated with high levels of storage resulting from the loss of stored resources due to dilution. We further show that, on the other hand, high levels of storage can benefit cells in variable environments by increasing biomass production during transitions from one medium to another. Our results thus suggest that cells may face trade-offs in their maintenance of resource storage based on the frequency of environmental change.

7.14. The algebras of graph rewriting

Participants: Nicolas Behr, Vincent Danos, Ilias Garnier [correspondant], Tobias Heindel.

The concept of diagrammatic combinatorial Hopf algebras in the form introduced for describing the Heisenberg-Weyl algebra is extended to the case of so-called rule diagrams that present graph rewriting rules and their composites. The resulting rule diagram algebra may then be suitably restricted in four different ways to what we call the rule algebras, which are non-commutative, unital associative algebras that implement the algebra of compositions of graph rewriting rules. Notably, our framework reveals that there exist two more types of graph rewriting systems than previously known in the literature, and we present an analysis of the structure of the rule algebras as well as a form of Poincaré-Birkhoff-Witt theorem for the rule diagram algebra. Our work lays the foundation for a fundamentally new way of analyzing graph transformation systems, and embeds this very important concept from theoretical computer science firmly into the realm of mathematical combinatorics and statistical physics.

7.15. PSYNC: A partially synchronous language for fault-tolerant distributed algorithms

Participants: Cezara Drăgoi [correspondant], Thomas Henzinger [IST Austria, Austria], Damien Zufferey [MIT, CSAIL, USA].

Fault-tolerant distributed systems, Programming languages, Verification Fault-tolerant distributed algorithms play an important role in many critical/high-availability applications. These algorithms are notori- ously difficult to implement correctly, due to asynchronous com- munication and the occurrence of faults, such as the network drop- ping messages or computers crashing. We introduce PSYNC in [4], a domain specific language based on the Heard-Of model, which views asynchronous faulty systems as syn- chronous ones with an adversarial environment that simulates asyn- chrony and faults by dropping messages. We define a runtime sys- tem for PSYNC that efficiently executes on asynchronous networks. We formalize the relation between the runtime system and PSYNC in terms of observational refinement. The high-level lockstep ab- straction introduced by PSYNC simplifies the design and imple- mentation of fault-tolerant distributed algorithms and enables auto- mated formal verification. We have implemented an embedding of PSYNC in the SCALA programming language with a runtime system for asynchronous networks. We show the applicability of PSYNC by implementing several important fault-tolerant distributed algorithms and we com- pare the implementation of consensus algorithms in PSYNC against implementations in other languages in terms of code size, runtime efficiency, and verification.

CELTIQUE Project-Team

4. New Results

4.1. Monitoring attacker knowledge with information flow analysis

Participants: Thomas Jensen, Frédéric Besson.

Motivated by the problem of stateless web tracking (fingerprinting) we have investigated a novel approach to hybrid information flow monitoring by tracking the knowledge that an attacker can learn about secrets during a program execution. We have proposed a general framework for combining static and dynamic information flow analysis, based on a precise representation of attacker knowledge. This hybrid analysis computes a precise description of what an attacker learns about the initial configuration (and in particular the secret part of it) by observing a specific output. An interesting feature of this knowledge-based information flow analysis is that it can be used to improve other information flow control mechanisms, such as no-sensitive upgrade. The whole framework is accompanied by a formalisation of the theory in the Coq proof assistant [18].

4.2. Semantic analysis of functional specifications of system software

Participants: Thomas Jensen, Oana Andreescu, Pauline Bolignano.

We have developed a static analysis for correlating input and output values in functional specifications, written in a functional, strongly typed, high-level specification formalism developed by the SME Prove & Run. In the context of interactive formal verification of complex systems, much effort is spent on proving the preservation of the system invariants. However, most operations have a localized effect on the system. Identifying correlations (in particular equalities) between input and output can substantially ease the proof burden for the programmer. Our correlation analysis is a flow-sensitive interprocedural analysis that handles arrays, structures and variant data types, and which computes a conservative approximation of the equality between sub-structures of input and of output fragments [27]. In a separate strand of work, we have used abstraction-based techniques for structuring and simplifying the proof of simulation between a high-level and a low-level specification of memory management algorithms in a hypervisor [22]. Both strands of work was carried out and validated on system software (a micro-kernel and a hypervisor) developed using the formal approach defined by Prove & Run.

4.3. Certified Static Analyses

4.3.1. Certified Semantics and Analyses for JavaScript

Participants: Martin Bodin, Gurvan Cabon, Thomas Jensen, Alan Schmitt.

We have continued our work on the certification of the semantics of JavaScript and of analyses for JavaScript on three different fronts.

First, on the language side, we have developed at tool in collaboration with Arthur Charguéraud (Inria Saclay) and Thomas Wood (Imperial College) to interactively explore the specification of JavaScript. More precisely, we have written a compiler for a subset of OCaml to a subset of JavaScript that generates an interpreter that can be executed step by step, inspecting both the state of the interpreted program but also the state of the interpreter. We have used this compiler on the JavaScript interpreter extracted from our Coq semantics of JavaScript. The resulting tool is available here and a demo can be run here. The tool has been presented to the Ecma TC39 committee in charge of standardizing JavaScript. We are currently identifying the improvements required to make it useful for the standardization process.

Second, Bodin, Schmitt, and Jensen have designed an abstract domain based on separation logic to faithfully abstract JavaScript heaps. This domain is able to capture interlinked dynamic and extensible objects, a central feature of the JavaScript memory model. In addition, we have introduced the notion of *membranes* that let us correctly define abstractions in a way that is compatible both with separation logic and abstract interpretation. As an extension of last year's work [32], this approach is globally correct as soon as each rule is independently proven correct. This result illustrates the robustness of our approach to define certified abstract semantics based on pretty-big-step semantics. This work has not yet been published.

Third, Cabon and Schmitt are developing a framework to automatically derive an information-flow tracking semantics from a pretty-big-step semantics. We have manually shown the approach works for complex examples, and are currently proving it in Coq. This work is submitted for publication.

4.3.2. *Certified Analyses for C and lower-level programs*

Participants: Sandrine Blazy, David Pichardie, Alix Trieu.

We have continued our work on the static analyzer Verasco [37], based on abstract interpretation and operating over most of the ISO C 1999 language (excluding recursion and dynamic allocation). Verasco establishes the absence of run-time errors in the analyzed programs. It enjoys a modular architecture that supports the extensible combination of multiple abstract domains. We have extended the memory abstract domain (that takes as argument any standard numerical abstract domain), so that it finely tracks properties about memory contents, taking into account union types, pointer arithmetic and type casts [19]. This memory domain is implemented and verified inside the Coq proof assistant with respect to the CompCert compiler memory model.

Motivated by applications to security and high efficiency, we are reusing the Verasco static analyzer and the CompCert compiler in order to design a lightweight and automated methodology for validating on low-level intermediate representations the results of a source-level static analysis. Our methodology relies on two main ingredients: a relative-safety checker, an instance of a relational verifier which proves that a program is safer than another, and a transformation of programs into defensive form which verifies the analysis results at runtime.

4.4. Certified Compilation

Participants: Sandrine Blazy, Frédéric Besson, Pierre Wilke, Alexandre Dang.

The CompCert C compiler provides the formal guarantee that the observable behaviour of the compiled code improves on the observable behaviour of the source code. A first limitation of this guarantee is that if the source code goes wrong, i.e. does not have a well-defined behaviour, any compiled code is compliant. Another limitation is that CompCert 's notion of observable behaviour is restricted to IO events.

Over the past years, we have developed the semantics theory so that unlike CompCert but like Gcc, the binary representation of pointers can be manipulated much like integers and where memory is a finite resource. We have now a formally verified C compiler, CompCertS, which is essentially the CompCert compiler, albeit with a stronger formal guarantee. The semantics preservation theorem applies to a wider class of existing C programs and, therefore, their compiled version benefits from the formal guarantee of CompCertS. CompCertS preserves not only the observable behaviour of programs but also ensures that the memory consumption is preserved by the compiler. As a result, we have the formal guarantee that the compiled code requires no more memory than the source code. This ensures that the absence of stack-overflows is preserved by compilation.

The whole proof of CompCertS represents a significant proof-effort and the details can be found in Pierre Wilke's PhD thesis [39].

CompCertS also implements the Portable Software Fault Isolation approach pioneered by Kroll *et al.* [38]. The advantage of CompCertS is that the masking operation of pointers has a defined semantics and can therefore be directly reasoned about.

4.5. Mechanical Verification of SSA-based Compilation Techniques

Participants: Delphine Demange, Yon Fernandez de Retana, David Pichardie.

We have continued our work on the mechanical verification of SSA-based compilation techniques [30], [31], [36].

A crucial phase for efficient machine code generation is the destruction of a middle-end SSA-like IR. To this end, we have studied a variant of SSA, namely the Conventional SSA form, which simplifies the destruction back to non-SSA code (i.e. at the exit point of the middle-end). This had long remained a difficult problem, even in a non-verified environment. We formally defined and proved the properties of the generation of Conventional SSA. Finally, we implemented and proved correct a coalescing destruction of the Conventional SSA form, à la Boissinot et al. [33], where variables can be coalesced according to a refined notion of interference. Our CSSA-based, coalescing destruction allows us to coalesce more than 99% of introduced copies, on average, and leads to encouraging results concerning spilling and reloading during post-SSA allocation. This work has been published in [24].

4.6. Semantics for shared-memory concurrency

Participants: Gurvan Cabon, David Cachera, David Pichardie.

Modern multicore processor architectures and compilers of shared-memory concurrent programming languages provide only weak memory consistency guarantees. A *memory model* specifies which write action can be seen by a read action between concurrent threads.

In a previous work on the Java memory model [35], we defined in an axiomatic style, a memory model where we embed the reorderings of memory accesses directly in the semantics, so that formalizing optimizations and their correctness proof is easier.

This year, following a similar approach, we have studied the RMO (Relaxed- Memory Order) model. More precisely, we defined a new multibuffer operational semantics with write and read buffers. We also introduced an intermediate semantics inspired from Boudol et al. [34], where actions are reordered within a single pipeline. Finally, another model formalizes the reordering semantics in an axiomatic way. We fully proved the equivalence between the first two models and present a methodology for the remaining part. This work has been published in an international workshop [23].

4.7. Static analysis of functional programs using tree automata and term rewriting

Participant: Thomas Genet.

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. Regular tree languages are (possibly) infinite languages which can be finitely represented using tree automata. To over-approximate sets of reachable terms, the tools we develop use the Tree Automata Completion (TAC) algorithm 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. In [16], we have shown two results. The first result is a precision result guaranteeing that, for most of term rewriting systems known to have a regular set of reachable terms, TAC always compute it in an exact way. The second result shows that tree automata completion can be applied to functional programs to over-approximate their image. In particular, we have shown that tree automata completion computes a safe over-approximation of the image of any first-order, purely functional, complete and terminating program. Now, our first next objective is to demonstrate the accuracy of those regular approximations to perform lightweight formal verification of functional programs. The second objective is to lift those results to higher-order purely functional programs.

DEDUCTEAM Team

6. New Results

6.1. Dedukti

A. Assaf, G. Burel, R. Cauderlier, D. Delahaye, G. Dowek, C. Dubois, F. Gilbert, P. Halmagrand, O. Hermant, and R. Saillard, have finished writing a general presentation of the Dedukti system. This paper is submitted for publication.

Under the supervision of P. Halmagrand and G. Burel, D. Pham worked on the conversion of TSTP proof traces, as produced by automated theorem provers such as E, Zipperposition or Vampire, into Dedukti proofs. To that purpose, he modified Zenon modulo so that it reads TSTP files and tries to reprove the proof steps given by the trace.

R. Cauderlier defended his PhD thesis on the translation of programming languages to Dedukti and interoperability of proof systems [11]. He also presented his work on the use of Dedukti for rewriting-based proof transformation [15] and on the translation of FoCaLiZe in Dedukti [16]

6.2. Proof theory

G. Dowek and Y. Jiang have finished a paper on co-inductive and inductive complementation of inference systems. This paper is submitted for publication.

The paper of G. Dowek on the introduction of rules and derivations in a logic course has been published [24].

F. Gilbert has finished a paper on the automated constructivization of proofs, to appear in the proceedings of FOSSACS'17.

F. Thiré is working on the translation of the Fermat little theorem proof written in Matita to a proof written in HOL. A part of this work is developed in its internship report [25]. He is continuing this translation during his PhD thesis.

6.3. B Method

The B Method is a formal method mainly used in the railway industry to specify and develop safety-critical software. To guarantee the consistency of a B project, one decisive challenge is to show correct a large amount of proof obligations, which are mathematical formulas expressed in a classical set theory extended with a specific type system. To improve automated theorem proving in the B Method, Pierre Halmagrand proposes [17], [12] to use a first-order sequent calculus extended with a polymorphic type system, which is in particular the output proof-format of the tableau-based automated theorem prover Zenon. After stating some modifications of the B syntax and defining a sound elimination of comprehension sets, he proposes a translation of B formulas into a polymorphic first-order logic format. Then, he introduces the typed sequent calculus used by Zenon, and shows that Zenon proofs can be translated to proofs of the initial B formulas in the B proof system.

6.4. Termination

F. Blanqui revised his paper on "size-based termination of higher-order rewrite systems" submitted to the Journal of Functional Programming [23]. This paper is concerned with the termination, in Church' simply-typed λ-calculus, of the combination of β-reduction and arbitrary user-defined rewrite rules fired using matching modulo α-congruence only. Several authors have devised termination criteria for fixpoint-based function definitions using deduction rules for bounding the size of terms inhabiting inductively defined types, where the size of a term is (roughly speaking) the set-theoretical height of the tree representation of its normal form. In the present paper, we extend this approach to rewriting-based function definitions and more general notions of size.

G. Dowek has finished writing a paper on the notion of model and its application to termination proofs for the $\lambda\Pi$-calculus modulo theory. This paper is submitted for publication.

6.5. Confluence

In $\lambda\Pi$modulo, congruences are expressed by rewrite rules that must enjoy precise properties, notably confluence, strong normalization, and type preservation. A difficulty is that these properties depend on each other in calculi of dependent types. To break the circularity, confluence is usually proved separately on untyped terms. A another difficulty then arises : computation do not terminate on untyped terms. A result of van Oostrom allows to show confluence of non-terminating left-linear higher-order rules, provided their critical pairs are development closed. This result was used for the encodings of HOL, Matita, and Coq up to version 8.4. Encoding the most recent version of Coq requires rules for universes that are confluent on open terms, while confluence on ground terms sufficed before. The encoding we recently developed for this new version of Coq has higher-order rules which are not left-linear, use pattern matching modulo associativity, commutativity and identity, and whose (joinable) critical pairs are not development closed. We have therefore developed a new powerful result for proving confluence of that sort of rules provided non-linear variables can only be instantiated by first-order expressions [18], [19].

6.6. Physics and computation

The paper of G. Dowek and P. Arrighi Free fall and cellular automata has been published [13]. As a sequel of this paper, G. Dowek and P. Arrighi have written a short note [22].

A. Díaz-Caro and G. Dowek have developed a new typing system for quantum λ-calculus allowing to distinguish between pure states and superpositions.

Under the supervision of S. Martiel and P. Arrighi, C. Chouteau worked on a particular notion of covariance in the model of causal graph dynamics. Causal graph dynamics are graph transformations constrained by Physics-inspired symmetries. The particular object of study of this internship was a restriction of this model to physical transformations of discrete geometrical spaces.

GALLIUM Project-Team

7. New Results

7.1. Formal verification of compilers and static analyzers

7.1.1. The CompCert formally-verified compiler

Participants: Xavier Leroy, Bernhard Schommer [AbsInt GmbH], Jacques-Henri Jourdan.

In the context of our work on compiler verification (§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 [7]. This compiler comprises a back-end, which translates the Cminor intermediate language to PowerPC assembly, and is reusable for source languages other than C [6]; and a front-end, which translates the CompCert C subset of C to Cminor. The compiler is mostly written within the specification language of the Coq proof assistant, out of which Coq's extraction facility generates executable OCaml 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, the CompCert C compiler was improved in several directions:

- The proof of semantic preservation was extended to account for separate compilation and linking. (See section 7.1.2 .)
- Support for 64-bit target processors was added, while keeping the original support for 32-bit processors. The x86 code generator, initially 32-bit only, was extended to handle x86 64-bit as well.
- The generation of DWARF debugging information in `-g` mode, developed last year for PowerPC, is now available for ARM and x86 as well.
- The semantics of conversions from pointer types to the `_Bool` type is fully defined again. (It was made temporarily undefined while addressing issues with comparisons between the null pointer and out-of-bound pointers.)
- More features of ISO C 2011 are supported, such as the `_Noreturn` attribute, or anonymous members of struct and union types.
- As a result of his research on implementing a correct parser for the C language (§7.1.5), Jacques-Henri Jourdan improved the implementation of the parser.

Version 2.7 of CompCert was released in June 2016, incorporating most of these enhancements, with the exception of 64-bit processor support and anonymous members, which will be released Q1 2017.

7.1.2. Separate compilation and linking in CompCert

Participants: Xavier Leroy, Chung-Kil Hur [KAIST, Seoul], Jeehoon Kang [KAIST, Seoul].

Separate compilation (of multiple C source files into multiple object files, followed by linking of the object files to produce the final executable program) has been supported for a long time by the CompCert implementation, but it was not accounted for by CompCert's correctness proof. That proof established semantic preservation in the case of a single, monolithic C source file which is compiled at once to produce the final executable, but not in the more general case of separate compilation and linking.

Version 2.7 of CompCert, released this year, extends the proof of semantic preservation in order to account for separate compilation and linking. It follows the approach described by Kang, Kim, Hur, Dreyer and Vafeiadis in their POPL 2016 paper [47] and prototyped by Kang on CompCert 2.4. In this approach, the proof considers a set of C compilation units, separately compiled to assembly then linked, and shows that the resulting assembly program preserves the semantics of the C program that would be obtained by syntactic linking of the source C compilation units. The simplicity of this approach follows from the fact that semantic preservation is still shown between whole programs (after linking); there is no need to give semantics to individual compilation units. Xavier Leroy integrated the approach of Kang *et al.* into the CompCert development, and extended it to several new optimization passes that were not present in Kang's prototype implementation.

7.1.3. Separation logic assertions for compiler verification

Participants: Xavier Leroy, Timothy Bourke [EPI Parkas], Lélio Brun [EPI Parkas], Maxime Dénès [EPI Marelle].

Separation logic is a powerful tool to reason about imperative programs. It is a Hoare-style program logic where preconditions and postconditions are assertions about the contents of mutable state. Those assertions are built in a compositional manner using a separating conjunction operator.

While effective to prove the correctness of a given program, separation logic and program logics in general are less effective to prove the correctness of a compiler or of a program transformation, in particular because it is difficult to show preservation of termination. The alternative approach that we investigated this year consists in using the assertion language of separation logic, and in particular its separating conjunction, in the context of a conventional, CompCert-style proof of semantic preservation based on simulation diagrams. Assertions from separation logic make it possible to state the invariant that relates the memory states of the program before and after the transformation in a compositional manner, simplifying the proof that this invariant is preserved through execution steps.

This approach was developed and experimentally evaluated in in three case studies.

The first case study was part of project CEEC and consisted in verifying a code generator from a domain-specific, purely-functional intermediate language down to the Clight language of CompCert. Xavier Leroy and Maxime Dénès used ad-hoc separation logic assertions to describe the memory states of the generated Clight programs, and in particular the use of pointers to return multiple function results via "out" parameters.

The second case study was a complete rewrite of the Stacking pass of the CompCert back-end and of its correctness proof, as part of the new support for 64-bit architectures (§7.1.2). For this new proof, Xavier Leroy reused and improved the separation logic assertions of the previous project, using a shallow embedding into Coq instead of a deep embedding. Separating conjunctions are used to specify the layout and current contents of the stack frames for every compiled function, in a way that accommodates 32- and 64-bit registers and pointer values equally well.

The third use takes place in the context of the verified Lustre-to-C compiler in development at team Parkas (see their activity report). The final pass of this compiler translates a simple object-oriented intermediate language, Obc, to CompCert's Clight. Timothy Bourke and Lélio Brun used the separation logic assertions from the second project to specify and reason about the Clight memory layout of the Obc nested objects. Timothy Bourke and Xavier Leroy also extended the separation logic with a "magic wand" operator. A paper on this compiler verification project is under review.

7.1.4. Formal verification of static analyzers based on abstract interpretation

Participants: Jacques-Henri Jourdan, Xavier Leroy, Sandrine Blazy [team Celtique], David Pichardie [team Celtique], Sylvain Boulmé [Grenoble INP, VERIMAG], Alexis Fouilhé [Université Joseph Fourier de Grenoble, VERIMAG], Michaël Périn [Université Joseph Fourier de Grenoble, VERIMAG].

In the context of the Verasco ANR 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 code.

This year, Jacques-Henri Jourdan published in his PhD thesis [11] an in-depth description of the mode of operation of the current version of the Verasco static analyzer. He also presented at the NSAD workshop [24] the new algorithms used in Verasco for the abstract domain of Octagons that he developed in 2015.

7.1.5. Correct parsing of C using LR(1)

Participants: Jacques-Henri Jourdan, François Pottier.

The C programming language cannot be parsed directly using LR technology. Indeed, the grammar described in the C standard exhibits ambiguities which are addressed in English prose. On the implementation side, it is known from the folklore that one can in fact use an LALR(1) parser to parse C, provided one sets up a so-called "lexer hack" to perform on-the-fly disambiguation of tokens, guided by the current state of the parser.

However, Jacques-Henri Jourdan and François Pottier found that a correct implementation of the "lexer hack" is, surprisingly, difficult. To clarify this situation, they implemented a reference C11 parser using Menhir. They invented new techniques that improve and simplify the "lexer hack", so as to write correct yet reasonably simple C11 parsers. They also created a test suite of C programs that exhibit particularly challenging corner cases. This work is described in a paper that is currently under review.

7.1.6. A SPARK front-end for CompCert

Participants: Pierre Courtieu, Zhi Zang [Kansas University].

SPARK is a language, and a platform, dedicated to developing and verifying critical software. It is a subset of the Ada language. It shares with Ada a strict typing discipline and gives strict guarantees in terms of safety. SPARK goes one step further by disallowing certain "dangerous" features, that is, those that are too difficult to statically analyze (aliasing, references, etc). Given its dedication to safety critical software, we think that the SPARK platform can benefit from a certified compiler. We are working on adding a SPARK front-end to the CompCert verified compiler.

Defining a semantics for SPARK in Coq is previous joint work with Zhi Zang. The current front-end is based on this semantics. The compiler has been written and tested and the proofs of correctness are nearing completion.

7.2. Language design and type systems

7.2.1. Types with unique inhabitants for code inference

Participants: Gabriel Scherer [Northeastern University], Didier Rémy.

Some programming language features (coercions, type-classes, implicits) rely on inferring a part of the code that is determined by its usage context. In order to better understand the theoretical underpinnings of this mechanism, we ask: when is it the case that there is a unique program that could have been guessed, or in other words, that all possible guesses result in equivalent program fragments? Which types have a unique inhabitant?

To approach the question of uniqueness, we build on work in proof theory on canonical representations of proofs. Using the proofs-as-programs correspondence, we adapt the logical technique of focusing to obtain canonical program representations.

In the setting of simply-typed lambda-calculus with sums, equipped with the strong $\beta\eta$-equivalence, we show that uniqueness is decidable. We present a saturating focused logic that introduces irreducible cuts on positive types "as soon as possible". Goal-directed proof search in this logic gives an effective algorithm that returns either zero, one or two distinct inhabitants for any given type.

This work, which was previously presented at a conference [56] and was the main part of Scherer's PhD dissertation [12], has been submitted for journal publication.

7.2.2. Refactoring with ornaments in ML

Participants: Thomas Williams, Didier Rémy.

Thomas Williams and Didier Rémy continued working on ornaments for program refactoring and program transformation in ML. Ornaments have been introduced as a way to describe some changes in data type definitions that preserve their recursive structure, reorganizing, adding, or dropping some pieces of data. After a new data structure has been described as an ornament of an older one, some functions operating on the bare structure can be partially or sometimes totally lifted into functions operating on the ornamented structure.

We have continued working on the decomposition of the algorithm in several steps. Using ornament inference, we first elaborate an ML program into a generic program, which can be seen as a template for all possible liftings of the original program. The generic program is defined in a superset of ML. It can then be instantiated with specific ornaments, and simplified back into an ML program. We studied the semantics of this intermediate language and used them to prove the correctness of the lifting, using logical relations techniques. A paper describing this process was submitted to PLDI.

On the practical side, we updated our prototype implementation to match our theoretical presentation: we create the generic program, then instantiate it. We then simplify the resulting term so that it remains readable to the programmer, and output an ML program. In the case of refactoring (the representation of a data type is modified without adding any data), the transformation is still fully automatic.

7.3. Shared-memory parallelism

7.3.1. Weak memory models

Participants: Luc Maranget, Jade Alglave [University College London–Microsoft Research, UK], Patrick Cousot [New York University], Andrea Parri [Sant'Anna School of Advanced Studies, Pisa, Italy].

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 executions of its constituent 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. Luc Maranget took part in an international research effort to define the semantics of the computers of the multi-core era, and more generally of shared-memory parallel devices or languages, with a clear focus on devices.

More precisely, in 2016, Luc Maranget pursued his collaboration with Jade Alglave and Patrick Cousot to extend "Cats", a domain-specific language for defining and executing weak memory models. Last year, a long article that presents a precise semantics for "Cats" and a study and formalisation of the HSA memory model was submitted. (The Heterogeneous System Architecture foundation is an industry standards body targeting heterogeneous computing devices.) As this article was rejected, a new paper, focused on the "Cats" semantics, was submitted this year, while the definition of the HSA memory model was made available on the web site of the HSA foundation (http://www.hsafoundation.com/standards/).

This year, our team hosted Andrea Parri, a Ph.D. student (supervised by Mauro Marinoni at Sant'Anna School of Advanced Studies, Pisa, Italy), for six months. Luc Maranget and Andrea Parri collaborated with Paul McKenney (IBM), Alan Stern (Harvard University) and Jade Alglave on the definition of a memory model for the Linux kernel. A preliminary version of this work was presented by Paul McKenney at the *2016 Linux Conference Europe*. While invited at the Dagstuhl seminar "*Concurrency with Weak Memory Models...*", Luc Maranget demonstrated the Diy toolsuite and the "Cats" language. It is worth noting that Cats models are being used independently of us by other researchers, most notably by Yatin Manerkar and Caroline J. Trippel (Princeton University) who discovered an anomaly in the published compilation scheme of the C11 language down to the Power architecture.

Luc Maranget also co-authored a paper that will be presented at POPL 2017 [23]. This work describes memory-model-aware "mixed-size" semantics for the ARMv8 architecture and for the C11 and Sequential Consistency models. A mixed-size semantics accounts for the behaviour of systems that access memory at different granularity levels (bytes, words, etc.) This is joint work with many researchers, including Shaked Flur and other members of Peter Sewell's team (University of Cambridge) as well as Mark Batty (University of Kent).

7.3.2. Algorithms and data structures for parallel computing

Participants: Umut Acar, Vitalii Aksenov, Arthur Charguéraud, Adrien Guatto, Michael Rainey, Filip Sieczkowski.

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 computation 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 four main results this year.

Our first result is a calculus for parallel computing on hardware shared-memory computers such as modern multicores. Many languages for writing parallel programs have been developed. These languages offer several distinct abstractions for parallelism, such as fork-join, async-finish, futures, etc. While they may seem similar, these abstractions lead to different semantics, language design and implementation decisions. In this project, we consider the question of whether it would be possible to unify these approaches to parallelism. To this end, we propose a calculus, called the *DAG-calculus*, which can encode existing approaches to parallelism based on fork-join, async-finish, and futures, and possibly others. We have shown that the approach is realistic by presenting an implementation in C++ and by performing an empirical evaluation. This work was presented at ICFP 2016 [18].

Our second result is a concurrent data structure that may be used to efficiently determine when a concurrently-updated counter reaches the value zero. Our data structure extends an existing data structure called SNZI [44]. While the latter imposes a fixed number of threads, our structure is able to dynamically grow in response to the increasing degree of concurrency in the system. We use our dynamic non-zero indicator data structure to derive an efficient runtime representation of async/finish programs. The async/finish paradigm for expressing parallelism is one that, in the past decade, has become a part of many research-language implementations (e.g. X10) and is now gaining traction in a number of mainstream languages, most notably Java. The implementation of async/finish is challenging because the finish-block mechanism permits, and even encourages, computations in which a large number of threads are required to synchronize on shared barriers, and this number is not statically known. We present an implementation of async/finish and prove that, in a model that takes contention into account, the cost of synchronization of the async-ed threads is amortized constant time, regardless of the number of threads. We also present experimental evaluation suggesting that the approach performs well in practice. This work has been accepted for publication at PPoPP [17].

Our third result is an extended, polished presentation of our prior work on granularity control for parallel algorithms using user-provided complexity functions. Granularity control denotes the problem of controlling the size of parallel threads created in implicitly parallel programs. If small threads are executed in parallel, the overheads due to thread creation can overwhelm the benefits of parallelism. If large threads are executed sequentially, processors may spin idle. In our work, we show that, if we have an oracle able to approximately predict the execution time of every sub-task, then there exists a strategy that delivers provably good performance. Moreover, we present empirical results showing that, for simple recursive divide-and-conquer programs, we are able to implement such an oracle simply by requiring the user to annotate functions with their asymptotic complexity. The idea is to estimate the constant factors that apply by conducting measures at runtime. This work is described in depth in an article published in the Journal of Functional Programming (JFP) [13].

Our fourth result is an extension of our aforementioned granularity control approach, with three major additions. First, we have developed an algorithm that ensures convergence of the estimators associated with the constant factors for all fork-join programs, and not just for a small class of programs. Second, we have built a theoretical analysis establishing bounds for the overall overheads of the convergence phase. Third, we have developed a C++ implementation accompanied with an extensive experimental study covering several benchmarks from the Problem Based Benchmark Suite (PBBS), a collection of high-quality parallel algorithms that delivers state-of-the-art performance. Even though our approach does not leverage a specific compiler and does not require any magic constant to be hard-coded in the source programs, our code either matches or exceeds the performance of the authors' original, hand-tuned codes. An article describing this work is in preparation.

7.4. The OCaml language and system

7.4.1. OCaml

Participants: Damien Doligez, Alain Frisch [Lexifi SAS], Jacques Garrigue [University of Nagoya], Sébastien Hinderer, Fabrice Le Fessant, Xavier Leroy, Luc Maranget, Gabriel Scherer, Mark Shinwell [Jane Street], Leo White [Jane Street], Jeremy Yallop [OCaml Labs, Cambridge University].

This year, we released versions 4.03.0 and 4.04.0 of the OCaml system. These are major releases that introduce a large number of new features. The most important features are:

- A new optimization subsystem called *flambda*, which does inlining and specialization of functions as well as static allocation of some data structures, etc.
- *ephemerons*: a generalization of weak pointers that is better suited for memoization of mutually-recursive functions.
- A fine-grained memory profiler to help programmers understand the allocation behavior of their programs.
- *unboxed types*: a user-controlled optimized representation for some simple data types.

7.4.2. Infrastructure for OCaml

Participant: Sébastien Hinderer.

Sébastien Hinderer worked on improving the test infrastructure of the OCaml compiler. These tests aim at verifying that the compiler works as expected. Currently, they are driven by a set of Makefiles which are hard to maintain and extend and make it difficult to add new tests. Sébastien developed the `ocamltest` driver, which parses test descriptions written in a domain-specific language and runs the appropriate tests.

Sébastien Hinderer also worked on merging the Makefiles used for building the compiler under Unix and Windows. The existence of separate sets of Makefiles, which is the result of a long development history, makes it especially hard to maintain and extend the compiler's build system. Sébastien worked on eliminating this redundancy, so that a single build system can be used on every platform. This is a prerequisite for using the GNU `autoconf` tools and for building easy-to-use cross-compilers for OCaml. A cross-compiler is required, for instance, to build iOS apps using OCaml.

7.4.3. Continuous integration of OCaml packages

Participant: Fabrice Le Fessant.

OPAM is a repository of OCaml source packages. It is now advertised as the official way of installing the OCaml distribution. To maintain a high level of quality for the thousands of source packages distributed in the repository, it is crucial to provide feedback to the developers on the impact of their modifications to the repository, in real-time, despite the high churn and the cascading costs of package recompilations.

We have designed and prototyped a simple modular architecture for a service that monitors the OPAM repository, and triggers recompilation of packages that are impacted by the latest modifications to the repository, for all major and minor OCaml versions since 3.12.1. Previous attempts to design such a system have failed to scale, although they targeted cloud systems of thousands of virtual machines. On the contrary, the new prototype has been deployed on a single quadcore server, and has been able to follow the OPAM repository for eight months, providing feedback in almost real-time. To achieve such a result, it uses many optimizations and caching techniques, to make recompilations as incremental as possible [37].

7.4.4. Global analyses of OCaml programs

Participants: Thomas Blanc [ENSTA-ParisTech & OCamlPro], Pierre Chambart [OCamlPro], Vincent Laviron [OCamlPro], Fabrice Le Fessant, Michel Mauny.

Exception handling in OCaml can be used for managing and reporting errors, as well as to express complex control flow constructs. As such, exceptions can be the source of errors, when, for instance, a function that may raise an exception is called in a context where this exception cannot be handled. In such situations, the program may fail unexpectedly, and the source of the error can be difficult to identify.

This work aims at performing global static analyses of OCaml programs using abstract interpretation techniques, with a particular focus on the detection of uncaught exceptions. Starting from one of the OCaml intermediate languages, we produce a hypergraph that represents the program to be analyzed. Each node of this hypergraph is a program state and each edge is an operation. Operations that may or may not raise an exception (such as function calls) have one or two successors. A fixpoint iteration is then performed on the graph, where function application edges are dynamically replaced by the corresponding subgraphs. In essence, environment information is propagated through the graph, adding at each node a superset of all possible values of each variable, until no additional information can be found. A description of the framework was presented at the 2015 OCaml workshop. We expect concrete results as well as Thomas Blanc's thesis manuscript during 2017.

7.4.5. Type-checking the OCaml intermediate languages

Participants: Pierrick Couderc [ENSTA-ParisTech & OCamlPro], Grégoire Henry [OCamlPro], Fabrice Le fessant, Michel Mauny.

This work aims at propagating type information through the intermediate languages used by the OCaml compiler. We started by the design and implementation of a consistency checker of the type-annotated abstract syntax trees (TASTs) produced by the OCaml compiler. It appears that, when presented as inference rules, the different cases of this TAST checker can be read as the rules of the OCaml type system. Proving the correctness of (part of) the checker would prove the soundness of the corresponding part of the OCaml type system. A preliminary report on this work has been presented at the 17th Symposium on Trends in Functional Programming (TFP 2016).

7.4.6. Optimizing OCaml for satisfiability problems

Participants: Sylvain Conchon [LRI, Univ. Paris Sud], Albin Coquereau [ENSTA-ParisTech], Fabrice Le fessant, Michel Mauny.

This work aims at improving the performance of the Alt-Ergo SMT solver, implemented in OCaml. For safety reasons, the implementation of Alt-Ergo uses as much as possible a functional programming style and persistent data structures, which are sometimes less efficient that the imperative style and mutable data structures. We would like to first obtain a better understanding of the OCaml memory and cache behavior, so as to understand where efficiency could be gained, and then design dedicated data structures (for instance, semi-persistent data structures) and compare their efficiency to the current ones. This work is still at a preliminary stage: we have selected benchmarks and profiled their execution in order to discover sources of inefficiency.

7.4.7. Type compatibility checking for dynamically loaded OCaml data

Participants: Florent Balestrieri [ENSTA-ParisTech], Michel Mauny.

The SecurOCaml project (FUI 18) aims at enhancing the OCaml language and environment in order to make it more suitable for building secure applications, following recommendations published by the French ANSSI in 2013. Michel Mauny and Florent Balistrieri (ENSTA-ParisTech) represent ENSTA-Paristech in this project for the two-year period 2016-2017.

The goal of this first year was to design and produce an effective OCaml implementation that checks whether a memory graph – typically the result obtained by un-marshalling some data – is compatible with a given OCaml type, following the algorithm designed by Henry *et al.* in 2012. As the algorithm needs a runtime representation of OCaml types, Florent Balestrieri implemented a library for generic programming in OCaml [21]. He also implemented a type-checker which, when given a type and a memory graph, checks whether the former could be the type of the latter. The algorithm handles sharing and polymorphism, but currently supports neither functional values nor existential types.

7.4.8. Pattern matching

Participants: Luc Maranget, Gabriel Scherer [Northeastern University, Boston], Thomas Réfis [Jane Street LLC].

A new pattern matching diagnostic message, which should help OCaml programmers to detect rare but vicious programming errors, was integrated in the yearly release of the OCaml compiler, and was presented at the OCaml Users and Developers Workshop [39].

7.4.9. Error diagnosis in Menhir parsers

Participant: François Pottier.

In 2015, François Pottier proposed a reachability algorithm for LR automata, which he implemented in the Menhir parser generator. He applied this approach to the C grammar in the front-end of the CompCert compiler, therefore allowing CompCert to produce better syntax error messages. This work has been presented at the conferences JFLA 2016 [31] and CC 2016 [26].

7.5. Software specification and verification

7.5.1. Step-indexing in program logics

Participant: Filip Sieczkowski.

Filip Sieczkowski pursued a line of work focused on techniques for formal reasoning about programs, in joint work with Lars Birkedal (Aarhus University) and Kasper Svendsen (Cambridge University). A modern and successful approach to grounding programs logics is to rely on so-called step-indexed models. Filip and his co-authors solved a problem that arises in most step-indexed models, due to a tight coupling between the unfoldings of a recursive domain equation and evaluation steps. Their approach is based on the use of transfinite step-indexing. This work appeared at ESOP 2016 [29].

7.5.2. TLA+

Participants: Damien Doligez, Leslie Lamport [Microsoft Research], Martin Riener [team VeriDis], Stephan Merz [team VeriDis].

Damien Doligez is head of the “Tools for Proofs” team in the Microsoft-Inria Joint Centre. The aim of this project is to extend the TLA+ language with a formal language for hierarchical proofs, formalizing Lamport’s ideas [48], and to build tools for writing TLA+ specifications and mechanically checking the proofs.

Our rewrite of the TLAPS tools is almost done and we hope to do a first release in the first quarter of 2017.

7.5.3. Hash tables and iterators: a case study in program verification

Participant: François Pottier.

In the setting of the Vocal ANR project, François Pottier developed the the specification and proof of an (imperative, sequential) hash table implementation, as found in the module `Hashtbl` of OCaml's standard library. This data structure supports the usual dictionary operations (insertion, lookup, and so on), as well as iteration via folds and iterators. The code was verified using higher-order separation logic, embedded in Coq, via Charguéraud's CFML tool and library. This work was presented at CPP 2017 [27]. It can be viewed as a case study that should help prepare the way for verifying other modules in the Vocal library.

7.5.4. Read-only permissions in separation logic

Participants: Arthur Charguéraud, François Pottier.

Separation Logic, as currently implemented in Charguéraud's CFML tool and library, imposes a simple ownership discipline on mutable heap-allocated data structures: a thread either has full read-write access to a data structure, or has no access at all. This implies, for instance, that two threads cannot temporarily share read-only access to a data structure. There exist more flexible disciplines in the literature, such as "fractional permissions" and "share algebras", but they are much more complex.

In the setting of the Vocal ANR project, Arthur Charguéraud and François Pottier noted that it would be desirable to define an extension of Separation Logic that allows temporary shared read-only access, yet remains very simple. They proposed a general mechanism for temporarily converting any assertion (or "permission") to a read-only form. The metatheory of this proposal has been verified in Coq. This work will be presented at ESOP 2017 [42].

Charguéraud and Pottier believe that this mechanism should allow more concise specifications and proofs. This remains to be confirmed, in future work, via an implementation in CFML and case studies in the Vocal project.

7.5.5. Formal reasoning about asymptotic complexity

Participants: Armaël Guéneau, Arthur Charguéraud, François Pottier.

Armaël Guéneau started his Ph.D. at Gallium in September 2016, supervised by Arthur Charguéraud and François Pottier. In the line of his previous M2 internship at Gallium, he continued his work on asymptotic reasoning in Coq. The challenge is to give a formal definition of the well-known big-O notation, covering both single-variable and multiple-variable scenarios, to establish its fundamental properties, and to define tactics that make asymptotic reasoning as convenient in Coq as it seemingly is on paper. The ultimate goal is to apply these techniques to machine-checked proofs of the asymptotic time complexity of programs.

7.5.6. Certified distributed algorithms for autonomous mobile robots

Participant: Pierre Courtieu.

The variety and complexity of the tasks that can be performed by autonomous robots are increasing. Many applications envision groups of mobile robots that self-organise and cooperate toward the resolution of common objectives, in the absence of any central coordinating authority.

Pierre Courtieu is elaborating a verification platform, based on Coq, for distributed algorithms for autonomous robots. (This is joint work with Xavier Urbain, Sebastien Tixeuil and Lionel Rieg.) As part of this effort, Pierre Courtieu designed and verified a protocol for mobile robots that achieves the "gathering" task in all cases where it has not been proved impossible [34], [35].

MARELLE Project-Team

5. New Results

5.1. Implementing Theorem Proving in Higher Order Logic Programming

Participants: Enrico Tassi, Cvetan Dunchev [University of Bologna], Ferruccio Guidi [University of Bologna], Claudio Sacerdoti Coen [University of Bologna].

We carried on our experiments with extensions of λ-prolog, based on the ELPI tool that we developed, in particular concerning implementations of higher-order logic and type theory in this context. This work lead to publication in June at LFMTP'16 [14] and to a preliminary report [25].

5.2. Coqoon: An IDE for interactive proof development in Coq

Participants: Enrico Tassi, Alexander Faithfull [ITU Copenhagen], Jesper Bengtson [ITU Copenhagen], Carst Tankink.

We carried on our experiments with the Coqoon integrated development environment. This lead to a preliminary report submitted for publication [24]

5.3. A book on mathematical components

Participants: Enrico Tassi, Yves Bertot, Laurence Rideau, Assia Mahboubi, Georges Gonthier.

As an effort to lower the entry barrier to use a structured library of formalized mathematics, we wrote a book explaining the principles of ssreflect and mathematical components. This book-in-the-making is available on github at https://math-comp.github.io/mcb/ and we plan to make it evolve as we teach schools on using the library and we gather feedback from readers and users.

5.4. Proofs of transcendence

Participants: Sophie Bernard, Yves Bertot, Laurence Rideau.

In the previous year, we developed formally verified proofs that e and π are transcendental. This result was published this year at the CPP conference (Certified Programs and Proofs) [12]. Since October, as part of the PhD of Sophie Bernard, we are working on the generalisation of these proofs, in order to prove the Lindemann theorem that states that no algebraic spans of exponentials of algebraic numbers can be equal to zero under some assumptions.

5.5. Cubical type theory and univalent foundations

Participants: Cyril Cohen, Anders Mörtberg, Benedikt Ahrens [ASCOLA project-team, Inria and LINA Nantes], Mark Bickford [Cornell Unversity, USA], Thierry Coquand [Chalmers and Göteborg University, Sweden], Ralph Matthes [CNRS, University of Toulouse].

This work mainly concerns Univalent Foundations and Homotopy Type Theory which builds on recently discovered connections between type theory and abstract homotopy theory. The main question we have been working on lately is finding a computational interpretation for the univalence axiom, the main fruit of this work is a recent paper on, and implementation of, cubical type theory [23] which provides a constructive justification for this axiom. The code is visible at https://github.com/mortberg/cubicaltt. The last year Anders Mörtberg has been working together with Mark Bickford at Cornell University and Thierry Coquand at University of Gothenburg and Chalmers University of Technology on the formal verification of this model in the Nuprl proof assistant, this code is visible at http://www.nuprl.org/wip/Mathematics/cubical!type!theory/index.html.

Anders Mörtberg also recently visited Thierry Coquand to start a collaboration on the formalization of this model in the UniMath system implemented in Coq. Together with Benedikt Ahrens in the Ascola team at Inria Nantes and Ralph Matthes at IRIT in Toulouse, Anders Mörtberg also worked on the formalization of a translation from binding signatures to monads for representing languages with binders in UniMath [21]. This work uses the new possibilities for representing category theory in type theory that univalence provides.

5.6. Formal study of double-word arithmetic algorithms

Participants: Laurence Rideau, Jean-Michel Muller [CNRS and ENS Lyon], Valentina Popescu [CNRS and ENS Lyon].

As part of the ANR Fastrelax project, we have started to formalize double-word arithmetic algorithms, in particular the sum of a double-word and a floating point number and the sum of two double-word numbers described in the article " Tight and rigourous error bounds for basic building blocks of double-word arithmetic" [26].

5.7. Formal foundations of 3D geometry for robot manipulators

Participants: Cyril Cohen, Reynald Affeldt [AIST, Japan].

We formalized the 3D geometry concepts used in the description of kinematics chains, in particular: rotations, rigid body tranformations, screw motions, frame changes, and the Denavit-Hartenberg Convention. This lead to a publication to appear in the internaltional conference CPP 2017 [7].

5.8. Finites sets, finite maps, multisets, order types

Participant: Cyril Cohen.

We extend the Mathematical Components library with a module concerning finite sets (in potentially infinite types), finite maps and multisets. This module plays a crucial role in the formalization of nominal sets, multinomials, semi-algebraic sets, and many experimental developments.

We also extend the Mathematical Components library with a module concerning orders, lattices, and sets. This serves as an abstraction on various libraries, including the finite set library, semi-algebraic sets, finite reunions of intervals, and boolean predicates (in classical theories).

5.9. CoqEAL and modular large scale reflection

Participants: Cyril Cohen, Damien Rouhling.

Extending work by Guillaume Cano, Cyril Cohen, Maxime Dénès, Anders Mörtberg and Vincent Silès, we reimplemented the foundations of the CoqEAL library on Keller and Lasson's parametricity plug-in and provided a more robust translation mechanism. We illustrated the use of this enhanced version of CoqEAL on a new version of the traditional ring tactic. This lead to a publication at JFLA 2017 (Journées Francophones des Langages Applicatifs, the article actually is in English) [17].

5.10. Formalization of semi-algebraic sets

Participants: Yves Bertot, Cyril Cohen, Boris Djalal.

We developed the necessary results about first-order logical formulae to be able to define semi-algebraic sets and semi-algebraic functions in Coq. This required that we provide elements of language to describe quantification over blocks of variables. We show that the equality of semi-algebraic sets is decidable, thanks to the already formalized decision procedure based on quantifier elimination. We then show that our formalized semi-algebraic sets do satisfy general abstract interfaces for sets, as seen in section 5.8

In the long run this work will be instrumental to describe the output of cylindrical algebraic decomposition algorithms. Indeed, this output is usually made of semi-algebraic sets.

5.11. Formalizing the Spectral Theorem

Participant: Cyril Cohen.

We formalize the spectral theorem for normal, hermitian and unitary matrices (this work in progress is available at https://github.com/Barbichu/spectral) These results are useful in the study of rotations and rigid body transformations in dimension 3. This is a key ingredient of the singular value decomposition (useful in inverse kinematics, signal processing, and many other practical applications).

5.12. A formal proof of La Salle's invariance principle

Participants: Yves Bertot, Cyril Cohen, Damien Rouhling.

We started formalizing the proof of La Salle's invariance principle using the Coquelicot library, with the goal of using it to formalize the proof of stability of a control function for the inverted pendulum (a basic exercise that can serve as an introduction to problems in robotics). For now, I have proven a few properties of the set of limit points of a function.

5.13. Formalizing Delaunay triangulations

Participants: Yves Bertot, Wassim Haffaf.

We studied the applicability of the mathematical component library to describe Delaunay triangulation algorithms in the most abstract way. We also formalized a theorem on convex functions known as *Jensen's inequality*.

5.14. Formalizing Quantum Computing

Participant: Laurent Théry.

We have formalized an algorithm proposed by Peter Selinger to synthesize quantum gates. His approach mixes number theoretical notions and linear algebra, two aspects that are well covered by the Mathematical Components Library.

5.15. Formalizing De Bruijn Sequences

Participant: Laurent Théry.

De Bruijn sequences are combinatorial objects. We have shown how they can be generated by exhibiting a link with irreducible polynomials in finite fields, with a formal proof in Coq.

5.16. Formalizing Hanoi towers

Participant: Laurent Théry.

The problem of Hanoi towers is a standard example to explain recursion. While trying to write a formalization, we discovered that there exists an interesting generalisation. Starting with two arbitrary valid positions, the problem is to find an optimal solution to go from one to the other. The solution is somewhat counter-intuitive, and not always unique. We formalized it in Coq.

5.17. Implementation of Bourbaki's Theory of Sets in Coq

Participant: José Grimm.

A paper describing our implementation of the sets of natural numbers, of rational numbers and of real numbers has been published by the Journal of Formalized Reasoning [6].

We implemented Chapter 3, Section 7 (Inverse Limits and Direct Limits) and the start of Chapter 4 (Structures) of the Theory of Sets of Bourbaki, details are found in the Research Report [19]

5.18. Factorization of ordinal numbers

Participant: José Grimm.

Ordinal numbers have been designed at approximately the same time that the foundations of mathematics were being revisited, in the beginning of the 20th century. These objects cross the boundaries of set theory and pose especially difficult challenges when considering the task of formalizing mathematics. This is the reason why we concentrate on formal proofs concerning these objects.

An ordinal number x is said to be prime if $x > 1$ and for every factorisation $x = ab$, one of a or b is equal to x (the other factor is not necessarily equal to 1). Prime ordinals are of three kinds; a power of a power of ω, the successor of a power of ω, or a prime natural number. Every ordinal can uniquely be written as a product of primes, with the following restriction: if a is followed by b in the factor list then: if b is of the first kind, so is a and $a \geq b$, if a and b are natural numbers, then $a \leq b$. The proof can be found in an updated version of [20]

5.19. New logics for differential privacy

Participants: Benjamin Grégoire, Gilles Barthe [IMDEA], Noémie Fong [ENS], Marco Gaboardi [University at Buffalo], Justin Hsu [University of Pennsylvania], Pierre-Yves Strub [IMDEA].

We proposed new logics to work on examples from the differential privacy literature, a hoare logic based on the union bound [10] and a logic based on the deep connection between differential privacy and probabilistic couplings [11], [9].

5.20. Formalizing counter-measures for differential power analysis

Participants: Benjamin Grégoire, Gilles Barthe [IMDEA], Sonia Belaïd [Thales Communications & Security], François Dupressoir [IMDEA], Sebastian Faust [Ruhr Universität Bochum], Pierre-Alain Fouque [Université de Rennes and Institut Universitaire de France], François-Xavier Standaert [Université Catholique de Louvain], Pierre-Yves Strub [IMDEA], Rébecca Zucchini [ENS Cachan and Inria].

Differential power analysis (DPA) is a side-channel attack in which an adversary retrieves cryptographic material by measuring and analyzing the power consumption of the device on which the cryptographic algorithm under attack executes. We introduced new notions and models allowing to check the correctness of counter measures (known as *masking schemes*) [8], [22]. Based on this idea we have developed a compiler to transform an unmasked program into it masked version.

MEXICO Project-Team

7. New Results

7.1. Analyzing Timed Systems Using Tree Automata

Timed systems, such as timed automata, are usually analyzed using their operational semantics on timed words. The classical region abstraction for timed automata reduces them to (untimed) finite state automata with the same time-abstract properties, such as state reachability. In [10], we propose a new technique to analyze such timed systems using finite tree automata instead of finite word automata. The main idea is to consider timed behaviors as graphs with matching edges capturing timing constraints. Such graphs can be interpreted in trees opening the way to tree automata based techniques which are more powerful than analysis based on word automata. The technique is quite general and applies to many timed systems. In this paper, as an example, we develop the technique on timed pushdown systems, which have recently received considerable attention. Further, we also demonstrate how we can use it on timed automata and timed multi-stack pushdown systems (with boundedness restrictions).

7.2. Interrupt Timed Automata with Auxiliary Clocks and Parameters

Interrupt Timed Automata (ITA) are an expressive timed model, introduced to take into account interruptions according to levels. Due to this feature, this formalism is incomparable with Timed Automata. However several decidability results related to reachability and model checking have been obtained. In , we add auxiliary clocks to ITA, thereby extending its expressive power while preserving decidability of reachability. Moreover, we define a parametrized version of ITA, with polynomials of parameters appearing in guards and updates. While parametric reasoning is particularly relevant for timed models, it very often leads to undecidability results. We prove that various reachability problems, including robust reachability, are decidable for this model, and we give complexity upper bounds for a fixed or variable number of clocks, levels and parameters.

7.3. One-Counter Automata with Counter Observability

In a one-counter automaton (OCA), one can produce a letter from some finite alphabet, increment and decrement the counter by one, or compare it with constants up to some threshold. It is well-known that universality and language inclusion for OCAs are undecidable. In [14], we consider OCAs with counter observability: Whenever the automaton produces a letter, it outputs the current counter value along with it. Hence, its language is now a set of words over an infinite alphabet. We show that universality and inclusion for that model are PSPACE-complete, thus no harder than the corresponding problems for finite automata. In fact, by establishing a link with visibly one-counter automata, we show that OCAs with counter observability are effectively determinizable and closed under all boolean operations.

7.4. Diagnosis in Infinite-State Probabilistic Systems

In a recent work, we introduced four variants of diagnosability (FA, IA, FF, IF) in (finite) probabilistic systems (pLTS) depending whether one considers (1) finite or infinite runs and (2) faulty or all runs. We studied their relationship and established that the corresponding decision problems are PSPACE-complete. A key ingredient of the decision procedures was a characterisation of diagnosability by the fact that a random run almost surely lies in an open set whose specification only depends on the qualitative behaviour of the pLTS. In [12], we investigate similar issues for infinite pLTS. We first show that this characterisation still holds for FF-diagnosability but with a Gδ set instead of an open set and also for IF-and IA-diagnosability when pLTS are finitely branching. We also prove that surprisingly FA-diagnosability cannot be characterised in this way even in the finitely branching case. Then we apply our characterisations for a partially observable probabilistic extension of visibly pushdown automata (POpVPA), yielding EXPSPACE procedures for solving diagnosability problems. In addition, we establish some computational lower bounds and show that slight extensions of POpVPA lead to undecidability.

7.5. Accurate Approximate Diagnosability of Stochastic Systems

Diagnosis of partially observable stochastic systems prone to faults was introduced in the late nineties. Diagnosability, i.e. the existence of a diagnoser, may be specified in different ways: (1) exact diagnosability (called A-diagnosability) requires that almost surely a fault is detected and that no fault is erroneously claimed while (2) approximate diagnosability (called ε-diagnosability) allows a small probability of error when claiming a fault and (3) accurate approximate diagnosability (called AA-diagnosability) requires that this error threshold may be chosen arbitrarily small. In [11], we mainly focus on approximate diagnoses. We first refine the almost sure requirement about finite delay introducing a uniform version and showing that while it does not discriminate between the two versions of exact diagnosability this is no more the case in approximate diagnosis. Then we establish a complete picture for the decidability status of the diagnosability problems: (uniform) ε-diagnosability and uniform AA-diagnosability are undecidable while AA-diagnosability is decidable in PTIME, answering a longstanding open question.

7.6. Diagnosability of Repairable Faults

The diagnosis problem for discrete event systems consists in deciding whether some fault event occurred or not in the system, given partial observations on the run of that system. Diagnosability checks whether a correct diagnosis can be issued in bounded time after a fault, for all faulty runs of that system. This problem appeared two decades ago and numerous facets of it have been explored, mostly for permanent faults. It is known for example that diagnosability of a system can be checked in polynomial time, while the construction of a diagnoser is exponential. In [21], we examine the case of transient faults, that can appear and be repaired. Diagnosability in this setting means that the occurrence of a fault should always be detected in bounded time, but also before the fault is repaired. Checking this notion of diagnosability is proved to be PSPACE-complete. It is also shown that faults can be reliably counted provided the system is diagnosable for faults and for repairs.

7.7. Optimal constructions for active diagnosis

The task of diagnosis consists in detecting, without ambiguity, occurrence of faults in a partially observed system. Depending on the degree of observability, a discrete event system may be diagnosable or not. Active diagnosis aims at controlling the system in order to make it diagnosable. Solutions have already been proposed for the active diagnosis problem, but their complexity remains to be improved. In [8], we solve the active diagnosability decision problem and the active diagnoser synthesis problem, proving that (1) our procedures are optimal w.r.t. to computational complexity, and (2) the memory required for the active diagnoser produced by the synthesis is minimal. Furthermore, focusing on the minimal delay before detection, we establish that the memory required for any active diagnoser achieving this delay may be highly greater than the previous one. So we refine our construction to build with the same complexity and memory requirement an active diagnoser that realizes a delay bounded by twice the minimal delay.

7.8. Verification of parameterized communicating automata via split-width

In [16] study verification problems for distributed systems communicating via unbounded FIFO channels. The number of processes of the system as well as the communication topology are not fixed a priori. Systems are given by parameterized communicating automata (PCAs) which can be run on any communication topology of bounded degree, with arbitrarily many processes. Such systems are Turing powerful so we concentrate on under-approximate verification. We extend the notion of split-width to behaviors of PCAs. We show that emptiness, reachability and model-checking problems of PCAs are decidable when restricted to behaviors of bounded split-width. Reachability and emptiness are EXPTIME-complete, but only polynomial in the size of the PCA. We also describe several concrete classes of bounded split-width, for which we prove similar results.

7.9. Cyclic Ordering through Partial Orders

The orientation problem for ternary cyclic order relations has been attacked in the literature from combinatorial perspectives, through rotations, and by connection with Petri nets. In [7], we propose a two-fold characterization of orientable cyclic orders in terms of symmetries of partial orders as well as in terms of separating sets (cuts). The results are inspired by properties of non-sequential discrete processeses, but also apply to dense structures of any cardinality.

7.10. Predicting Traffic Load in Public Transportation Networks

This work is part of an ongoing effort to understand the dynamics of passenger loads in modern, multimodal transportation networks (TNs) and to mitigate the impact of perturbations, under the restrictions that the precise number of passengers in some point of the TN that intend to reach a certain destination (i.e. their distribution over different trip profiles) is unknown. In [29], we introduce an approach based on a stochastic hybrid automaton model for a TN that allows to compute how such probabilistic load vectors are propagated through the TN. In [23], [30], develop a computation strategy for forecasting the network's load a certain time in the future.

In [22], [28], we continue our work on perturbation analysis of multimodal transportation networks (TNs) by means of a stochastic hybrid automaton (SHA) model. We focus here on the approximate computation , in particular on the major bottleneck consisting in the high dimensionality of systems of stochastic differential balance equations (SDEs) that define the continuous passenger-flow dynamics in the different modes of the SHA model. In fact, for every pair of a mode and a station, one system of coupled SDEs relates the passenger loads of all discrete points such as platforms considered in this station, and all vehicles docked to it, to the passenger flows in between. In general, such an SDE system has many dimensions, which makes its numerical computation and thus the approximate computation of the SHA model intractable. We show how these systems can be canonically replaced by lower-dimensional ones, by decoupling the passenger flows inside every mode from one another. We prove that the resulting approximating passenger-flow dynamics converges to the original one, if the replacing set of balance equations set up for all decoupled passenger flows communicate their results among each other in vanishing time intervals.

For more information about the whole project, see [27].

7.11. Unfolding of Parametric Logical Regulatory Networks

In systems biology, models of cellular regulatory processes such as gene regulatory networks or signalling pathways are crucial to understanding the behaviour of living cells. Available biological data are however often insufficient for full model specification. In [18], we focus on partially specified models where the missing information is abstracted in the form of parameters. We introduce a novel approach to analysis of parametric logical regulatory networks addressing both sources of combinatoric explosion native to the model. First, we introduce a new compact representation of admissible parameters using Boolean lattices. Then, we define the unfolding of parametric regulatory networks. The resulting structure provides a partial- order reduction of concurrent transitions, and factorises the common transitions among the concrete models. A comparison is performed against state-of-the-art approaches to parametric model analysis.

7.12. Relationship between the Reprogramming Determinants of Boolean Networks and their Interaction Graph

In [24], we address the formal characterization of tar- gets triggering cellular trans-differentiation in the scope of Boolean net- works with asynchronous dynamics. Given two fixed points of a Boolean network, we are interested in all the combinations of mutations which allow to switch from one fixed point to the other, either possibly, or in- evitably. In the case of existential reachability, we prove that the set of nodes to (permanently) flip are only and necessarily in certain connected components of the interaction graph. In the case of inevitable reachabil- ity, we provide an algorithm to identify a subset of possible solutions.

7.13. D-SPACES: An Implementation of Declarative Semantics for Spatially Structured Information

We introduce in [17] D-SPACES, an implementation of constraint systems with space and extrusion operators. Constraint systems are algebraic models that allow for a semantic language-like representation of information in systems where the concept of space is a primary structural feature. We give this information mainly an epistemic interpretation and consider various agents as entities acting upon it. D-SPACES is coded as a c++11 library providing implementations for constraint systems, space functions and extrusion functions. The interfaces to access each implementation are minimal and thoroughly documented. D-SPACES also provides property-checking methods as well as an implementation of a specific type of constraint systems (a boolean algebra). This last implementation serves as an entry point for quick access and proof of concept when using these models. Furthermore, we offer an illustrative example in the form of a small social network where users post their beliefs and utter their opinions.

7.14. Belief, Knowledge, Lies and Other Utterances in an Algebra for Space and Extrusion

The notion of constraint system (cs) is central to declarative formalisms from concurrency theory such as process calculi for concurrent constraint programming (ccp). Constraint systems are often represented as lattices: their elements, called constraints, represent partial information and their order corresponds to entailment. Recently a notion of n-agent spatial cs was introduced to represent information in concurrent constraint programs for spatially distributed multi-agent systems. From a computational point of view a spatial constraint system can be used to specify partial information holding in a given agent's space (local information). From an epistemic point of view a spatial cs can be used to specify information that a given agent considers true (beliefs). Spatial constraint systems, however, do not provide a mechanism for specifying the mobility of information/processes from one space to another. Information mobility is a fundamental aspect of concurrent systems. In [6] we develop the theory of spatial constraint systems with operators to specify information and processes moving from a space to another. We shall investigate the properties of this new family of constraint systems and illustrate their applications. From a computational point of view the new operators provide for process/information extrusion, a central concept in formalisms for mobile communication. From an epistemic point of view extrusion corresponds I to a notion we shall call utterance; a piece of information that an agent communicate to others but that may be inconsistent with the agent's beliefs. Utterances can then be used to express instances of epistemic notions such as hoaxes or intentional lies which are common place in social media. Spatial constraint system can express the epistemic notion of belief by means of space functions that specify local information. We shall also show that spatial constraint can also express the epistemic notion of knowledge by means of a derived spatial operator that specifies global information.

7.15. Goal-Driven Unfolding of Petri Nets

Unfoldings provide an efficient way to avoid the state-space explosion due to interleavings of concurrent transitions when exploring the runs of a Petri net. The theory of adequate orders allows one to define finite prefixes of unfoldings which contain all the reachable markings. In this paper we are interested in reachability of a single given marking, called the goal. In [26], We propose an algorithm for computing a finite prefix of the unfolding of a 1-safe Petri net that preserves all minimal configurations reaching this goal. Our algorithm combines the unfolding technique with on-the-fly model reduction by static analysis aiming at avoiding the exploration of branches which are not needed for reaching the goal. We present some experimental results.

PARSIFAL Project-Team

7. New Results

7.1. Linear rewriting systems for Boolean logic

Participant: Lutz Straßburger.

Last year's result on the nonexistence of a complete linear term rewriting system for propositional logic [53] has been generalized and some applications to proof theory have been invesigated. For example, we have found that the medial rule which plays a central role in deep inference systems is canonical in a strong sense: It is minimal, and every rule that reduce contraction to an atomic form is indeed derivable via medial. This is published in [15] (joint work with Anupam Das).

7.2. Non-crossing Tree Realizations of Ordered Degree Sequences

Participant: Lutz Straßburger.

We investigate the enumeration of non-crossing tree realizations of integer sequences, and we consider a special case in four parameters, that can be seen as a four-dimensional tetrahedron that generalizes Pascal's triangle and the Catalan numbers. This work is motivated by the study of ambiguities arising in the parsing of natural language sentences using categorial grammars. This is joint work with Laurent Méhats and published in [31].

7.3. Focusing for Nested Sequents

Participants: Kaustuv Chaudhuri, Sonia Marin, Lutz Straßburger.

Focusing is a general technique for transforming a sequent proof system into one with a syntactic separation of non-deterministic choices without sacrificing completeness. This not only improves proof search, but also has the representational benefit of distilling sequent proofs into synthetic normal forms. We have shown how to apply the focusing technique to nested sequent calculi, a generalization of ordinary sequent calculi to tree-like instead of list-like structures. We thus improve the reach of focusing to the most commonly studied modal logics, the logics of the modal S5 cube. Among our key contributions is a focused cut-elimination theorem for focused nested sequents. This is published in [25].

Then we further extend our results to intuitionistic nested sequents, which can capture all the logics of the intuitionistic S5 cube in a modular fashion. We obtained an internal cut-elimination procedure for the focused system which in turn is used to show its completeness. This is published in [26]

7.4. Combining inference systems: a generalization of Nelson-Oppen and MCSAT

Participant: Stéphane Graham-Lengrand.

Nelson-Oppen [79] and Model-Constructing Satisfiability (MCSAT) [89], [65] are two methodologies that allow the reasoning mechanisms of different theories to collaborate, in order to tackle hybrid problems. While these methodologies are often used and implemented for the practical applications of Automated Reasoning, their rather sophisticated foundations are traditionally explained in terms of model theory. SRI International pioneered some work providing such methodologies with new and more general foundations in terms of *inference systems* [57], closer to proof theory and to Parsifal's research. The more recent MCSAT methodology was not captured, more generally lacked any kind of theorem about the generic combination of arbitrary theories, and was also thought to be incompatible with the Nelson-Oppen approach, so that SMT-solvers are either working with one methodology or the other, unable to get the best of both worlds.

In 2016 we designed a combination methodology, based on *inference systems*, that supersedes both Nelson-Oppen and MCSAT [34]. We showed its soundness and completeness, and identified for this the properties that the theories to combine are required to satisfy. This generalized MCSAT with the generic combination mechanism that it lacked, and showed that it is perfectly compatible with the Nelson-Oppen methodology, which can now cohabit within the same solver.

7.5. Linear lambda terms as invariants of rooted trivalent maps

Participant: Noam Zeilberger.

Recent studies of the combinatorics of linear lambda calculus have uncovered some unexpected connections to the old and well-developed theory of graphs embedded on surfaces (also known as "maps") [47], [87], [88]. In [19], we aimed to give a simple and conceptual account for one of these connections, namely the correspondence (originally described by Bodini, Gardy, and Jacquot [47]) between α-equivalence classes of closed linear lambda terms and isomorphism classes of rooted trivalent maps on compact oriented surfaces without boundary. One immediate application of this new account was a characterization of trivalent maps which are *bridgeless* (in the graph-theoretic sense of having no disconnecting edge) as linear lambda terms with no closed proper subterms. In turn, this lead to a surprising but natural reformulation of the Four Color Theorem as a statement about typing in lambda calculus.

7.6. A bifibrational reconstruction of Lawvere's presheaf hyperdoctrine

Participant: Noam Zeilberger.

In joint work with Paul-André Melliès, we have been investigating the categorical semantics of type refinement systems, which are type systems built "on top of" a typed programming language to specify and verify more precise properties of programs. The fibrational view of type refinement we have been developing (cf. [72]) is closely related to the categorical perspective on first-order logic introduced by Lawvere [66], but with some important conceptual and technical differences that provide an opportunity for reflection. For example, Lawvere's axiomatization of first-order logic (his theory of so-called "hyperdoctrines") was based on the idea that existential and universal quantification can be described respectively as left and right adjoints to the operation of substitution, this giving rise to a family of *adjoint triples* $\Sigma_f \dashv \mathcal{P}_f \dashv \Pi_f$ (one such triple for every function $f : A \to B$). On the other hand, a bifibration only induces a family of *adjoint pairs* $\mathsf{push}_f \dashv \mathsf{pull}_f$ (again, one such pair for every $f : A \to B$). In [33], we resolved this and other apparent mismatches by applying ideas inspired by the semantics of linear logic and the shift from the cartesian closed category $\mathbf{Set}$ to the symmetric monoidal closed category $\mathbf{Rel}$. Two other applications of our analysis include an axiomatic treatment of *directed* equality predicates (which can be modelled as "hom" presheaves, realizing an early vision of Lawvere), as well as a simple calculus of string diagrams that is highly reminiscent of C. S. Peirce's "existential graphs" for predicate logic.

7.7. Towards a link between CPS and focusing

Participant: Matthias Puech.

Continuation-passing style translations make a functional program more explicit by sequentializing its computations and reifying its control. They have been used as an intermediate language in many compilers. They are also understood as classical-to-intuitionistic proof embedding (so-called double negation translations). Matthias Puech studied a novel correspondence between CPS and focusing: to each CPS transform corresponds a focused proof system that is identifiable as a particular polarization of classical statements. Since, after Miller's and others work, we know the full design space of focused sequent calculi, we expect to understand the full design space of CPS translation.

The first step of this goal is to study the syntax and typing of variants of the CPS translation. Puech designed and implemented in OCaml a compacting, optimizing CPS translation, while using OCaml's type system to verify that it maps well-typed terms to well-typed terms in a tightly restricted syntactical form (the "typeful" approach to formalization) [82]. The resulting type system is in Curry-Howard isomorphism with a weakly focused proof system: LJQ.

7.8. Proof Checking and Logic Programming

Participants: Roberto Blanco, Tomer Libal, Dale Miller, Marco Volpe.

In a world where trusting software systems is increasingly important, formal methods and formal proofs can help provide some basis for trust. Proof checking can help to reduce the size of the *trusted base* since we do not need to trust an entire theorem prover: instead, we only need to trust a (smaller and simpler) proof checker. Many approaches to building proof checkers require embedding within them a full programming language. In most modern proof checkers and theorem provers, that programming language is a functional programming language, often a variant of ML. In fact, aspects of ML (e.g., strong typing, abstract data types, and higher-order programming) were designed to make ML a trustworthy "meta-language" for checking proofs. While there is considerable overlap between logic programming and proof checking (e.g., both benefit from unification, backtracking search, efficient term structures, etc), the discipline of logic programming has, in fact, played a minor role in the history of proof checking. Miller has been pushing the argument that logic programming can have a major role in the future of this important topic [18]. Many aspects of the ProofCert project are based on this perspective that logic programming techniques and methods can have significant utility within proof checking. This perspective stands in constrast to the work on the Dedukti proof checking framework [44] where functional programming principles are employed for proof checking.

7.9. Proof Certificates for First-Order Equational Logic

Participants: Dale Miller, Zakaria Chihani.

The kinds of inference rules and decision procedures that one writes for proofs involving equality and rewriting are rather different from proofs that one might write in first-order logic using, say, sequent calculus or natural deduction. For example, equational logic proofs are often chains of replacements or applications of oriented rewriting and normal forms. In contrast, proofs involving logical connectives are trees of introduction and elimination rules. Chihani and Miller have shown [13] how it is possible to check various equality-based proof systems with a programmable proof checker (the *kernel* checker) for first-order logic. That proof checker's design is based on the implementation of *focused proof search* and on making calls to (user-supplied) *clerks and experts* predicates that are tied to the two phases found in focused proofs. This particular design is based on the work of Chihani, Miller, and Renaud [14].

The specification of these clerks and experts provide a formal definition of the structure of proof evidence and they work just as well in the equational setting as in the logic setting where this scheme for proof checking was originally developed. Additionally, executing such a formal definition on top of a kernel provides an actual proof checker that can also do a degree of proof reconstruction. A number of rewriting based proofs have been defined and checked in this manner.

7.10. Extended Pattern Unification

Participants: Tomer Libal, Dale Miller.

Unification is a central operation in the construction of a range of computational logic systems based on first-order and higher-order logics. First-order unification has a number of properties that dominates the way it is incorporated within such systems. In particular, first-order unification is decidable, unary, and can be performed on untyped term structures. None of these three properties hold for full higher-order unification: unification is undecidable, unifiers can be incomparable, and term-level typing can dominate the search for unifiers. The so-called *pattern* subset of higher-order unification was designed to be a small extension to first-order unification that respected the basic laws governing λ-binding (the equalities of α, β, and η-conversion) but which also satisfied those three properties. While the pattern fragment of higher-order unification has been popular in various implemented systems and in various theoretical considerations, it is too weak for a number of applications. Libal and Miller [28] have defined an extension of pattern unification that is motivated by some existing applications and which satisfies these three properties. The main idea behind their extension is that the arguments to a higher-order, free variable can be more than just distinct bound variables: they can also be terms constructed from (sufficient numbers of) such variables using term constructors and where no

argument is a subterm of any other argument. This extension to pattern unification satisfies the three properties mentioned above. R. Blanco is currently adding this extended unification to the Abella theorem prover.

7.11. Focused proofs for modal logics

Participants: Tomer Libal, Sonia Marin, Dale Miller, Marco Volpe.

Several deductive formalisms (e.g., sequent, nested sequent, labeled sequent, hypersequent calculi) have been used in the literature for the treatment of modal logics, and some connections between these formalisms are already known. Marin, Miller, and Volpe [30] have propose a general framework, which is based on a focused version of the labeled sequent calculus by Negri [78], augmented with some parametric devices allowing to restrict the set of proofs. By properly defining such restrictions and by choosing an appropriate polarization of formulas, one can obtain different, concrete proof systems for the modal logic K and for its extensions by means of geometric axioms. The expressiveness of the labeled approach and the control mechanisms of focusing allow a clean emulation of a range of existing formalisms and proof systems for modal logic. These results make it possible to write Foundational Proof Certificate definitions of common modal logic proof systems.

7.12. Preserving differential privacy under finite-precision semantics

Participant: Dale Miller.

(Joint work with Ivan Gazeau and Catuscia Palamidessi). The approximation introduced by finite-precision representation of continuous data can induce arbitrarily large information leaks even when the computation using exact semantics is secure. Such leakage can thus undermine design efforts aimed at protecting sensitive information. Gazeau, Miller, and Palamidessi [16] have applied differential privacy—an approach to privacy that emerged from the area of statistical databases—to this problem. In their approach, privacy is protected by the addition of noise to a true (private) value. To date, this approach to privacy has been proved correct only in the ideal case in which computations are made using an idealized, infinite-precision semantics. An analysis of implementation levels, where the semantics is necessarily finite-precision, i.e. the representation of real numbers and the operations on them are rounded according to some level of precision. In general there are violations of the differential privacy property but a limited (but, arguably, totally acceptable) variant of the property can be used insteand, under only a minor degradation of the privacy level. In fact, two cases of noise-generating distributions can be employed: the standard Laplacian mechanism commonly used in differential privacy, and a bivariate version of the Laplacian recently introduced in the setting of privacy-aware geolocation.

7.13. Certification of Prefixed Tableau Proofs for Modal Logic

Participants: Tomer Libal, Marco Volpe.

This work [29] describes the theory and implementation of a proof checker for tableau theorem provers for modal logics. The tool supports proofs in both the traditional tableau format as well as the free variable variant. The implentation can be found at https://github.com/proofcert/checkers under the gandalf2016 branch.

7.14. Towards a Substitution Tree Based Index for Higher-order Resolution Theorem Provers

Participant: Tomer Libal.

First-order resolution theorem provers depend on efficient data structures for redundancy elimination. These data structures do not exist for higher-order resolution theorem provers. In [32] we discuss a new approach to this problem. (Joint work with Alexander Steen).

7.15. Open Call-by-Value

Participant: Beniamino Accattoli.

Functional programming languages are often based on the call-by-value λ-calculus, whose elegant theory relies on weak evaluation and closed terms, that are natural hypotheses in the study of programming languages. To model proof assistants, however, strong evaluation and open terms are required, and it is well known that the operational semantics of call-by-value becomes problematic in this case. In this joint work with Giulio Guerrieri we studied the intermediate setting—that we call Open Call-by-Value—of weak evaluation with open terms, on top of which Gregoire and Leroy designed the abstract machine of Coq. Various calculi for Open Call-by-Value already exist, each one with its pros and cons. We did a detailed comparative study of the operational semantics of four of them, coming from different areas such as the study of abstract machines, denotational semantics, linear logic proof nets, and sequent calculus. We showed that these calculi are all equivalent from a termination point of view, justifying the slogan Open Call-by-Value. The work has been published in the proceedings of the international conference APLAS 2016 [22].

7.16. A Reasonable Abstract Machine for the Strong λ-Calculus

Participant: Beniamino Accattoli.

We provided a new proof that the strong λ-calculus is a reasonable computational model. The original proof is by B. Accattoli and H. Dal Lago uses a calculus with explicit substitutions while the new one relies on a new sophisticated abstract machine, the Useful MAM. The work has been published in the proceeding of the international conference WoLLIC 2016 [21].

7.17. Space-efficient Acyclicity Constraints

Participant: Taus Brock-Nannestad.

Acyclicity constraints can be used to encode a large variety of useful constraints on graphs. The basic constraint itself can be encoded in terms of simpler constraints (e.g. integer linear constraints) in a straightforward and intuitive way, associating to each vertex of the (fixed) input graph a variable with domain linear in the size of the graph. For large graphs, this quickly becomes inefficient.

In [24], we show that in the case of planar graphs, a more efficient encoding (using a two-valued variable per vertex) is possible.

7.18. Exp-log normal form of types and the axioms for η-equality of the λ-calculus with sums

Participant: Danko Ilik.

In the presence of sum types, the λ-calculus has but one implemented (and incomplete) heuristic for deciding $\beta\eta$-equality of terms, in spite of a dozen of meta-theoretic works showing that the equality is decidable.

In the work discussed here, we first used the exp-log decomposition of the arrow type—inspired from the analytic transformation $a^b = \exp(b \times \log a)$—to obtain a type normal form for the type languages $\{\rightarrow, \times, +\}$. We then made a quotient of the $\beta\eta$-equality of terms modulo the terms coerced into their representation at the exp-log normal form of their type. This allows to obtain a *simplification* of the so far standard axioms for $\beta\eta$-equality.

Moreover, we provided a Coq implementation of a heuristic decision procedure for this equality. Although a heuristic, this implementation manages to tackle examples of equal terms that need a complex program analysis in the only previously implemented heuristic of Vincent Balat.

This work is described in a paper accepted for presentation at POPL 2017, [27].

7.19. Invertible-rule-free sequent calculi and an intuitionistic arithmetical hierarchy

Participants: Taus Brock-Nannestad, Danko Ilik.

In sequent calculi, proof rules can be divided into two groups: invertible (asynchronous) proof rules and non-invertible (synchronous) proof rules. Even in focusing sequent calculi the two groups of rules are present, albeit grouped together in synthetic rules (we speak of the synchronous and asynchronous phase).

In this work, we used the exp-log decomposition (described above) in the context of logic in order to obtain a version of sequent caclulus which contains synchronous rules only, a first such formalism for intuitionistic logic.

We extended the picture from the setting of propositional to the one of first-order intuitionistic logic, where the exp-log decomposition provided us with an intuitionistic hierarchy of formulas analogous to the classical arithmetical hierachy; although the classical arithmetical hierarchy exists since the 1920s, a correspondingly versatile notion for intuitionistic logic has been elusive up to this day.

This work is described in the manuscript [37], submitted to an academic journal.

PI.R2 Project-Team

5. New Results

5.1. Effects in proof theory and programming

Participants: Hugo Herbelin, Gabriel Lewertowski, Étienne Miquey, Alexis Saurin, Matthieu Sozeau.

5.1.1. A classical sequent calculus with dependent types

Dependent types are a key feature of type systems, typically used in the context of both richly-typed programming languages and proof assistants. Control operators, which are connected with classical logic along the proof-as-program correspondence, are known to misbehave in the presence of dependent types [11], unless dependencies are restricted to values. As a step in his work to develop a sequent-calculus version of Hugo Herbelin's dPA_ω system [13], Étienne Miquey proposed a sequent calculus with classical logic and dependent types. His calculus—named dL—is an extension of the $\mu\tilde{\mu}$-calculus with a syntactical restriction of dependent types to the fragment of *negative-elimination free* proofs. The corresponding type system includes a list of explicit dependencies, which maintains type safety. He showed that a continuation-passing style translation can be derived by adding delimited continuations, and how a chain of dependencies can be related to a manipulation of the return type of this continuations. This work has been accepted for publication at ESOP 2017 [39].

5.1.2. Logical foundations of call-by-need evaluation

Alexis Saurin, in collaboration with Pierre-Marie Pédrot, extended their reconstruction of call-by-need based on linear head reduction with control. They showed how linear head reduction could be adapted to the $\lambda\mu$-calculus. This classical linear head reduction lifts the usual properties of the intuitionistic one (with respect to σ-equivalence) to the $\lambda\mu$-calculus (and its σ-equivalence already formulated by Olivier Laurent in his PhD thesis). Moreover, they showed that substitution sequences of the $\lambda\mu$-calculus' linear head reduction are in correspondence with the classical Krivine abstract machine substitution sequences, validating the known fact that the KAM implements linear head reduction. This work has been published at ESOP'16 [29]. They plan to lift to the $\lambda\mu$-calculus their three-step transformation from linear head reduction to call-by-need, and to study the correspondence with Ariola, Herbelin and Saurin's classical call-by-need.

5.1.3. Call-by-name forcing for Dependent Type Theory

Guilhem Jaber, Gabriel Lewertowski, Pierre-Marie Pédrot, Matthieu Sozeau, and Nicolas Tabareau studied a variant of the forcing translation for dependent type theory, moving from the call-by-value variant to a call-by-name version which naturally preserves definitional equalities, avoiding the coherence pitfalls of the former one. This new version was inspired by Pierre-Marie Pédrot's former decomposition of forcing in call-by-push-value. It allows to show various metatheoretical results in a succint fashion, notably for the independence of axioms. Work is ongoing to produce more positive results including abstracting reasoning on step-indexing using this technique. This work was presented at LICS 2016 [28].

5.1.4. Classical realizability and implicative algebras

Étienne Miquey has been working with Alexandre Miquel in Montevideo on the topic of implicative algebras. Implicative algebras are an algebraization of the structure needed to develop a realizability model. In particular, they give rise to the usual ordered combinatory algebras and thus to the triposes used to model classical realizability. An implicative algebra is given by an implicative structure (which consists of a complete semi-lattice with a binary operation $\rightarrow$) together with a separator containing the element interpreted as true in the structure. Étienne Miquey has been working on a formalization of implicative algebras theory in Coq. Following the work of Guillaume Munch-Maccagnoni on focalization and classical realizability, he also worked on alternative presentations within structures based on other connectives, (negation, "par", tensor),

rather than $\rightarrow$. Such connectives correspond to the decomposition of the arrow according to the strategy of evaluation (call-by-name/call-by-value). The aim of this work is to obtain a classification of the possible algebraic structures to interpret classical realizability, in order to prove that different strategies of evaluation actually provide us with equivalent models.

5.2. Reasoning and programming with infinite data

Participants: Amina Doumane, Yann Régis-Gianas, Alexis Saurin.

This theme is part of the ANR project Rapido (see the National Initiatives section).

5.2.1. Proof theory of infinitary and circular proofs

In collaboration with David Baelde, Amina Doumane and Alexis Saurin developed further the theory of infinite proofs. In their study of the proof theory of circular and infinitary proofs in $\mu MALL$, they established two fundamental proof-theoretical and computational results, namely cut-elimination and focalisation. This result appeared in CSL 2016 (long version in [33]).

The usual result of focalisation for linear logic can actually be extended to circular proofs, but, contrarily to finitary $\mu MALL$ proofs where fixed-points operators can be given an arbitrary polarity, the least fixed-points must be set to be a positive construction and the greatest fixed-points to be negative, which is consistent with intuition from programming with inductive and co-inductive datatypes. An interesting phenomenon arising with focalisation is that some infinite but regular proofs may not have any regular focused proofs. This is similar to what happens for cut-elimination of regular proofs.

The proof of cut-elimination is quite involved and proceeds in two steps relying on semantic arguments, even though the paper actually proves a cut-elimination result and not only a cut-admissibility result as usual semantic arguments provide. A first part of the proof shows that some cut-reduction strategy is actually productive while a second part of the proof shows that the proof-object produced is actually a correct proof in the sense that it satisfies the validity condition of $\mu MALL$ infinite proofs. Previous cut-elimination results were only known for the restricted additive fragment of linear logic with fixed points, a result due to Santocanale and Fortier.

Baelde, Doumane and Saurin are currently working with Jaber to extend the cut-elimination result to a more expressive validity condition for $\mu MALL$ infinite proofs.

5.2.2. Automata theory meets proof theory: proof certificates for Büchi inclusion

In a joint work with David Baelde and Lucca Hirschi, Amina Doumane and Alexis Saurin carried out a proof-theoretical investigation of the linear-time μ-calculus, proposing well-structured proof systems and showing constructively that they are complete for inclusions of Büchi automata suitably encoded as formulas.

They do so in a way that combines the advantages of two lines of previous work: Kaivola gave a proof of completeness for an axiomatisation that amounts to a finitary proof system, but his proof is non-constructive and yields no reasonable procedure. On the other hand, Dax, Hofmann and Lange recently gave a deductive system that is appropriate for algorithmic proof search, but their proofs require a global validity condition and do not have a well understood proof theory.

They work with well-structured proof systems, effectively constructing proofs in a finitary sequent calculus that enjoys local correctness and cut elimination. This involves an intermediate circular proof system in which one can obtain proofs for all inclusions of parity automata, by adapting Safra's construction. In order to finally obtain finite proofs of Büchi inclusions, a translation result from circular to finite proofs is designed.

These results appeared in LICS 2016 (long version in [37]). Since then, Doumane extended the result and obtained a constructive proof of completeness for the full linear-time μ-calculus.

5.2.3. Co-patterns

In collaboration with Paul Laforgue (Master 1, University Paris Diderot), Yann Régis-Gianas studied the mechanisms of co-patterns introduced by Abel and Pientka from a programming language perspective. More precisely, they defined an untyped version of this calculus as well as an abstract machine to efficiently evaluate cofunctions. In addition, they designed several (type preserving) encodings of co-patterns using generalized algebraic datatypes and purely functional objects. Finally, they started to revisit an optimisation called "stream fusion" in a purely equational way by application of copattern-based program definitions.

5.2.4. Functional reactive programming

In collaboration with Sylvain Ribstein (Master 1, University Paris Diderot), Yann Régis-Gianas defined an OCaml library for differential functional reactive programming (DFRP). This framework extends standard functional reactive programming with the possibility to modify past events and to compute the consequences of this modification in all the events that depend on it. A paper is in preparation.

Saurin and Tasson co-advised in the spring/summer of 2016 the master internship of Rémi Nollet who started his PhD thesis under their supervision in September 2016. The topic of his thesis is the extension of Curry-Howard correspondence between FRP and LTL as recently noticed by Jeffrey and Jeltsch. During his internship, Nollet studied various proof systems for LTL and compared them to type systems for FRP. He notably studied various translations between natural deduction and sequent calculus, which led him to study precisely the role played by structural rules in those translations and preparing the work for future extensions to classical constructive LTL, and to work out the foundations for an extension of Curien-Herbelin's system L, closer to abstract machines, for LTL.

5.3. Effective higher-dimensional algebra

Participants: Cyrille Chenavier, Pierre-Louis Curien, Yves Guiraud, Maxime Lucas, Philippe Malbos, Samuel Mimram, Jovana Obradović.

5.3.1. Rewriting and Garside theory

Yves Guiraud has collaborated with Patrick Dehornoy (LNO, Univ. Caen) to develop an axiomatic setting for monoids with a special notion of quadratic normalisation map with good computational properties. This theory generalises the normalisation procedure known for monoids that admit a special family of generators called a Garside family [53] to a much wider class that also includes the plactic monoids. It is proved that good quadratic normalisation maps correspond to quadratic convergent presentations, together with a sufficient condition for this to happen, based on the shape of the normalisation paths on length-three words. This work has been published in the International Journal of Algebra and Computation [21].

Building on this last article, Yves Guiraud currently collaborates with Matthieu Picantin (IRIF, Univ. Paris 7) to generalise the main results of Gaussent, Guiraud and Malbos on coherent presentations of Artin monoids [7], to monoids with a Garside family. This will allow an extension of the field of application of the rewriting methods to other geometrically interesting classes of monoids, such as the dual braid monoids.

Still in collaboration with Matthieu Picantin, Yves Guiraud develops an improvement of the classical Knuth-Bendix completion procedure, called the KGB completion procedure. The original algorithm tries to compute, from an arbitrary terminating rewriting system, a finite convergent presentation by adding relations to solve confluence issues. Unfortunately, this algorithm fails on standard examples, like most Artin monoids with their usual presentations. The KGB procedure uses the theory of Tietze transformations, together with Garside theory, to also add new generators to the presentation, trying to reach the convergent Garside presentation identified in [21]. The KGB completion procedure is partially implemented in the prototype Rewr, developed by Yves Guiraud and Samuel Mimram.

5.3.2. Higher-dimensional linear rewriting

With Eric Hoffbeck (LAGA, Univ. Paris 13), Yves Guiraud and Philippe Malbos have introduced in [65] the setting of linear polygraphs to formalise a theory of linear rewriting, generalising Gröbner bases. They have adapted the method of Guiraud and Malbos [9] to compute polygraphic resolutions of associative algebras, with applications to the decision of the Koszul homological property. They are currently finishing the major overhaul of this work, started in 2015, whose main goal is to ease the adaptation of the results to other algebraic varieties, like commutative algebras or Lie algebras.

Cyrille Chenavier, supervised by Yves Guiraud and Philippe Malbos, explored the use of Berger's theory of reduction operators [45] to improve the theory of Gröbner bases for associative algebras. This work has permitted to unveil two interesting algebraic structures that are hidden in rewriting theory. First, the operations that associate a normal form to an arbitrary word admit a structure of lattice, that gives a new algebraic characterisation of confluence and a new algorithm for completion, based on an iterated use of the meet-operation of the lattice. Second, under mild technical conditions, the different normalisation strategies are related through braid-like relations, as in Artin monoids, that have been used to propose a new method for a particular problem in homological algebra (namely, the construction of a contracting homotopy for the Koszul complex). The second result is published in Algebra and Representation Theory [20], the first one is submitted for publication [35], and both are contained in Cyrille Chenavier's PhD thesis [19].

5.3.3. Rewriting methods for coherence

Yves Guiraud and Philippe Malbos have written a survey on the use of rewriting methods in algebra, centered on a formulation of Squier's homotopical and homological theorems in the modern language of higher-dimensional categories. This article is intended as an introduction to the domain, mainly for graduate students, and will appear in Mathematical Structures in Computer Science [23].

Maxime Lucas, supervised by Yves Guiraud and Pierre-Louis Curien, has applied the rewriting techniques of Guiraud and Malbos [68] to prove coherence theorems for bicategories and pseudofunctors. He obtained a coherence theorem for pseudonatural transformations thanks to a new theoretical result, improving on the former techniques, that relates the properties of rewriting in 1- and 2-categories. This result is published in the Journal of Pure and Applied Algebra [25]. Maxime is currently engaged into a major rework of the results of [9], that will produce improved methods to build Squier's polygraphic resolution from a convergent presentation, based on the use of cubical higher categories instead of globular ones. He has already achieved a first result in this direction [77], and conducted a major foundational work towars the full result [78], which have just been submitted for publication.

Pierre-Louis Curien and Jovana Obradović pursued their work on cyclic operads (started in [36], now acepted in the Journal Applied Categorical Structures). They established the notion of categorified cyclic operad. Categorification involves weakening the axioms of cyclic operads (from equalities to natural isomorphisms) and formulating conditions concerning these isomorphisms which ensure coherence. For entries-only cyclic operads, this coherence is of the same kind as the coherence of symmetric monoidal categories: all diagrams made of associator and commutator isomorphisms are required to commute. However, in the setting of cyclic operads, where the existence of objects and morphisms depends on the shape of a fixed unrooted tree, these arrows do not always exist. In other words, the coherences that Mac Lane established for symmetric monoidal categories do not solve the coherence problem of categorified cyclic operads. They exhibited the appropriate conditions of this setting and proved the coherence theorem, relying on a result of Došen and Petrić, coming from the coherence of categorified operads. Additionally, by the equivalence between the two possible characterisations of cyclic operads, for cyclic operads introduced as operads with extra structure (that exchanges the output of an operation with one of its inputs), i.e. for exchangeable-output cyclic operads, they examined which of the axioms of the extra structure needs to be weakened (in order to lift that equivalence to weakened structures), and they exhibited the appropriate coherence conditions in this setting as well.

5.4. Incrementality

Participants: Thibaut Girka, Yann Régis-Gianas.

5.4.1. Incrementality in proof languages

In collaboration with Paolo Giarrusso and Yufei Cai (Univ Marburg, Allemagne), Yann Régis-Gianas developed a new method to incrementalise higher-order programs using formal derivatives and static caching. Yann Régis-Gianas has developed a mechanized proof for this transformation. A paper will be submitted to ICFP 2017.

5.4.2. Difference languages

In collaboration with David Mentré (Mitsubishi), Thibaut Girka and Yann Régis-Gianas have developed a theoretical framework to define a notion of differential operational semantics: a general mathematical object to characterise the difference of behavior of two close programs. A paper is under submission. A technical report is available [8].

Thibaut Girka and Yann Régis-Gianas presented this work in several working groups: Gallium (Paris), "Journée annuelle du groupe LTP" of the GDR GPL (Saclay), LIMA (Nantes), IRIF (Paris).

5.5. Metatheory and development of Coq

Participants: Hugo Herbelin, Pierre Letouzey, Yann Régis-Gianas, Matthieu Sozeau.

5.5.1. Dependent pattern-matching

Hugo Herbelin supervised the internship of Meven Bertrand on compiling dependent pattern-matching using a combination of techniques known as small inversion and generalization, as a following of Pierre Boutillier's PhD.

5.5.2. Transferring theorems along isomorphisms

Théo Zimmermann has developed a tool for transferring theorems along isomorphic structures. The long-term objective is to provide a language of proof methods matching the level of abstraction common in mathematics. Théo Zimmermann is applying his tool to introduce higher "mathematical" levels of abstraction to the basic Coq method for applying theorems. The proof of concept of this idea will be presented at the TTT POPL workshop in January.

5.5.3. Unification

Matthieu Sozeau worked in collaboration with Beta Ziliani (assistant professor at Córdoba, Argentina) on a journal version of the formalisation of the unification algorithm used in Coq, which is central for working with advanced type inference features like Canonical Structures. The presentation of this journal version is incremental (it is presented feature by feature), with an aim of easing the understanding of how the algorithm actually works for users who want to take advantage of it. It has been accepted for publication in the Journal of Functional Programming.

5.5.4. Explicit Cumulativity

Pierre Letouzey started exploring with the help of Matthieu Sozeau a version of Coq's logic (CIC) where the cumulativity rule would be explicit. This cumulativity rule is a form of coercion between Coq universes, and is done silently in Coq up to now. Having a version of CIC where the use of the cumulativity bewteen Prop and Type is traceable woud be of great interest. In particular this would lead to a solid ground for the Coq extraction tool and solve some of its current limitations. Moreover, an explicit cumulativity would also help significantly the studies of Coq theoretical models. Preliminary results are encouraging, but this work has not been finalized yet. This work is related to the studies of Ali Assaf (Google Zurich, formerly PhD student in the team Deducteam), but uses different technical choices for different goals. This work is now pursued by Gae¨tan Gilbert (PhD student of Nicolas Tabareau and Matthieu Sozeau at the E´cole des Mines in Nantes), with the goal of providing a version of the calculus of constructions with definitional proof-irrelevance. The absence of explicit cumulativity between Prop and Type was identified in earlier work by Benjamin Werner and Giesik Lee as an important obstacle to building models of the theory, we hence expect this work to simplify the (relative) consistency proof of the theory.

5.6. Formalisation work

Participants: Jean-Jacques Lévy, Daniel de Rauglaudre.

5.6.1. Proofs of algorithms on graphs

Jean-Jacques Lévy and Chen Ran (a PhD student of the Institute of Software, Beijing, visiting the Toccata team) pursue their work about formal proofs of algorithms. Their goal is to provide proofs of algorithms which ought to be both checked by computer and easily human readable. If these kinds of proofs exist for algorithms on inductive structures or recursive algorithms on arrays, they seem less easy to design for combinatorial structures such as graphs. In 2016, they completed proofs for algorithms computing the strongly connected components in graphs. There are mainly two algorithms: one by Kosaraju (1978) working in two phases (some formal proofs of it have already been achieved by Pottier with Coq-classic and by Théry and Gonthier with Coq-ssreflect), one by Tarjan (1972) working in a single pass.

Their proofs use a first-order logic with definitions of inductive predicates. This logic is the one defined in Why3 (research-team Toccata, Saclay). They widely use automatic provers interfaced by Why3. A very minor part of these proofs is also achieved in Coq. The difficulty of this approach is to combine automatic provers and intuitive design.

Part of this work (Tarjan 1972) is presented at JFLA 2017 in Gourette [30] A more comprehensive version is under submission to another conference [34]. Scripts of proofs can be found at http://jeanjacqueslevy.net/why3.

5.6.2. Formalization of theorems in Coq

This section reports on formalisation work by Daniel de Rauglaudre.

5.6.2.1. Puiseux' Theorem

Puiseux' theorem states that the set of Puiseux series (series with rational powers) is an algebraically closed field, i.e. every non-constant polynomial with Puiseux series coefficients admits a zero. This theorem was formalized in Coq a couple of years ago, but it depended on five ad hoc axioms. This year, all these axioms have been grouped together into the only axiom LPO (Limited Principle of Omniscience), stating that for each sequence of booleans, we can decide whether it is always false or if there is at least one true element. This formalized theorem now depends only on this axiom.

5.6.2.2. Banach-Tarski Paradox

Banach-Tarski Paradox states that, if we admit the axiom of choice, a sphere is equidecomposable into two spheres identical to the initial one. The equidecomposability is a property of geometric objects: two objects (sets) are equidecomposable if we can partition them into a same finite number of sets, and each set of the first object is mapped to a set of the second object by only rotations and translations. In other words, we break the first object into a finite number of pieces, and with them, we reconstitute the second object. Its pen and paper proof was done in 1924 by Banach and Tarski.

Its formal proof in Coq has been started this year. About 80% of the proof has been done. The already proved part includes a lemma which says that the sphere without some specific countable number of points is equidecomposable into twice itself. It also includes a formal proof that equidecomposability is an equivalence relation. This makes about 7000 lines of Coq. The remaining part is to formalize the proof that the sphere is equidecomposable into the sphere without this countable set of points.

The version of axiom of choice used for this proof is named TTCA (Type Theoretical Axiom of Choice, introduced by Benjamin Werner [88]), stating that for each equivalence relation, there exists a function mapping each relation class to one of its elements.

SUMO Project-Team

7. New Results

7.1. Analysis and verification of quantitative systems

7.1.1. Quantitative verification of distributions of stochastic models

Participant: Blaise Genest.

In [24], we obtained conditions under which quantitative verification of distributions of stochastic systems is decidable. This is a challenging question as for general Markov Chains, verification of distribution is Skolem-complete, a problem on linear recurrence sequences whose decidability is a long-standing problem open for 40 years. In this paper, we approach this problem by studying the languages generated by Markov Chains, whose regularity would entail the decidability of quantitative verification. Given an initial distribution, we represent the trajectory of Markov Chain over time as an infinite word over a finite alphabet, where the n^{th} letter represents a probability range after n steps. We extend this to a language of trajectories (a set of words), one trajectory for each initial distribution from a (possibly infinite) set. We show that if the eigenvalues of the transition matrix associated with the Markov Chain are all distinct positive real numbers, then the language is *effectively regular*. Further, we show that this result is at the boundary of regularity, as non-regular languages can be generated when the restrictions are even slightly relaxed. The regular representation of the language allows us to reason about more general properties, e.g., robustness of a regular property in a neighbourhood around a given distribution.

7.1.2. Diagnosability of repairable faults

Participants: Éric Fabre, Loïc Hélouët, Hervé Marchand, Engel Lefaucheux.

For (partially observable) discrete event systems, diagnosability characterizes the ability to detect the occurrence of a permanent fault in bounded time after it occurs, given the observations available on that system. Diagnosability can be decided in polynomial time, relying on the so-called twin-machine construction. We have examined the case of repairable faults, and a notion of diagnosability that requires the detection of the fault before it is repaired. It was proved in [35]that diagnosability is a PSpace complete problem.

7.1.3. Diagnosability of stochastic systems

Participants: Éric Fabre, Blaise Genest, Hugo Bazille, Ocan Sankur.

Diagnosis of partially observable stochastic systems prone to faults was introduced in the late nineties. Diagnosability, i.e. the existence of a diagnoser, may be specified in different ways: (1) exact diagnosability (called A-diagnosability) requires that almost surely a fault is detected and that no fault is erroneously claimed while (2) approximate diagnosability (called ε-diagnosability) allows a small probability of error when claiming a fault and (3) accurate approximate diagnosability (called AA-diagnosability) requires that this error threshold may be chosen arbitrarily small. In a recent work [27], we focused on approximate diagnoses. We first refined the almost sure requirement about finite delay introducing a uniform version and showing that while it does not discriminate between the two versions of exact diagnosability this is no more the case in approximate diagnosis. We then gave a complete picture of relations between the different diagnosability specifications for probabilistic systems and establish characterisations for most of them in the finite-state case. Based on these characterisations, we developped decision procedures, studied their complexity and proved their optimality. We also designed synthesis algorithms to construct diagnosers and we analysed their memory requirements. Finally we established undecidability of the diagnosability problems for which we provided no characterisation. Notably, we proved the AA-diagnosability problem to be undecidable, answering a longstanding open question.

In another work [28], we investigated semantical and computational issues for exact notions of diagnosability in the context of infinite-state probabilistic systems. We first showed established a characterisation of the so-called FF-diagnosability using a $G\delta$ set (instead of an open set for finite-state systems) and also for two other notions, IF- and IA-diagnosability, when models are finitely branching. We also proved that surprisingly the last notion, FA-diagnosability, cannot be characterised in this way even in the finitely branching case. Then we applied our characterisations for a partially observable probabilistic extension of visibly pushdown automata, yielding EXPSPACE procedures for solving diagnosability problems. In addition, we establish some computational lower bounds and show that slight extensions of these probabilistic visibly pushdown automata lead to undecidability.

7.1.4. Analysing decisive stochastic processes

Participant: Nathalie Bertrand.

In 2007, Abdulla et al. introduced the elegant concept of decisive Markov chain. Intuitively, decisiveness allows one to lift the good properties of finite Markov chains to infinite Markov chains. For instance, the approximate quantitative reachability problem can be solved for decisive Markov chains (enjoying reasonable effectiveness assumptions) including probabilistic lossy channel systems and probabilistic vector addition systems with states. In a recent work [26], we extended the concept of decisiveness to more general stochastic processes. This extension is non trivial as we consider stochastic processes with a potentially continuous set of states and uncountable branching (common features of real-time stochastic processes). This allowed us to obtain decidability results for both qualitative and quantitative verification problems on some classes of real-time stochastic processes, including generalized semi-Markov processes and stochastic timed automata.

7.1.5. Concurrent timed systems

Participants: Loïc Hélouët, Blaise Genest.

Adding real time information to Petri net models often leads to undecidability of classical verification problems such as reachability and boundedness. For instance, models such as Timed-Transition Petri nets (TPNs) [47] are intractable except in a bounded setting. On the other hand, the model of Timed-Arc Petri nets [50] enjoys decidability results for boundedness and control-state reachability problems at the cost of disallowing urgency (the ability to enforce actions within a time delay).

We have addressed semantics variants of time and timed Petri nets to obtain concurrent models with interesting expressive power, but yet allowing decidability of verification and robustness questions. Robustness of timed systems aims at studying whether infinitesimal perturbations in clock values can result in new discrete behaviors. A model is robust if the set of discrete behaviors is preserved under arbitrarily small (but positive) perturbations.

In [25] we have considered time in Petri nets under a strong semantics with multiple enabling of transitions. We focus on a structural subclass of unbounded TPNs, where the underlying untimed net is free-choice, and show that it enjoys nice properties under a multi-server semantics. In particular, we showed that the questions of fireability (whether a chosen transition can fire), and termination (whether the net has a non-terminating run) are decidable for this class. We then consider the problem of robustness under guard enlargement [48], i.e., whether a given property is preserved even if the system is implemented on an architecture with imprecise time measurement. Unlike in [15], where decidability of several problems is obtained for bounded classes of nets, we showed that robustness of fireability is decidable for unbounded free choice TPNs with a multi-server semantics.

The robustness of time Petri nets was addressed in [15] by considering the model of parametric guard enlargement which allows time-intervals constraining the firing of transitions in TPNs to be enlarged by a (positive) parameter. We show that TPNs are not robust in general and checking if they are robust with respect to standard properties (such as boundedness, safety) is undecidable. We then extend the marking class timed automaton construction for TPNs to a parametric setting, and prove that it is compatible with guard enlargements. We apply this result to the (undecidable) class of TPNs which are robustly bounded (i.e., whose finite set of reachable markings remains finite under infinitesimal perturbations): we provide two decidable

robustly bounded subclasses, and show that one can effectively build a timed automaton which is timed bisimilar even in presence of perturbations. This allows us to apply existing results for timed automata to these TPNs and show further robustness properties.

The goal of [23] is to investigate decidable classes of Petri nets with time that capture some urgency and still allow unbounded behaviors, which go beyond finite state systems. We have shown, up to our knowledge, the first decidability results on reachability and boundedness for Petri net variants that combine unbounded places, time, and urgency. For this, we have introduced the class of Timed-Arc Petri nets with restricted Urgency, where urgency can be used only on transitions consuming tokens from bounded places. We showed that control-state reachability and boundedness are decidable for this new class, by extending results from Timed-Arc Petri nets (without urgency) [43]. Our main result concerns (marking) reachability, which is undecidable for both TPNs (because of unrestricted urgency) [46] and Timed-Arc Petri Nets (because of infinite number of "clocks") [49]. We obtained decidability of reachability for unbounded TPNs with restricted urgency under a new, yet natural, timed-arc semantics presenting them as Timed-Arc Petri Nets with restricted urgency. Decidability of reachability under the intermediate marking semantics is also obtained for a restricted subclass.

7.1.6. Petri nets realizability

Participants: Loïc Hélouët, Abd El Karim Kecir.

We considered in [30] the realizability of urban train schedules by stochastic concurrent timed systems. Schedules are high level views of desired timetables that a metro system should implement. They are represented as partial orders decorated with timing constraints. Train systems are represented as elementary stochastic time Petri nets. We have first considered logical realizability: a schedule is realizable by a net $\mathcal{N}$ if it embeds in a time process of $\mathcal{N}$ that satisfies all its constraints. However, with continuous time domains, the probability of a time process that realizes a schedule is null. We have extended the former notion of realizability to consider probabilistic realizability of schedules up to some imprecision α. This probabilistic realizability holds if the probability that $\mathcal{N}$ logically realizes S with constraints enlarged by α time units is strictly positive. We have shown that upon a sensible restriction guaranteeing time progress (systems can not perform an arbitrary number of actions within a single time unit), logical and probabilistic realizability of a schedule can be checked on the finite set of symbolic prefixes extracted from a bounded unfolding of the net. We have provided a construction technique for these prefixes and shown that they represent all time processes of a net occurring up to a given maximal date. We have then shown how to verify existence of an embedding and compute the probability of its realization.

7.2. Control of quantitative systems

7.2.1. Smart regulation for urban trains

Participants: Éric Fabre, Loïc Hélouët, Hervé Marchand, Abd El Karim Kecir.

The regulation of subway lines consists in accomodating small random perturbations in transit times as well as more impacting incidents, by playing on continuous commands (transit times and dwell times) and by making more complex decisions (insertions or extractions of trains, changes of missions, overpassing, shorter returns, etc.). The objectives are multiple : ensuring the regularity and punctuality of trains, adapting to transportation demand, minimizing energy consumption, etc. We have developed an event-based control strategy that aims at equalizing headways on a line. This distributed control strategy is remarquably robust to perturbations and reactive enough to accomodate train insertions/extractions. We have also developed another approach based on event graphs in order to optimally interleave trains at a junction.

7.2.2. Games and reactive synthesis

Participant: Ocan Sankur.

In game theory, a strategy is *dominated* by another one if the latter systematically yields a payoff as good as the former, while also yielding a better payoff in some cases. A strategy is *admissible* if it is not dominated. This notion is well studied in game theory and is useful to describe the set of strategies that are "reasonable" whose choice can be justified. Recent works studied this notion in graph games with omega-regular objectives and investigated its applications in controller synthesis. For multi-agent controller synthesis, admissibility can be used as a hypothesis on the behaviors of each agent, thus enabling a compositional reasoning framework for controller synthesis. In [29], we investigate this framework for quantitative graph games. We characterize admissible strategies, study their existence, and give an effective characterization of the set of paths that are compatible with admissible payoffs. This is then used to derive algorithms for model checking under admissibility, but also assume-admissible synthesis.

In [21], we present the reactive synthesis competition (SYNTCOMP), a long-term effort intended to stimulate and guide advances in the design and application of synthesis procedures for reactive systems. The first iteration of SYNTCOMP is based on the controller synthesis problem for finite-state systems and safety specifications. We provide an overview of this problem and existing approaches to solve it, and report on the design and results of the first SYNTCOMP. This includes the definition of the benchmark format, the collection of benchmarks, the rules of the competition, and the five synthesis tools that participated. We present and analyze the results of the competition and draw conclusions on the state of the art. Finally, we give an outlook on future directions of SYNTCOMP.

In the invited [22], we summarize new solution concepts useful for the synthesis of reactive systems that we have introduced in several recent publications. These solution concepts are developed in the context of non-zero sum games played on graphs. They include the assume-admissible synthesis on Boolean games, synthesis under multiple environments for Markov decision processes, and multi-objective synthesis with probability thresholds for Markov decision processes with multi-dimensional weights. They are part of the contributions obtained in the inVEST project funded by the European Research Council.

7.2.3. Runtime enforcement

Participants: Hervé Marchand, Thierry Jéron.

In the [20] we generalize our line of work on runtime enforcement for timed properties. Runtime enforcement is a verification/validation technique aiming at correcting possibly incorrect executions of a system of interest. In this work we consider enforcement monitoring for systems where the physical time elapsing between actions matters. Executions are thus modelled as timed words (i.e., sequences of actions with dates). We consider runtime enforcement for timed specifications modelled as timed automata. Our enforcement mechanisms have the power of both delaying events to match timing constraints, and suppressing events when no delaying is appropriate, thus possibly allowing for longer executions. To ease their design and their correctness-proof, enforcement mechanisms are described at several levels: enforcement functions that specify the input-output behaviour in terms of transformations of timed words, constraints that should be satisfied by such functions, enforcement monitors that describe the operational behaviour of enforcement functions, and enforcement algorithms that describe the implementation of enforcement monitors.

This year we went one step ahead [33] and consider predictive runtime enforcement, where the system is not entirely black-box, but we know something about its behavior. This *a priori* knowledge about the system allows to output some events immediately, instead of delaying them until more events are observed, or even blocking them permanently. This in turn results in better enforcement policies. We also show that if we have no knowledge about the system, then the proposed enforcement mechanism reduces to a classical non-predictive runtime enforcement framework. All our results are formalized and proved in the Isabelle theorem prover.

7.2.4. Decentralized control

Participant: Hervé Marchand.

In collaboration with Laurie Ricker, we have been interested in decentralized control of discrete event systems. In decentralized discrete-event system (DES) architectures, agents fuse their local decisions to arrive at the global decision. The contribution of each agent to the final decision is never assessed; however, it may be the case that only a subset of agents, i.e., a (static) coalition, perpetually contribute towards the correct final decisions. In casting the decentralized DES control (with and without communication) problem as a cooperative game, it is possible to quantify the average contribution that each agent makes towards synthesizing the overall correct control strategy. Specifically, we explore allocations that assess contributions of non-communicating and communicating controllers for this class of problems. This allows a quantification of the contribution that each agent makes to the coalition with respect to decisions made solely based on its partial observations and decisions made based on messages sent to another agent(s) to facilitate a correct control decision [34].

7.3. Management of large distributed systems

7.3.1. Non-interference in partial order models

Participant: Loïc Hélouët.

We obtained new results on security issues such as non-interference [41]. Noninterference (NI) is a property of systems stating that confidential actions should not cause effects observable by unauthorized users. Several variants of NI have been studied for many types of models but rarely for true concurrency or unbounded models. In [45], we had already demonstrated the discriminating power of partial orders, and investigated NI for High-level Message Sequence Charts (HMSCs), a partial order language for the description of distributed systems. We had proposed a general definition of security properties in terms of equivalence among observations of behaviors, and showed that equivalence, inclusion, and NI properties were undecidable for HMSCs. We defined a new formalism called *partial order automata*, that captures natural observations of distributed systems, and in particular observations of HMSCs. It generalizes HMSCs and permits assembling partial orders. We have then considered subclasses of partial order automata and HMSCs for which Non-Interference is decidable. This allowed us to exhibit more classes of HMSCs for which NI is decidable. Finally, we have defined weaker local Non-interfernce properties, describing situations where a system is attacked by a single agent, and shown that local NI is decidable. We have then refined local NI to a finer notion of causal NI that emphasizes causal dependencies between confidential actions and observations and extended it to causal NI with (selective) declassification of confidential events, which allows to consider that confidential actions need can be kept secret for a limited duration and can then be declassified. Checking whether a system satisfies local and causal NI and their declassified variants are PSPACE-complete problems.

7.3.2. Simulations for stochastic abstractions of large systems

Participants: Éric Fabre, Blaise Genest, Matthieu Pichené.

In [32], we developped a new simulation strategy to accurately simulate DBNs (Dynamic Bayesian Networks) obtained as stochastic abstractions of large systems. The DBN abstractions are given under the form of probability tables, describing the probability for a variable to take a given value given the values of some variables at the previous time point. To be able to handle large systems with many variables, there is a table for each variable (coupling between variable is not explicitely represented). This creates discrepencies when simulating variables independently. Our new algorithm simulates tuples of variables together by looking ahead for such discrepencies in order to avoid them. Such simulations are still efficient, and match more faithfully the original systems.

7.4. Data driven systems

7.4.1. Structured data nets

Participants: Éric Badouel, Loïc Hélouët, Christophe Morvan.

In [16] we proposed a Petri net extension, called Structured Data Nets (SDN), that describes transactional systems with data. In these nets, tokens are semi-structured documents. Each transition is attached to a query, guarded by patterns, (logical assertions on the contents of its preset) and transforms tokens.

We define SDNs and their semantics and consider their formal properties: coverability of a marking, termination and soundness of transactions.

Unrestricted SDNs are Turing complete, so these properties are undecidable. We thus used an order on documents, and showed that under reasonable restrictions on documents and on the expressiveness of patterns and queries, SDNs are well-structured transition systems, for which coverability, termination and soundness are decidable.

7.4.2. An active workspace model for disease surveillance

Participant: Éric Badouel.

Flexibility and change at both design- and run-time are fast becoming the Rule rather than the Exception in Business Process Models. This is attributed to the continuous advances in domain knowledge, the increase in expert knowledge, and the diverse and heterogeneous nature of contextual variables. Such processes are characterized by collaborative work and decision making between users with heterogeneous profiles on a processes designed on-the-fly. A model for such processes should thus natively support human interactions. We showed in [31] how the Active Workspaces model proposed [44] for distributed collaborative systems supports these interactions.

TOCCATA Project-Team

7. New Results

7.1. Deductive Verification

A bit-vector library for deductive verification. C. Fumex and C. Marché developed a new library for bit-vectors, in Why3 and SPARK. This library is rich enough for the formal specification of functional behavior of programs that operate at the level of bits. It is also designed to exploit efficiently the support for bit-vectors built-in in some SMT solvers. This work is done in the context of the ProofInUse joint laboratory. The SPARK front-end of Why3, for the verification of Ada programs, is extended to exploit this new bit-vector theory. Several cases studies are conducted: efficient search for rightmost bit of a bit-vector, efficient computation of the number of bits set to 1, efficient solving of the n-queens problem. At the level of SPARK, a program inspired from some industrial code (originally developed in C par J. Gerlach, Fraunhofer FOKUS Institute, Germany and partially proved with Frama-C and Coq) is specified in SPARK and proved with automatic solvers only. A paper on that library together with the way it is connected with the built-in support for bitvectors in SMT solver was presented at the NASA Formal methods Conference [24]. The support for bit-vectors is distributed with SPARK since 2015, and SPARK users already reported that several verification conditions, that couldn't be proved earlier, are now proved automatically.

Counterexamples from proof failures. D. Hauzar and C. Marché worked on counterexample generation from failed proof attempts. They designed a new approach for generating potential counterexamples in the deductive verification setting, and implemented in Why3. When the logic goal generated for a given verification condition is not shown unsatisfiable by an SMT solvers, some solver can propose a model. By carefully reverting the transformation chain (from an input program through the VC generator and the various translation steps to solvers), this model is turned into a potential counterexample that the user can exploit to analyze why its original code is not proved. The approach is implemented in the chain from Ada programs through SPARK, Why3, and SMT solvers CVC4 and Z3. This work is described in a research report [35] and a paper was rpesented at the SEFM Conference [25]. The work on the implementation was continued by S. Dailler. It was considered robust enough to be distributed in the release Pro 16 of SPARK.

Static versus dynamic verification. C. Marché, together with Y. Moy from AdaCore, J. Signoles and N. Kosmatov from CEA-LIST, wrote a survey paper about the design of the specification languages of Why3 and its front-ends Frama-C and SPARK. The choices made when designing these specification languages differ significantly, in particular with respect to the executability of specifications. The paper reviews these differences and the issues that result from these choices. The paper also emphasizes two aspects where static and dynamic aspects of the specifications play an important role: the specific feature of *ghost code*, and the techniques that help users understand why static verification fails. This paper was presented at the Isola Symposium [26].

Higher-Order Representation Predicates. A. Charguéraud investigated how to formalize in Separation Logic representation predicates for describing mutable container data structures that store mutable elements that are themselves described using representation predicates. (In Separation Logic, representation predicates are used to describe mutable data structures, by establishing a relationship between the entry point of the structure, the piece of heap over which this structure spans, and the logical model associated with the structure.) The solution proposed, based on "higher-order representation predicates", allows for concise specifications of such containers. A. Charguéraud has published a paper presenting, through a collection of practical examples, solutions to the challenges associated with verification proofs based on higher-order representation predicates [19].

Temporary Read-Only Permissions for Separation Logic A. Charguéraud and François Pottier (Inria Paris) have developed an extension of Separation Logic with temporary read-only permissions. This mechanism allows to temporarily convert any assertion (or "permission") to a read-only form. Unlike with fractional permissions, no accounting is required: the proposed read-only permissions can be freely duplicated and discarded. Where mutable data structures are temporarily accessed only for reading, the proposed read-only permissions enable more concise specifications and proofs. All the metatheory is verified in Coq. An article has been submitted to a conference [20].

Reasoning About Iteration. J.-C. Filliâtre and M. Pereira proposed a new approach to the problem of specifying iteration, verifying iterators (such as cursors or higher-order functions), and using iterators. The idea is to characterize the sequence of elements enumerated so far, and only those. The proof methodology is modular, iterator implementations and clients being verified independently of each other. The proposed method is validated experimentally in Why3. This work has been published first at JFLA 2016 [33] and then at NFM 2016 [22]. A journal version of this work is under submission.

Defunctionalization for proving higher-order programs. J.-C. Filliâtre and M. Pereira proposed a new approach to the verification of higher-order programs, using the technique of defunctionalization, that is, the translation of first-class functions into first-order values. This is an early experimental work, conducted on examples only within the Why3 system. This work has been published at JFLA 2017 [30].

A Type System for Deductive Verification. J.-C. Filliâtre, L. Gondelman, and A. Paskevich proposed a practical method to track pointer aliases statically in a large family of computer programs. Their approach relies on a special type system with singleton regions and effects which both can be inferred automatically, without requiring additional user annotations. This kind of static analysis is important for deductive program verification, since it allows us to construct verification conditions using the traditional rules in the spirit of Hoare and Dijkstra, without recurring to more sophisticated solutions (memory models, separation logic) which incur additional complexity both for a user and a verification tool. The proposed method is implemented in Why3 and described in a technical report [37].

Ghost Code. J.-C. Filliâtre, L. Gondelman, and A. Paskevich published a paper on a general approach to the concept of ghost code in the journal of *Formal Methods in System Design* [14]. Ghost code is a subset of program code that serves the purposes of specification and verification: it can be erased from the program without affecting its result. This work forms the basis of the support for ghost code in Why3. This work is an extended version of the paper presented at the 26th International Conference on Computer Aided Verification (CAV) in 2014.

7.2. Automated Reasoning

Decision Procedures via Axiomatizations with Triggers. C. Dross, A. Paskevich, J. Kanig and S. Conchon published a paper in the *Journal of Automated Reasoning* [13] about integration of first-order axiomatizations with triggers as decision procedures in an SMT solver. This work extends a part of C. Dross PhD thesis [83]. A formal semantics of the notion of trigger is presented, with a general setting to show how a first-order axiomatization with triggers can be proved correct, complete, and terminating. An extended DPLL(T) algorithm can then integrate such an axiomatization with triggers, as a decision procedure for the theory it defines.

Lightweight Approach for Declarative Proofs. M. Clochard designed an extension of first-order logic, for describing reasoning steps needed to discharge a proof obligation. The extension is under the form of two new connectives, called proof indications, that allow the user to encode reasoning steps inside a logic formula. This extension makes possible to use the syntax of formulas as a proof language. The approach was presented at the JFLA conference [29] and implemented in Why3. It brings a lightweight mechanism for declarative proofs in an environment like Why3 where provers are used as black boxes. Moreover, this mechanism restricts the scope of auxiliary lemmas, reducing the size of proof obligations sent to external provers.

7.3. Certification of Algorithms, Languages, Tools and Systems

Case study: Matrix Multiplication. M. Clochard, L. Gondelman and M. Pereira wrote a paper describing a complete solution for the first challenge of the VerifyThis 2016 competition held at the 18th ETAPS Forum, where they obtain the award for the best student team. Two variants for the multiplication of matrices are presented and proved: a naive version using three nested loops and Strassen's algorithm. To formally specify the two multiplication algorithms, they developed a new Why3 theory of matrices, and they applied a reflection methodology to condut some of the proofs. This work was presented at the VSTTE Conference [21]. An extended version that considers arbitrary rectangular martices instead of square ones is in preparation. The development is available at http://toccata.lri.fr/gallery/verifythis_2016_matrix_multiplication.en.html.

Case study: Koda-Ruskey's algorithm for generating ideals of a forest. J.-C. Filliâtre and M. Pereira presented the first formal proof of an implementation of Koda and Ruskey's algorithm (an algorithm for generating all ideals of a forest poset as a Gray code) at VSTTE 2016 [23]. The proof is conducted within the Why3 system and is mostly automatic.

The Lax-Milgram Theorem. S. Boldo, F. Clément, F. Faissole, V. Martin, and M. Mayero have worked on a Coq formal proof of the Lax–Milgram theorem. The Finite Element Method is a widely-used method to solve numerical problems coming for instance from physics or biology. To obtain the highest confidence on the correction of numerical simulation programs implementing the Finite Element Method, one has to formalize the mathematical notions and results that allow to establish the sound-ness of the method. The Lax–Milgram theorem may be seen as one of those theoretical cornerstones: under some completeness and coercivity assumptions, it states existence and uniqueness of the solution to the weak formulation of some boundary value problems. This article presents the full formal proof of the Lax–Milgram theorem in Coq. It requires many results from linear algebra, geometry, functional analysis , and Hilbert spaces. This has been published at the 6th ACM SIGPLAN Conference on Certified Programs and Proofs (CPP 2017) [18].

ALEA library extended with continuous datatypes The ALEA library uses a monadic construction to formalize discrete measure theory. F. Faissole and B. Spitters proposed to extend it to continuous datatypes. They used both synthetic topology and homotopy type theory to achieve the formalization. This work is presented at the Workshop on Coq for Programming Languages [32].

Case study: Strongly Connected Components of a Graph R. Chen and J.-J. Lévy designed a formal proof of Tarjan's algorithm for computing the strongly connected component of a directed graph. The proof is conducted using Why3. This work is presented at the JFLA conference [28]. This case study is part of a larger set of case studies on algorithms on graphs http://pauillac.inria.fr/~levy/why3/.

Case study: Unix Pathname Resolution R. Chen, M. Clochard and C.-Marché designed a formal proof of an algorithm for resolving a pathname in Unix file systems. The proof is conducted using Why3 [34]. This case study is part of the CoLiS project.

7.4. Floating-Point and Numerical Programs

-

Interval arithmetic and Taylor models. É. Martin-Dorel and G. Melquiond have 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. This work has been published in the *Journal of Automated Reasoning* [15].

Interval arithmetic and univariate integrals. A. Mahboubi, G. Melquiond, and T. Sibut-Pinote have extended the CoqInterval library with support for definite univariate integrals. The library is now able to automatically and formally verify bounds on the value of integrals by computing rigorous polynomial approximations of integrands. This work has been presented at the 7th International Conference on Interactive Theorem Proving [27].

Robustness of 2Sum and Fast2Sum. S. Boldo, S.Graillat, and J.-M. Muller have worked on the 2Sum and Fast2Sum algorithms, that are important building blocks in numerical computing. They are used (implicitely or explicitely) in many compensated algorithms or for manipulating floating-point expansions. They showed that these algorithms are much more robust than it is usually believed: the returned result makes sense even when the rounding function is not round-to-nearest, and they are almost immune to overflow. This work has been submitted [36].

Computing error bounds without changing the rouding mode. S. Boldo has created an algorithm to compute a correct and tight rounding error bound for a floating-point computation. The rounding error can be bounded by folklore formulas, such as $\varepsilon|x|$ or $\varepsilon|\circ(x)|$. This gets more complicated when underflow is taken into account. To compute this error bound in practice, a directed rounding is usually used. This work describes an algorithm that computes a correct bound using only rounding to nearest, therefore without requiring a costly change of the rounding mode. This is formally proved using the Coq formal proof assistant to increase the trust in this algorithm. This has been published at the 9th International Workshop on Numerical Software Verification [17].

Floating-Point Computations and Iterators. S. Boldo has worked on the formal verification of a floating-point case study where the common iterators `fold_left` and `fold_right` have not the wanted behaviors. She then had to define other iterators, which are very similar in most cases, but that do behave well in our case study. This has been published at the 1st Workshop on High-Consequence Control Verification [31].

VERIDIS Project-Team

7. New Results

7.1. Automated and Interactive Theorem Proving

Participants: Gabor Alági, Haniel Barbosa, Jasmin Christian Blanchette, Martin Bromberger, Simon Cruanes, Mathias Fleury, Pascal Fontaine, Marek Košta, Stephan Merz, Martin Riener, Martin Strecker, Thomas Sturm, Marco Voigt, Uwe Waldmann, Daniel Wand, Christoph Weidenbach.

7.1.1. IsaFoL: Isabelle Formalization of Logic

Joint work with Heiko Becker (MPI-SWS Saarbrücken), Peter Lammich (TU München), Andrei Popescu (Middlesex University London), Anders Schlichtkrull (DTU Copenhagen), Dmitriy Traytel (ETH Zürich), and Jørgen Villadsen (DTU Copenhagen).

Researchers in automated reasoning spend a significant portion of their work time specifying logical calculi and proving metatheorems about them. These proofs are typically carried out with pen and paper, which is error-prone and can be tedious. As proof assistants are becoming easier to use, it makes sense to employ them.

In this spirit, we started an effort, called IsaFoL (Isabelle Formalization of Logic), that aims at developing libraries and methodology for formalizing modern research in the field, using the Isabelle/HOL proof assistant.[0] Our initial emphasis is on established results about propositional and first-order logic. In particular, we are formalizing large parts of Weidenbach's forthcoming textbook, tentatively called *Automated Reasoning—The Art of Generic Problem Solving*.

The objective of formalization work is not to eliminate paper proofs, but to complement them with rich formal companions. Formalizations help catch mistakes, whether superficial or deep, in specifications and theorems; they make it easy to experiment with changes or variants of concepts; and they help clarify concepts left vague on paper.

The repository contains six completed entries and three entries that are still in development. Notably:

- Mathias Fleury formalized a SAT solver framework with learn, forget, restart, and incrementality and published the result at a leading conference, together with Jasmin Blanchette and Christoph Weidenbach [25].
- Anders Schlichtkrull, remotely co-supervised by Jasmin Blanchette, formalized unordered first-order resolution in Isabelle and presented the result at ITP 2016 [37].
- Together with an intern, Jasmin Blanchette, Uwe Waldmann, and Daniel Wand formalized a generalization for the recursive path order and the transfinite Knuth-Bendix order to higher-order terms without λ-abstractions. The result is published in the Isabelle *Archive of Formal Proofs*.

7.1.2. Combination of Satisfiability Procedures

Joint work with Christophe Ringeissen from the PESTO project-team at Inria Nancy – Grand Est, and Paula Chocron at IIIA-CSIC, Bellaterra, Catalonia, Spain.

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).

[0] https://bitbucket.org/jasmin_blanchette/isafol/wiki/Home

In 2015, we defined [42] a sound and complete combination procedure à la Nelson-Oppen for the theory of absolutely free data structures (including lists and trees) connected to another theory via bridging functions. This combination procedure has also been refined for standard interpretations. The resulting theory has a nice politeness property, enabling combinations with arbitrary decidable theories of elements. We also investigated [43] other theories amenable to similar combinations: this class includes the theory of equality, the theory of absolutely free data structures, and all the theories in between.

More recently, we have been improving the framework and unified both results. A new paper is in preparation.

7.1.3. Quantifier handling in SMT

Joint work with Andrew J. Reynolds, Univ. of Iowa, USA.

SMT solvers generally rely on various instantiation techniques to handle quantifiers. We are building a unifying framework for handling quantified formulas with equality and uninterpreted functions, such that the major instantiation techniques in SMT solving can be cast in that framework. It is based on the problem of E-ground (dis)unification, a variation of the classic Rigid E-unification problem. We introduced a sound and complete calculus to solve this problem in practice: Congruence Closure with Free Variables (CCFV). Experimental evaluations of implementations of CCFV in the state-of-the-art solver CVC4 and in the solver veriT exhibit improvements in the former and makes the latter competitive with state-of-the-art solvers in several benchmark libraries stemming from verification efforts. A publication is in preparation.

7.1.4. Non-linear arithmetic in SMT

In the context of the SMArT ANR-DFG (Satisfiability Modulo Arithmetic Theories) and KANASA projects (cf. sections 9.1 and 9.3), we study the theory, design techniques, and implement software to push forward the non-linear arithmetic (NLA) reasoning capabilities in SMT. This year, we designed a framework to combine interval constraint propagation with other decision procedures for NLA, with promising results. We are also currently studying integration of these procedures into combinations of theories. The ideas are validated within the veriT solver, together with code from the raSAT solver (from JAIST). An article is in preparation.

7.1.5. Encoding Set-Theoretic Formulas in First-Order Logic

Proof obligations that arise during the verification of high-level specifications of algorithms in languages such as (Event-)B and TLA$^+$ mix theories corresponding to sets, functions, arithmetic, tuples, and records. Finding encodings of such formulas in the input languages of automatic first-order provers (superposition-based provers or SMT solvers, which are based on multi-sorted first-order logic) is paramount for obtaining satisfactory levels of automation. We describe a method, based on a combination of injection of unsorted expressions into sorted languages, simplification by rewriting, and abstraction, that is the kernel of the SMT backend of the TLA$^+$ proof system (section 6.4). A paper describing our technique was presented at ABZ 2016 [31] and an extension of that article was invited for a special issue of Science of Computer Programming.

During the internship of Matthieu Lequesne, we experimented with an adaptation of the technique for constructing models of formulas in set theory, which could be useful for understanding why proof attempts fail. A prototype generating input for the Nunchaku model finder (section 6.1) allowed us to validate the idea for a core sublanguage of TLA$^+$.

7.1.6. Modal and Description Logics for Graph Transformations

Graph transformations are a research topic that is interesting in its own right, but with many possible applications ranging from the modification of pointer structures in imperative programs, through model transformations in model-driven engineering, to schema-preserving transformations of graph databases. Our particular focus is on verifying these transformations.

Modal logics and variants (such as description logics that are the basis for the web ontology language OWL) have turned out to be suitable specification formalisms because graph structures can typically be perceived as models of modal logics, but these logics suffer from some weaknesses when reasoning about transformations. The first aim of our work was therefore to identify and define sufficiently expressive modal logics, while retaining their pleasant properties, in particular decidability [30].

Our next aim is to implement practically useful proof methods. We have first concentrated on the more natural tableau proofs, with a verification of meta-theoretic properties of the calculi (such as termination) in the Isabelle proof assistant. We now turn to an investigation of encodings as satisfiability problems that can be handled with SAT and SMT solvers, with the hope to achieve a better performance.

7.1.7. Standard Models with Virtual Substitution

Joint work with A. Dolzmann from Leibniz-Zentrum für Informatik in Saarbrücken, Germany.

Extended quantifier elimination for the reals using virtual substitution methods have been successfully applied to various problems in science and engineering. Recently they have attracted attention also as theory solvers within SMT. Such solvers typically ask also for models in the satisfiable case. Models obtained with virtual substitution are in general obtained in certain non-archimedian extension fields of the reals with a corresponding expanded signature. Consequently, the obtained values for the variables include non-standard symbols such as positive infinitesimals and infinite values.

We introduce a complete post-processing procedure to convert our models, for fixed values of parameters, into real models [15]. We furthermore demonstrate the successful application of an implementation of our method within Redlog to a number of extended quantifier elimination problems from the scientific literature including computational geometry, motion planning, bifurcation analysis for models of genetic circuits and for mass action, and sizing of electrical networks. This solves a long-standing problem with the virtual substitution method, which had been explicitly criticized in the scientific literature.

7.1.8. Decidability of Fragments of Free First-Order Logic

We introduce a new decidable fragment of first-order logic with equality, the *Separated Fragment* (SF). It strictly generalizes two already well-known decidable fragments of first-order logic: the Bernays-Schönfinkel-Ramsey (BSR) Fragment and the Monadic Fragment. The defining principle is that universally and existentially quantified variables may not occur together in atoms. Thus, our classification neither rests on restrictions of quantifier prefixes (as in the BSR case) nor on restrictions on the arity of predicate symbols (as in the monadic case).

We show that SF exhibits the finite model property and derive a non-elementary upper bound on the computing time required for deciding satisfiability of SF sentences. For the subfragment of prenex sentences with the quantifier prefix $\exists^*\forall^*\exists^*$ the satisfiability problem is shown to be NEXPTIME-complete. Furthermore, we discuss how automated reasoning procedures can take advantage of our results [34].

Continuing the work presented in the initial publication, we further investigated the computational complexity of SF satisfiability. It nicely scales across the nondeterministic standard complexity classes, depending on joint occurrences of existentially quantified variables from $\exists^*$-blocks that are separated by nonempty $\forall^+$-blocks.

In another line of work, we relaxed the definition of SF, leading to an even larger fragment for which satisfiability is still decidable. In this fragment, variables of $\exists^*$-blocks and $\forall^+$-blocks may occur together in some atom if the respective quantifiers obey a certain order.

7.1.9. Ordered resolution with dismatching constraints

The identification and algorithmic exploration of decidable logic fragments is key to the automation of logics and to obtaining push-button verification technologies. We extend a well-known decidable fragment, linear monadic shallow Horn theories, with straight dismatching constraints, preserving decidability. Furthermore, we show that the restriction to Horn clauses is not needed. The fragment has various applications in security, automata theory and theorem proving [35].

7.1.10. Undecidable combinations of first-order logic with background theories

We show that the universal fragment of Presburger arithmetic augmented with a single uninterpreted predicate (or function) symbol is already undecidable. The result has immediate consequences for verification techniques that combine uninterpreted functions or predicate symbols with (fragments of) Presburger arithmetic. For example, data structures such as arrays can be viewed as a collection of uninterpreted functions that obey certain axioms.

Our result is a sharpening of previously known results. In particular, undecidability holds for a fragment with purely universal quantification: no quantifier alternation is necessary. While in this case the set of unsatisfiable sentences is still recursively enumerable, and in fact hierarchic superposition constitutes a semi-decision procedure, allowing for one quantifier alternation ($\exists\forall$ or $\forall\exists$) leads to a fragment in which neither the satisfiable sentences nor the unsatisfiable ones form a recursively-enumerable set. Hence, there cannot be any refutationally complete calculus for such a combined theory.

7.1.11. Novel techniques for linear arithmetic constraint solving

In [26], [27], we investigate new techniques for linear arithmetic constraint solving. They are based on the linear cube transformation, which allows us to efficiently determine whether a system of linear arithmetic constraints contains a hypercube of a given edge length.

Our first findings based on this transformation are two sound tests that find integer solutions for linear arithmetic constraints. While many complete methods search along the problem surface for a solution, these tests use cubes to explore the interior of the problems. The tests are especially efficient for constraints with a large number of integer solutions, e.g., those with infinite lattice width. Inside the SMT-LIB benchmarks, we have found almost one thousand problem instances with infinite lattice width. Experimental results confirm that our tests are superior on these instances compared to several state-of-the-art SMT solvers.

We also discovered that the linear cube transformation can be used to investigate the equalities implied by a system of linear arithmetic constraints. For this purpose, we developed a method that computes a basis for all implied equalities, i.e., a finite representation of all equalities implied by the linear arithmetic constraints. The equality basis can be used to decide whether a system of linear arithmetic constraints implies a given equality.

7.2. Formal Methods for Developing and Analyzing Algorithms and Systems

Participants: Noran Azmy, Gabriel Corona, Margaux Duroeulx, Marie Duflot-Kremer, Souad Kherroubi, Dominique Méry, Stephan Merz, Nicolas Schnepf, Christoph Weidenbach.

7.2.1. Making explicit domain knowledge in formal system development

Joint work with partners of the IMPEX project.

Modeling languages are concerned with providing techniques and tool support for the design, synthesis and analysis of the models resulting from a given modeling activity, and this activity is usually part of a system development model or process. These languages quite successfully focus on the analysis of the designed system, exploiting the semantic features of the underlying modeling language. These semantics are well understood by the system designers and/or the users of the modeling language, that is why we speak of implicit semantics.

In general, modeling languages are not equipped with resources, concepts or entities handling explicitly domain engineering features and characteristics (domain knowledge) in which the modeled systems evolve.

We posit that designers should explicitly handle the knowledge resulting from an analysis of the application domain, i.e. explicit semantics. As of today, making explicit the domain knowledge inside system design models does not follow any methodological rule; instead, features of domain knowledge are introduced in an ad-hoc way through types, constraints, profiles, etc.

Our claim [11] is that ontologies are good candidates for handling explicit domain knowledge. They define domain theories and provide resources for uniquely identifying concepts of domain knowledge. Therefore, allowing models to make references to ontologies is a modular solution for models to explicitly handle domain knowledge. Overcoming the absence of explicit semantics expression in the modeling languages used to specify systems models will increase the robustness of the designed system models. Indeed, references to the axioms and theorems resulting from the ontologies can be used to strengthen the properties of the designed models. The objective is to offer rigorous mechanisms for handling domain knowledge in design models. We also show how these mechanisms are set up in the cases of formal system models, both for static and dynamic aspects.

7.2.2. Incremental Development of Systems and Algorithms

Joint work with Andriamiarina, Manamiary Bruno, with Neeraj Kumar Singh from IRIT, Toulouse, with Rosemary Monahan, NUI Maynooth, Ireland, and with Zheng Cheng, LINA, Nantes.

The development of distributed algorithms and, more generally, of distributed systems, is a complex, delicate, and challenging process. The approach based on refinement applies a design methodology that starts from the most abstract model and leads, in an incremental way, to a distributed solution. The use of a proof assistant gives a formal guarantee on the conformance of each refinement with the model preceding it.

Our main results during 2016 are:

- An extension [18] for handling the verification of concurrent programs. In a second step, we show the importance of the concept of refinement, and how it can be used to found a methodology for designing concurrent programs using the coordination paradigm.
- A fully mechanized proof [36] of correctness of self-$*$ systems along with an interesting case study related to P2P-based self-healing protocols.
- We report on our progress in implementing a software development environment that integrates two formal software engineering techniques: program refinement as supported by Event-B, and program verification as supported by the Spec# programming system. We improve the usability of formal verification tools by providing a general framework for integrating these two approaches to software verification. We show how the two approaches, based respectively on correctness by construction and on post-hoc verification, can be used in a productive way. In [32], we focus on the final steps in this process where the final concrete specification is transformed into an executable algorithm. We present EB2RC, a plug-in for the RODIN platform that reads in an Event-B model and uses the control framework introduced during its refinement to generate a graphical representation of the executable algorithm. EB2RC also generates a recursive algorithm that is easily translated into executable code. We illustrate our technique through case studies and their analysis.

7.2.3. Verification of the Pastry routing protocol

In her PhD thesis, Noran Azmy develops a formal proof in TLA$^+$ of the routing protocol used in the Pastry protocol [51] for maintaining a distributed hash table over a peer-to-peer network. In a previous thesis [47], Tianxiang Lu had found problems with all published versions of the original protocol, introduced a variant of Pastry, and given a first correctness proof of that protocol, assuming that no node ever disconnects. Due to limitations of TLAPS at that time, Lu's proof relied on many unchecked assumptions on arithmetic and on the underlying data structures, and it was later discovered that several of these assumptions were not valid.

Noran Azmy simplified the proof by introducing intermediate abstractions that allowed her to avoid low-level arithmetic reasoning in the main proof steps, and she proved lemmas corresponding to those assumptions that were actually used in the proof. As a result, she obtained a complete machine-checked proof of Lu's variant of the Pastry protocol, still under the assumption that no node leaves the network. Moreover, a close analysis of the invariant used in her simplified proof revealed that the protocol could be simplified by leaving out the final "lease exchange" protocol. The results were published at ABZ 2016 [22], and an extended article was invited for publication in Science of Computer Programming.

7.2.4. Proof of Determinacy of PharOS

Joint work with Selma Azaiez and Matthieu Lemerre (CEA Saclay), and Damien Doligez (Inria Paris).

As the main contribution of our group to the ADN4SE project funded by PIA, in cooperation with colleagues from CEA LIST, we wrote a high-level TLA$^+$ specification of the real-time operating system PharOS [46] and proved its executions to be deterministic. Roughly speaking, determinacy means that the sequence of local states of any process during a computation does not depend on the order in which processes are scheduled. The proof assumes that no deadlines are missed (which in practice is ensured by schedulability analysis of the particular applications). This property greatly simplifies the analysis and verification of programs that are executed within PharOS. The results were published at ABZ 2016 [21].

7.2.5. Formal Verification of Chains of Security Functions

Joint work with Rémi Badonnel and Abdelkader Lahmadi of the Madynes research group of Inria Nancy.

During his Master's thesis, Nicolas Schnepf studied formal techniques for the automatic verification of chains of security functions in a setting of software-defined networks (SDN). Concretely, he defined an extension of the Pyretic language [44] taking into account the data plane of SDN controllers and implemented a translation of that extension to the input languages of the nuXmv model checker and of SMT solvers. The approach and its scalability was validated over crafted security chains, and a conference paper describing the results is under preparation. Nicolas Schnepf started a PhD thesis in October 2016, jointly supervised by members of the Madynes and VeriDis groups.

7.2.6. Auditing hybrid systems for compliance

There is a huge gap in complexity between the actual analysis of a complex hybrid system and the analysis of the eventual control needed for safe operation. For example, for the combustion process of an engine there is not even a closed formal model, but the eventual control can be represented in a finite domain language. Such a language can then in particular be used for run-time control of a system through an auditor, providing the detection of faulty components or compliance violations. We have studied the consequences of such an approach if applied to the overall life time process of a technical system [29].

CARTE Team

6. New Results

6.1. Quantum Computing

Participants: Simon Perdrix, Quanlong Wang.

- **ZX-calculus**

  The ZX-calculus is a powerful diagrammatic language for quantum mechanics and quantum information processing. The completeness of the ZX-calculus is crucial: the language would be complete if any equation involving two diagrams representing the same quantum evolution can be derived using the rules of the language. While the language is known to be incomplete in general with no obvious way to add some new rules [75], two interesting fragments have been studied: the $\pi/2$ and the $\pi/4$-fragments, obtained by restricting the angles of diagrams to be multiples of $\pi/2$ and $\pi/4$ respectively.

  The $\pi/4$-fragment is approximatively universal for quantum mechanics, i.e. any quantum evolution can be approximated with an arbitrary accuracy using a diagram involving only angles multiple of $\pi/4$. The completeness of this fragment was one of the main open question in this domain. We have proved that this fragment is incomplete. We exhibit a fairly simple equation called supplementarity and we prove that this equation cannot be derived in the ZX-calculus. We propose as a consequence, to add supplementarity to the set of rules of the ZX-calculus. This result has been published at MFCS 2016 [20].

  The $\pi/2$-fragment is not universal, even approximatively. However it corresponds to the so-called stabiliser quantum mechanics, an interesting fragment of quantum mechanics. The $pi/2$-fragment is known to be complete for stabiliser quantum mechanics [33]. We have proved recently that the rules of the language can be simplified, leading to a simpler set of axioms. Moreover we have proved that most of the remaining rules being necessary are the completeness of the $\pi/2$-fragment. This result has been published at QPL 2016 [16].

- **Causal Graph Dynamics**

  Causal Graph Dynamics [30] 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). We add a further physics-like symmetry, namely reversibility. This result has been presented at RC 2016 [15].

6.2. Implicit Computational Complexity

Participants: Emmanuel Hainry, Romain Péchoux.

We have written a full journal paper, accepted in Information and Computation (special issue of DICE 2015), on the complexity analysis of Object Oriented programming languages based on tiered types. The corresponding type system provides sound and complete characterization of the set of polynomial time computable functions. As a consequence, the heap-space and the stack-space requirements of typed programs are also bounded polynomially. This type system is inspired by previous works on Implicit Computational Complexity, using tiering and non-interference techniques. The presented methodology has several advantages. First, it provides explicit big O polynomial upper bounds to the programmer, hence its use could allow the programmer to avoid memory errors. Second, type checking is decidable in polynomial time. Last, it has a good expressivity since it analyzes most object oriented features like inheritance, overload, override and recursion. Moreover it can deal with loops guarded by objects and can also be extended to statements that alter the control flow like break or return.

6.3. Computing with Infinite Objects

Participant: Mathieu Hoyrup.

- **Decidable properties of subrecursive functions**

  We have studied the following problem : given a subrecursive class (like the primitive recursive functions, the polynomial-time computable functions, etc.) and a sound and complete programming language for that class, what are the properties of functions that are decidable (by a Turing machine), given a program for that function in the restricted language? We give a complete characterization of these properties. We show that they can be expressed as unions of elementary properties of being compressible. If $h : \mathbb{N} \to \mathbb{N}$ is a computable increasing unbounded function (like $\log(n)$ or 2^n), we say that a function $f : \mathbb{N} \to \mathbb{N}$ is h-compressible if for each n there is a program (in the restricted language) of size at most $h(n)$ computing a function that coincides with f on entries $0, 1, ..., n$. Whether f is h-compressible is decidable given a program for f, and every decidable property can be obtained as a combination of such elementary properties.

  We also prove that such a characterization does not hold for the whole class of total recursive functions, and leave the problem open for that class.

  The results appears in an article presented at ICALP 2016 [19].

- **Baire category and computability theory**

  Baire category is a very powerful tool in mathematical analysis to prove existence of objects with prescribed properties without having to explicitly build them, but showing instead that the class of objects with these properties is large in some sense. In Computability theory one often builds objects with very specific properties, notably to separate classes, and the proofs are often very involved. We show how Baire category can be adapted in order to be applied to computability theory, to prove existence results without the need of an explicit construction. We review notions that we introduced in the last years and provide new results in an invited paper at CiE 2016 [14].

6.4. Cellular automata as a model of computation

Participant: Nazim Fatès.

The density classification problem is a simple computational problem where a distributed system composed of many cells need to find the majority state in its initial configuration. It is known that no deterministic cellular automaton can solve this problem without making errors. On the other hand, it was shown that a probabilistic mixture of the traffic rule and the majority rule solves the one-dimensional problem correctly with a probability arbitrarily close to one. We investigated the possibility of a similar approach in two dimensions and introduced a companion problem, the particle spacing problem, as an intermediary step. We showed that although this second problem does not have a cellular automaton solution, the use of randomized frameworks, via interacting particle systems, could allow us to have interesting solutions, which were analysed with a theoretical approach and with numerical simulations [18].

In the same direction of research, we studied how to coordinate a team of agents to locate a hidden source on a two-dimensional discrete grid. The challenge here is to find the position of the source with only sporadic detections. This problem arises in various situations, for instance when insects emit pheromones to attract their partners. A search mechanism named infotaxis was previously proposed to explain how agents may progressively approach the source by using only intermittent detections.

We studied the problem of doing a collective infotaxis search with agents that are almost memoryless. We presented a bio-inspired model which mixes stochastic cellular automata and reactive multi-agent systems. The model, inspired by the behaviour of the social amoeba *Dictyostelium discoideum*, relies on the use of reaction-diffusion waves to guide the agents to the source. The random emissions of waves allows the formation of a group of amoebae, which successively act as emitters of waves or listeners, according to their local perceptions. Our worked showed that the model is worth considering and may provide a simple solution to coordinate a team to perform a distributed form of infotaxis [17].

COMETE Project-Team

7. New Results

7.1. Foundations of information hiding

Information hiding refers to the problem of protecting private information while performing certain tasks or interactions, and trying to avoid that an adversary can infer such information. This is one of the main areas of research in Comète; we are exploring several topics, described below.

7.1.1. Axioms for Information Leakage

Quantitative information flow aims to assess and control the leakage of sensitive information by computer systems. A key insight in this area is that no single leakage measure is appropriate in all operational scenarios; as a result, many leakage measures have been proposed, with many different properties. To clarify this complex situation, we studied in [17] information leakage axiomatically, showing important dependencies among different axioms. We also established a completeness result about the g-leakage family, showing that any leakage measure satisfying certain intuitively-reasonable properties can be expressed as a g-leakage.

7.1.2. Up-To Techniques for Generalized Bisimulation Metrics

Bisimulation metrics allow us to compute distances between the behaviors of probabilistic systems. In [18] we presented enhancements of the proof method based on bisimulation metrics, by extending the theory of up-to techniques to (pre)metrics on discrete probabilistic concurrent processes.

Up-to techniques have proved to be a powerful proof method for showing that two systems are bisimilar, since they make it possible to build (and thereby check) smaller relations in bisimulation proofs. We defined soundness conditions for up-to techniques on metrics, and studied compatibility properties that allow us to safely compose up-to techniques with each other. As an example, we derived the soundness of the up-to-bisimilarity-metric-and-context technique.

The study was carried out for a generalized version of the bisimulation metrics, in which the Kantorovich lifting is parametrized with respect to a distance function. The standard bisimulation metrics, as well as metrics aimed at capturing multiplicative properties such as differential privacy, are specific instances of this general definition.

7.1.3. Compositional methods for 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 with the inference of the secret information. In [12] 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 derive a generalization of Chaum's strong anonymity result.

7.1.4. Differential Privacy Models for Location-Based Services

In [13], we considered the adaptation of differential privacy to the context of location-based services (LBSs), which personalize the information provided to a user based on his current position. Assuming that the LBS provider is queried with a perturbed version of the position of the user instead of his exact one, we relied on differential privacy to quantify the level of indistinguishability (i.e., privacy) provided by this perturbation with respect to the user's position. In this setting, the adaptation of differential privacy can lead to various models depending on the precise form of indistinguishability required. We discussed the set of properties that

hold for these models in terms of privacy, utility and also implementation issues. More precisely, we first introduced and analyzed one of these models, the (D,eps)-location privacy, which is directly inspired from the standard differential privacy model. In this context, we described a general probabilistic model for obfuscation mechanisms for the locations whose output domain is the Euclidean space E^2. In this model, we characterized the satisfiability conditions of (D,eps)-location privacy for a particular mechanism and also measured its utility with respect to an arbitrary loss function. Afterwards, we presented and analyzed symmetric mechanisms in which all locations are perturbed in a unified manner through a noise function, focusing in particular on circular noise functions. We proved that, under certain assumptions, the circular functions are rich enough to provide the same privacy and utility levels as other more complex (i.e., non-circular) noise functions, while being easier to implement. Finally, we extended our results to a generalized notion for location privacy, called 'l-privacy' capturing both (D,eps)-location privacy and also the notion of geo-indistinguishability recently introduced by Andrès, Bordenabe, Chatzikokolakis and Palamidessi.

7.1.5. Practical Mechanisms for Location Privacy

The continuously increasing use of location-based services poses an important threat to the privacy of users. A natural defense is to employ an obfuscation mechanism, such as those providing geo-indistinguishability, a framework for obtaining formal privacy guarantees that has become popular in recent years.

Ideally, one would like to employ an optimal obfuscation mechanism, providing the best utility among those satisfying the required privacy level. In theory optimal mechanisms can be constructed via linear programming. In practice, however, this is only feasible for a radically small number of locations. As a consequence, all known applications of geo-indistinguishability simply use noise drawn from a planar Laplace distribution.

In [23] we studied methods for substantially improving the utility of location obfuscation, while having practical applicability as a central constraint. We provided such solutions for both infinite (continuous or discrete) as well as large but finite domains of locations, using a Bayesian remapping procedure as a key ingredient. We evaluated our techniques in two real world complete datasets, without any restriction on the evaluation area, and showed important utility improvements wrt the standard planar Laplace approach.

7.1.6. Preserving differential privacy under finite-precision semantics

The approximation introduced by finite-precision representation of continuous data can induce arbitrarily large information leaks even when the computation using exact semantics is secure. Such leakage can thus undermine design efforts aimed at protecting sensitive information. In [14] we focussed on differential privacy, an approach to privacy that emerged from the area of statistical databases and is now widely applied also in other domains. In this approach, privacy is protected by adding noise to the values correlated to the private data. The typical mechanisms used to achieve differential privacy have been proved correct in the ideal case in which computations are made using infinite-precision semantics. We analyzed the situation at the implementation level, where the semantics is necessarily limited by finite precision, i.e., the representation of real numbers and the operations on them are rounded according to some level of precision. We showed that in general there are violations of the differential privacy property, and we studied the conditions under which we can still guarantee a limited (but, arguably, acceptable) variant of the property, under only a minor degradation of the privacy level. Finally, we illustrated our results on two examples: the standard Laplacian mechanism commonly used in differential privacy, and a bivariate version of it recently introduced in the setting of privacy-aware geolocation.

7.1.7. Quantifying Leakage in the Presence of Unreliable Sources of Information

Belief and min-entropy leakage are two well-known approaches to quantify information flow in security systems. Both concepts stand as alternatives to the traditional approaches founded on Shannon entropy and mutual information , which were shown to provide inadequate security guarantees. In [16] we unified the two concepts in one model so as to cope with the frequent (potentially inaccurate, misleading or outdated) attackers' side information about individuals on social networks, online forums, blogs and other forms of online communication and information sharing. To this end we proposed a new metric based on min-entropy that takes into account the adversary's beliefs.

7.1.8. On the Compositionality of Quantitative Information Flow

In the min-entropy approach to quantitative information flow, the leakage is defined in terms of a minimization problem, which, in the case of large systems, can be computationally rather heavy. The same happens for the recently proposed generalization called g-vulnerability. In [25] 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 is the derivation of bounds on the g-leakage of the whole system in terms of the g-leakages of its components. We also considered the particular cases of min-entropy leakage and of parallel channels, generalizing and systematizing results from the literature. We demonstrated the effectiveness of our method and evaluate the precision of our bounds using examples.

7.2. Foundations of Concurrency

Distributed systems have changed substantially in the recent past with the advent of phenomena like social networks and cloud computing. In the previous incarnation of distributed computing the emphasis was on consistency, fault tolerance, resource management and related topics; these were all characterized by *interaction between processes*. Research proceeded along two lines: the algorithmic side which dominated the Principles Of Distributed Computing conferences and the more process algebraic approach epitomized by CONCUR where the emphasis was on developing compositional reasoning principles. What marks the new era of distributed systems is an emphasis on managing access to information to a much greater degree than before.

7.2.1. Belief, Knowledge, Lies and Other Utterances in an Algebra for Space and Extrusion

Spatial constraint systems are algebraic structures from concurrent constraint programming to specify spatial and epistemic behavior in multi-agent system. In [15], [11] we developed the theory of spatial constraint systems with operators to specify information and processes moving from a space to another. We investigated the properties of this new family of constraint systems and illustrated their applications. From a computational point of view the new operators provide for process/information extrusion, a central concept in formalisms for mobile communication. From an epistemic point of view extrusion corresponds to a notion we called utterance; a piece of information that an agent communicates to others but that may be inconsistent with the agent's beliefs. Utterances can then be used to express instances of epistemic notions such as hoaxes or intentional lies. Spatial constraint system can express the epistemic notion of belief by means of space functions that specify local information. We showed that spatial constraint can also express the epistemic notion of knowledge by means of a derived spatial operator that specifies global information. In [21] we reported on our progress using spatial constraint system as an abstract representation of modal and epistemic behaviour.

7.2.2. Deriving Inverse Operators for Modal Logic

In [20] we used spatial constraint systems to give an abstract characterization of the notion of normality in modal logic and to derive right inverse/reverse operators for modal languages. In particular, we identified the weakest condition for the existence of right inverses and showed that the abstract notion of normality corresponds to the preservation of finite suprema. We applied our results to existing modal languages such as the weakest normal modal logic, Hennessy-Milner logic, and linear-time temporal logic. We also discussed our results in the context of modal concepts such as bisimilarity and inconsistency invariance.

7.2.3. D-SPACES: Implementing Declarative Semantics for Spatially Structured Information

In [22] we introduced D-SPACES, an implementation of constraint systems with space and extrusion operators. D-SPACES is coded as a c++11 library providing implementations for constraint systems, space functions and extrusion functions. D-SPACES provides property-checking methods as well as an implementation of a specific type of constraint systems (boolean algebras). We illustrated the implementation with a small social network where users post their beliefs and utter their opinions.

7.2.4. Slicing Concurrent Constraint Programs

Concurrent Constraint Programming (CCP) is a declarative model for concurrency where agents interact by telling and asking constraints (pieces of information) in a shared store. Some previous works have developed (approximated) declarative debuggers for CCP languages. However, the task of debugging concurrent programs remains difficult. In [19] we defined a dynamic slicer for CCP and we showed it to be a useful companion tool for the existing debugging techniques. Our technique starts by considering a partial computation (a trace) that shows the presence of bugs. Often, the quantity of information in such a trace is overwhelming, and the user gets easily lost, since she cannot focus on the sources of the bugs. Our slicer allows for marking part of the state of the computation and assists the user to eliminate most of the redundant information in order to highlight the errors. We showed that this technique can be tailored to timed variants of CCP. We also developed a prototypical implementation freely available for making experiments.

DICE Team

6. New Results

6.1. Intermediation platforms

Our study of the geopolitics of intermediation aims at grasping the balance of power between platforms, as well as between states - in their relation to platforms - and between platforms and states. We have extended our studies with insights from law in [1] and economics in [2]. We have tuned the metrics we already had in order to better grasp the economic weight of intermediation economy. As we did so, we improved our understanding of the social weight of intermediation platforms and the legal issues which they raise.

Our focus has turned to the analysis of public and private policies and their relation to the development of intermediation platforms. In [3], we study a set of cases ranging from the Safe Harbour to the right to be forgotten. Using the "coalition framework" as our analysis framework, we identify the actors influencing policy-making and potential reasons for the success of failure of policies. Such failures include forbidding innovation or preventing public bodies from stepping up their digital capabilities.

Our work has been intrinsically interdisciplinary, the main result of our work is a global modal of intermediation platforms and their economy, presented in [6]. This model helps to understand the current issues raised by the ubderization for instance.

6.2. Development of platforms

Dice team designs software architectures for intermediation platforms. C3PO and BitBallot targets spontaneous and ephemeral social networks whereas Jumplyn focuses on pure central based system. All these architectures share a common JavaScript layout both at the client and the server sides. In the research context we validate state-of-the art technologies promoted by web leaders such as Google AngularJS, Facebook ReactJS and many others such as Netflix, Wallmart, or the Linux foundation for node.js. The Web environment raises many big issues since all equipments are basically connected to the Internet and the balance between end-user equipment cost and processing power is still a work in progress. Our main research track in such context is to find proper software toolkits hiding Web complexity. We mainly focus on time jitter, cornerstone of Web development, since it implies both end-user and network TCP indecisions. Due to this jitter combination the Web programming model has mutated toward the promises paradigm. It is a complex event based development model provided without external API help. It handles future execution whether successful or not, in a time jittered context. AngularJS, ReactJS, CoffeeScript, NodeJS, MongoDB, ElasticSearch are all time jitter compliant technologies designed for the Web constrains and revolutionising the way we build intermediation platforms.

In C3PO, we tested application in real conditions during the marathon of Vannes and the semi-marathon of Beaune. A few hundred users have downloaded and use the application. The returns on this one are rather positive. [4].

Our joint work with Worldline explores the promises paradigm model to enable automation extraction of independent micro-service. These micro-services called fluxion [9], from the contraction of flow and functions, may be dynamically and transparently moved over a cluster of servers. Our novelty resides in the fact that the original code is not redesigned for the cluster architecture. Fluxion are extracted from the initial code, and an equivalence is maintained between the initially promissified code and the fluxionized one. Code has two facets, a promise one, used to express software services and a fluxion one, used to express software bottlenecks [5].

Eventually our work with Jumplyn explores complex centralised social network. We want to design a software system to later support our technical research hot topics. The target theme is a software platform that helps students handle their projects. University depends more and more on external resources to teach students. Theses resources are both known by students and their teachers, and the pace and range of explored technologies leads to difficulties in teaching state-of-the-art subjects. The more dedicated a professor needs to be in his research activity, the more broad knowledge he has to teach. For instance 20 years ago one could cope software development teaching with one or two programming languages. Nowadays, a single code involves more then four programming languages to be fully understood. This technology spreading issue stands still in many teaching domains, since past technologies are still actives and future one are promising. We build Jumplyn to cope with this unbalanced game. To help student improving their project and avoid working with obsolete technologies, and to help teacher face the universal and inexpensive availability of knowledge. Jumplyn is a complex JavaScript development stack that collects resources for improving student work and providing services to help them from day to day activities. The current stack integrates the following technologies : MaterialDesign, AngularJS, CoffeeScript, NodeJs, MongoDb, ElasticSearch. Managing and developing software service above this stack is a complex research issue for a small sized development team. We do not have any publication on Jumplyn since our first goal is to build a support intermediation platform to study classical issues such as recommendation or web crawling, scraping and indexation with our own sources of raw data.

PESTO Project-Team

7. New Results

7.1. Modelling

7.1.1. New protocol and adversary models

Participants: Jannik Dreier, Steve Kremer.

Isolated Execution Environments (IEEs), such as ARM TrustZone and Intel SGX, offer the possibility to execute sensitive code in isolation from other malicious programs, running on the same machine, or a potentially corrupted OS. A key feature of IEEs is the ability to produce reports binding cryptographically a message to the program that produced it, typically ensuring that this message is the result of the given program running on an IEE. In collaboration with Jacomme (ENS Cachan) and Scerri (Univ. Bristol), Kremer presented a symbolic model for specifying and verifying applications that make use of such features. For this they introduced the SℓAPiC process calculus to reason about reports issued at given locations. They also provide tool support, extending the SAPIC/TAMARIN toolchain and demonstrate the applicability of their framework on several examples implementing secure outsourced computation (SOC), a secure licensing protocol and a one-time password protocol that all rely on such IEEs. This work has been accepted for publication at EuroS&P'17 [27].

Most security properties are modelled as *safety* properties (*"bad things do not happen"*). Another important class of properties is that of *liveness* properties (*"eventually, good things happen"*). Reasoning about the class of *liveness* properties of cryptographic protocols, has received little attention in the literature, even though this class is vital in many security-sensitive applications, such as fair exchange protocols, or security layers in industrial control systems. In collaboration with Backes and Künnemann (U. Saarland, Germany), Dreier and Kremer have designed a protocol and adversary model that are suitable for reasoning about liveness properties. Tool support is also provided by extending the *SAPIC/TAMARIN* tool chain and several case studies demonstrate the effectiveness of the approach. This work has been accepted for publication at EuroS&P'17 [20].

7.1.2. New properties

Participants: Véronique Cortier, Jannik Dreier.

Defining security properties correctly is often a challenging problem on its own: too strict definitions may lack generality and exclude systems that should be considered as secure, while relaxing definitions may lead to accepting insecure systems.

In e-voting, *verifiability* is the property meant to defend against voting devices and servers that have programming errors or are outright malicious. While the first formal definitions of verifiability were devised in the late 1980s already, new verifiability definitions are still being proposed. The definitions differ in various aspects, including the classes of protocols they capture and even their formulations of the very core of the meaning of verifiability. This is an unsatisfying state of affairs, leaving the research on the verifiability of e-voting protocols and systems in a fuzzy state. Cortier, in collaboration with Galindo (U. Birmingham, UK), Küsters, Müller (U. Trier, Germany) and Truderung (Polyas GmbH, Germany), review all formal definitions of verifiability proposed in the literature and cast them in a framework proposed by the KTV framework, yielding a uniform treatment of verifiability. This enables a detailed comparison of the various definitions of verifiability from the literature and a discussion of advantages and disadvantages, limitations and problems. Finally, a general definition of verifiability is distilled, which can be instantiated in various ways. This work has been presented at S&P'16 [26].

Industrial systems are nowadays regularly the target of cyberattacks, the most famous being Stuxnet. At the same time such systems are increasingly interconnected with other systems and insecure media such as Internet. In contrast to other IT systems, industrial systems often do not only require classical properties like data confidentiality or authentication of the communication, but have special needs due to their interaction with the physical world. For example, the reordering or deletion of some commands sent to a machine can cause the system to enter an unsafe state with potentially catastrophic effects. To prevent such attacks, the integrity of the message flow is necessary.

In joint work with Lafourcade (Université Clermont-Ferrand), Potet, and Puys (University Grenoble Alpes), Dreier developed a formal definition of Flow Integrity in the context of industrial systems. The framework is applied to two well-known industrial protocols: OPC-UA and MODBUS. Using *TAMARIN*, a cryptographic protocol verification tool, they identified several design flaws in some of the different versions of these protocols. We also discussed how to efficiently model counters and timestamps in *TAMARIN*, as they are key ingredients of the analyzed protocols. This work is currently under submission.

7.2. Analysis

7.2.1. Analysis of equivalence properties

Participants: Vincent Cheval, Véronique Cortier, Antoine Dallon, Ivan Gazeau, Steve Kremer, Christophe Ringeissen.

Automatic tools based on symbolic models have been successful in analyzing security protocols. These tools are particularly well adapted for trace properties (e.g. secrecy or authentication). However, they often fail to analyse equivalence properties. Equivalence properties can express a variety of security properties, including in particular privacy properties (vote privacy, anonymity, untraceability). Several decision procedures have already been proposed but the resulting tools are often rather limited, and lack efficiency.

In the case of a passive adversary, Ringeissen, in collaboration with Marshall (U. of Mary Washington, USA) and Erbatur (LMU, Germany) present new combination techniques for the study of deducibility and static equivalence in unions of equational theories sharing constructors. This allows us to develop new modularity results for the decidability of deducibility and static equivalence. In turn, this should allow for the security analysis of protocols which previous disjoint combination methods could not address because their axiomatization corresponds to the union of non-disjoint equational theories.

In case of an active adversary, and a bounded number of sessions, we made several advances. In [14], Cheval and Kremer, in collaboration with Chadha (U. of Missouri, USA) and Ciobâcă (U. Iasi, Romania), present the theory underlying the *Akiss* tool, a Horn clause resolution based procedure for both under- and over-approximating trace equivalence. They show partial correctness for a large class of cryptographic primitives, modelled as an arbitrary convergent equational theory that has the finite variant properties. Additionally, termination is shown for subterm convergent theories. Gazeau and Kremer, in collaboration with Baelde (LSV, ENS Cachan) and Delaune (IRISA) have extended the *Akiss* tool with support for exclusive or. They analyse unlinkability in several RFID protocols and resistance to guessing attacks of several password base protocols. Cortier and Dallon, in collaboration with Delaune (IRISA) propose a novel algorithm, based on graph planning and SAT-solving, which significantly improves the efficiency of the analysis of equivalence properties. The resulting implementation, SAT-Equiv, can analyze several sessions where most tools have to stop after one or two sessions. Finally, Cheval and Kremer propose a novel decision procedure for verifying trace equivalence. Unlike most existing tools, they support a rich class of cryptographic primitives and protocols that may use else branches. An implementation of the procedure is currently under development.

These results are currently under submission.

7.2.2. Simplification results

Participants: Véronique Cortier, Antoine Dallon, Steve Kremer.

Bounding the number of agent identies is a current practice when modeling a protocol. In 2003, it has been shown that one honest agent and one dishonest agent are indeed sufficient to find all possible attacks, for trace properties. This is no longer the case for equivalence properties, crucial to express many properties such as vote privacy or untraceability. As a first result of his PhD, Antoine Dallon has shown that it is sufficient to consider two honest agents and two dishonest agents for equivalence properties, for deterministic processes with standard primitives and without else branches. More generally, we show how to bound the number of agents for arbitrary constructor theories and for protocols with simple else branches. We show that our hypotheses are tight, providing counter-examples for non action-deterministic processes, non constructor theories, or protocols with complex else branches. This work has been presented at POST 2016 [24] and obtained the EASST best paper award of the ETAPS conference.

When verifying e-voting protocols, one of the difficulties is that they need to be secure for an arbitrary number of malicious voters. In collaboration with Arapinis (U. Edinburgh, UK), Cortier and Kremer identify a class of voting protocols for which only a small number of voters needs to be considered: if there is an attack on vote privacy, for an arbitrary number of honest and dishonest voters, then there is also an attack that involves at most 3 voters (2 honest voters and 1 dishonest voter). In the case where the protocol allows a voter to cast several votes and counts, e.g., only the last one, we also reduce the number of ballots required for an attack to 10, and under some additional hypotheses, 7 ballots. They illustrate the applicability of our results on several case studies, including different versions of Helios and Prêt-à-Voter, as well as the JCJ protocol. For some of these protocols the ProVerif tool is used to provide the first formal proofs of privacy for an unbounded number of voters. This work has been presented at ESORICS 2016 [19].

7.2.3. Analysis of stateful security protocols

Participants: Jannik Dreier, Charles Duménil, Steve Kremer.

In collaboration with Künnemann (U. Saarland, Germany), Kremer proposes *SAPIC* (stateful applied pi calculus), a process calculus with constructs for manipulation of a global state by processes running in parallel. They show that this language can be translated to multiset rewriting 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. The tool is applied to several case studies among which a simplified fragment of PKCS#11, the Yubikey security token, and an optimistic contract signing protocol. This work has been published in the Journal of Computer Security [15]. Dreier, Duménil and Kremer, in collaboration with Sasse (ETH Zurich, Switzerland) improve the underlying theory and the *TAMARIN* tool to allow for more general user-specified equational theories: the extension supports arbitrary convergent equational theories that have the finite variant property, making *TAMARIN* the first tool to support at the same time this large set of user-defined equational theories, protocols with global mutable state, an unbounded number of sessions, and complex security properties. The effectiveness of this generalization is demonstrated by analyzing several protocols that rely on blind signatures, trapdoor commitment schemes, and ciphertext prefixes that were previously out of scope. This work has been accepted for publication at POST'17.

7.2.4. Analysis of e-voting protocols

Participants: Véronique Cortier, Constantin-Catalin Dragan.

Cortier and Dragan provide the first machine-checked proof of privacy-related properties (including ballot privacy) for an electronic voting protocol in the computational model. They target the popular Helios family of voting protocols, for which they identify appropriate levels of abstractions to allow the simplification and convenient reuse of proof steps across many variations of the voting scheme. The resulting framework enables machine-checked security proofs for several hundred variants of Helios and should serve as a stepping stone for the analysis of further variations of the scheme.

In addition, they highlight some of the lessons learned regarding the gap between pen-and-paper and machine-checked proofs, and report on the experience with formalizing the security of protocols at this scale. This work is submitted for publication.

7.2.5. Analysis of Electrum Bitcoin wallet

Participants: Michaël Rusinowitch, Mathieu Turuani.

Electrum is a popular Bitcoin wallet. We introduce a formal modeling in ASLan++ of the two-factor authentication protocol used by the Electrum Bitcoin wallet. This allows us to perform an automatic analysis of the wallet and show that it is secure for standard scenarios in the Dolev Yao model [30]. The result could be derived thanks to some advanced features of the Cl-Atse protocol analyzer such as the possibility to specify i) new intruder deduction rules with clauses and ii) non-deducibility constraints.

7.2.6. Satisfiability Modulo Bridging Theories

Participant: Christophe Ringeissen.

Bridging theories are equational theories defining recursive functions. They are useful to handle equational theories of interest in protocol analysis, as advocated in [48], where a locality approach is promoted to solve the satisfiability problem. In collaboration with Pascal Fontaine (Veridis project-team) and Paula Chocron (IIIA-CSIC Barcelona), we investigate a combination approach for the satisfiability problem modulo this particular non-disjoint union of theories, where a source theory is connected to a target one through a bridging function. In 2016, we have prepared a new full paper unifying previous results presented respectively at CADE 2015 [4] and FroCoS 2015. In that papers, we focused on source theories admitting term-generated models. In [21], we have also explored an extension to deal with terms modulo a congruence relation. This joint work with Raphaël Berthon (ENS Rennes) allows us to consider not only trees but also data structure theories such as lists, multisets and sets.

7.2.7. Analysis of Security Properties for an Unbounded Number of Sessions

Participants: Jonathan Proietto-Stallone, Mathieu Turuani, Laurent Vigneron.

The internship of Jonathan Proietto-Stallone has permitted to study the method described in [37] for analyzing protocols without bounding the number of sessions. We have clarified the formalization of this method, including the consideration of xor and exp operators, and implemented it in *CL-AtSe*.

7.3. Design

7.3.1. E-voting protocols

Participants: Véronique Cortier, Steve Kremer, Peter Roenne.

We propose a new voting scheme, BeleniosRF, that offers both receipt-freeness and end-to-end verifiability. It is receipt-free in a strong sense, meaning that even dishonest voters cannot prove how they voted. We provide a game-based definition of receipt-freeness for voting protocols with non-interactive ballot casting, which we name strong receipt-freeness (sRF). To our knowledge, sRF is the first game-based definition of receipt-freeness in the literature, and it has the merit of being particularly concise and simple. Built upon the Helios protocol, BeleniosRF inherits its simplicity and does not require any anti-coercion strategy from the voters. We implement BeleniosRF and show its feasibility on a number of platforms, including desktop computers and smartphones. This work has been presented at CCS 2016 [26].

Another challenging problem in e-voting is to provide guarantees when the voting platform itself is corrupted. Du-Vote [45] is a recently presented remote electronic voting scheme that aims to be malware tolerant, i.e., provide security even in the case where the platform used for voting has been compromised by dedicated malware. For this it uses an additional hardware token, similar to tokens distributed in the context of online banking. Du-Vote aims at providing vote privacy as long as either the vote platform or the vote server is honest. For verifiability, the security guarantees are even higher, as even if the token's software has been changed, and the platform and the server are colluding, attempts to change the election outcome should be detected with high probability. We provide an extensive security analysis of Du-Vote and show several attacks on both privacy as well as verifiability. We also propose changes to the system that would avoid many of these attacks. This work has been presented at Euro S&P 2016 [28].

7.3.2. Designing and proving an EMV-compliant payment protocol for mobile devices

Participants: Véronique Cortier, Alicia Filipiak.

In collaboration with Gharout, Traoré and Florent (Orange Labs), we devised a payment protocol that can be securely used on mobile devices, even infected by malicious applications. Our protocol only requires a light use of Secure Elements, which significantly simplifies certification procedures and protocol maintenance. It is also fully compatible with the EMV-SDA protocol and allows off-line payments for the users. We provide a formal model and full security proofs of the protocol using the TAMARIN prover. This work has been accepted for publication at Euro S&P'17 [25].

7.3.3. Composition and design of PKIs

Participants: Vincent Cheval, Véronique Cortier.

Public Key Infrastructures (PKIs) is the backbone of public key cryptography, as it ensures that public keys can be correctly linked to identities. Their security typically relies on honest Certificate Authorities that distribute and/or generate keys to all parties. This trust assumption is a vulnerability exploited in numerous attacks. Recent proposals using public logs have succeeded in making certificate management more transparent and verifiable. However, those proposals involve a fixed set of authorities which means an oligopoly is created. Another problem with current log-based system is their heavy reliance on trusted parties that monitor the logs. Cheval, in collaboration with Ryan and Yu (U. Birmingham, UK) propose a distributed transparent key infrastructure (DTKI), which greatly reduces the oligopoly of service providers and allows verification of the behaviour of trusted parties. Their work also formalises the public log data structure and provides a formal analysis of the security that DTKI guarantees. The work has been published in The Computer Journal [17].

In protocol analysis one makes the (strong) assumption that honestly generated keys are available to all parties and that the link between identities and public keys is fixed and known to everyone. The abstraction is grounded in solid intuition but there are currently no theoretical underpinnings to justify its use. Cheval and Cortier, in collaboration with Warinschi (U. Bristol, UK), initiate a rigorous study of how to use PKIs within other protocols, securely. They first show that the abstraction outlined above is in general unsound by exhibiting a simple protocol which is secure with idealized key distribution but fails in the presence of more realistic PKI instantiation. Their main result is a generic composition theorem that identifies under which conditions protocols that require public keys can safely use any PKI protocol (which satisfies a security notion which we identify). Interestingly, unlike most existing composition results in symbolic models they do not require full tagging of the composed protocols. Furthermore, the results confirm the recommended practice that keys used in the PKI should not be used for any other cryptographic task. This work is currently under submission.

7.3.4. Physical Zero-Knowledge Proofs

Participant: Jannik Dreier.

In this work we develop physical algorithms to realize zero-knowledge proofs for Akari, Takuzu, Kakuro, and KenKen, which are logic games similar to Sudoku. The zero-knowledge proofs allow a player to show that he knows a solution without revealing it. These interactive proofs can be realized with simple office material as they only rely on cards and envelopes. They can thus be used for example for scientific outreach activities, or in teaching. Moreover, we also formalized our algorithms and proved their security. This joint work with Bultel (U. Clermont-Ferrand), Dumas (U. Grenoble Alpes), and Lafourcade (U. Clermont-Ferrand) was published at FUN 2016 [22].

7.3.5. Privacy Protection in Social Networks

Participants: Younes Abid, Abdessamad Imine, Huu Hiep Nguyen, Clément Pascutto, Michaël Rusinowitch, Laura Trivino.

Hiep Nguyen's PhD thesis addresses three privacy problems of social networks: graph anonymization, private community detection and private link exchange. The main goal is to provide new paradigms for publication of social graphs in noisy forms, private community detection over graphs as well as distributed aggregation of graphs via noisy link exchange processes. The graph anonymization problem is solved via two different semantics: uncertainty semantics and differential privacy. For uncertainty semantics, a general obfuscation model is proposed that keeps the expected node degree equal to those in the unanonymized graph. Over the last decade, a great number of algorithms for community detection have been proposed to deal with the increasingly complex networks. However, the problem of doing this in a private manner is rarely considered. We analyze the major challenges behind the problem and propose several schemes to tackle them under differential privacy from two perspectives: input perturbation and algorithm perturbation [29].

We address the problem of rapidly disclosing many friendship links using only legitimate queries (i.e., queries and tools provided by the targeted social network). Our study [18] sheds new light on the intrinsic relation between communities (usually represented as groups) and friendships between individuals. To develop an efficient attack we analysed group distributions, densities and visibility parameters from a large sample of a social network. By effectively exploring the target group network, our proposed algorithm is able to perform friendship and mutual-friend attacks along a strategy that minimizes the number of queries. Pascutto has established a state-of-the-art on inference techniques for social networks. Trivino has developed a user interface for privacy risk evaluation on social networks.

PRIVATICS Project-Team

6. New Results

6.1. MobileAppScrutinator: A Simple yet Efficient Dynamic Analysis Approach for Detecting Privacy Leaks across Mobile OSs

Participants: Jagdish Achara, Vincent Roca, Claude Castelluccia.

Smartphones, the devices we carry everywhere with us, are being heavily tracked and have undoubtedly become a major threat to our privacy. As " Tracking the trackers " has become a necessity, various static and dynamic analysis tools have been developed in the past. However, today, we still lack suitable tools to detect, measure and compare the ongoing tracking across mobile OSs. To this end, we propose MobileAppScrutinator [24], based on a simple yet efficient dynamic analysis approach, that works on both Android and iOS (the two most popular OSs today). To demonstrate the current trend in tracking, we select 140 most representative Apps available on both Android and iOS AppStores and test them with MobileAppScrutinator. In fact, choosing the same set of apps on both Android and iOS also enables us to compare the ongoing tracking on these two OSs. Finally, we also discuss the effectiveness of privacy safeguards available on Android and iOS. We show that neither Android nor iOS privacy safeguards in their present state are completely satisfying.

6.2. MyTrackingChoices: Pacifying the Ad-Block War by Enforcing User Privacy Preferences

Participants: Jagdish Achara, Claude Castelluccia.

Free content and services on the Web are often supported by ads. However, with the proliferation of intrusive and privacy-invasive ads, a significant proportion of users have started to use ad blockers. As existing ad blockers are radical (they block all ads) and are not designed taking into account their economic impact, ad-based economic model of the Web is in danger today. In this paper, we target privacy-sensitive users and provide them with fine-grained control over tracking. Our working assumption is that some categories of web pages (for example, related to health, religion, etc.) are more privacy-sensitive to users than others (education, science, etc.). Therefore, our proposed approach consists in providing users with an option to specify the categories of web pages that are privacy-sensitive to them and block trackers present on such web pages only. As tracking is prevented by blocking network connections of third-party domains, we avoid not only tracking but also third-party ads. Since users will continue receiving ads on web pages belonging to non-sensitive categories, our approach essentially provides a trade-off between privacy and economy. To test the viability of our solution, we implemented it as a Google Chrome extension, named MyTrackingChoices (available on Chrome Web Store). Our real-world experiments with MyTrackingChoices [23] show that the economic impact of ad blocking exerted by privacy-sensitive users can be significantly reduced.

6.3. Security or privacy?

Participants: Amrit Kumar, Cédric Lauradoux.

Security softwares such as anti-viruses, IDS or filters help Internet users to protect their privacy from hackers. The cost of this protection is that the users privacy is stripped away by the companies providing these security solutions. Currently, Internet users can choose between no security with the risk of being hacked or use security software and lose all personal data to security companies. As a example of this dilemma, we analyze the solution proposed by Google and Yandex for Safe Browsing [8] and shows that their privacy policies do not match the reality: Google can perform tracking.

6.4. Near-Optimal Fingerprinting with Constraints

Participants: Gabor Gulyas, Gergely Acs, Claude Castelluccia.

Several recent studies have demonstrated that people show large behavioural uniqueness. This has serious privacy implications as most individuals become increasingly re-identifiable in large datasets or can be tracked while they are browsing the web using only a couple of their attributes, called as their fingerprints. Often, the success of these attacks depend on explicit constraints on the number of attributes learnable about individuals, i.e., the size of their fingerprints. These constraints can be budget as well as technical constraints imposed by the data holder. For instance, Apple restricts the number of applications that can be called by another application on iOS in order to mitigate the potential privacy threats of leaking the list of installed applications on a device. In [15], we address the problem of identifying the attributes (e.g., smartphone applications) that can serve as a fingerprint of users given constraints on the size of the fingerprint. We give the best fingerprinting algorithms in general, and evaluate their effectiveness on several real-world datasets. Our results show that current privacy guards limiting the number of attributes that can be queried about individuals is insufficient to mitigate their potential privacy risks in many practical cases.

6.5. Data anonymization Evaluation

Participants: Claude Castelluccia, Gergely Acs, Daniel Le Metayer.

Anonymization is a critical issue because data protection regulations such as the European Directive 95/46/EC and the European General Data Protection Regulation (GDPR) explicitly exclude from their scope anonymous information" and personal data rendered anonymous"1. However, turning this general statement into effective criteria is not an easy task. In order to facilitate the implementation of this provision, the Working Party 29 (WP29) has published in April 2014 an Opinion on Anonymization Techniques. This Opinion puts forward three criteria corresponding to three risks called respectively "singling out", "linkability" and "inference". In this work, we first evaluated these criteria and showed that they are neither necessary nor effective to decide upon the robustness of an anonymization algorithm. Then we proposed an alternative approach relying on the notions of acceptable versus unacceptable inferences in [4] and we introduced differential testing, a practical way to implement this approach using machine learning techniques.

6.6. Wi-Fi and privacy

Participants: Mathieu Cunche, Celestin Matte.

- **Geolocation spoofing attack** We present several novel techniques to track (unassociated) mobile devices by abusing features of the Wi-Fi standard. This shows that using random MAC addresses, on its own, does not guarantee privacy. First, we show that information elements in probe requests can be used to fingerprint devices. We then combine these fingerprints with incremental sequence numbers, to create a tracking algorithm that does not rely on unique identifiers such as MAC addresses. Based on real-world datasets, we demonstrate that our algorithm can correctly track as much as 50% of devices for at least 20 minutes. We also show that commodity Wi-Fi devices use predictable scrambler seeds. These can be used to improve the performance of our tracking algorithm. Finally, we present two attacks that reveal the real MAC address of a device, even if MAC address randomization is used. In the first one, we create fake hotspots to induce clients to connect using their real MAC address. The second technique relies on the new 802.11u standard, commonly referred to as Hotspot 2.0, where we show that Linux and Windows send Access Network Query Protocol (ANQP) requests using their real MAC address.
- **Extraction of semantical information from Wi-Fi network identifiers** MAC address randomization in Wi-Fi-enabled devices has recently been adopted to prevent passive tracking of mobile devices. However, Wi-Fi frames still contain fields that can be used to fingerprint devices and potentially allow tracking. Panoptiphone is a tool inspired by the web browser fingerprinting tool Panopticlick, which aims to show the identifying information that can be found in the frames broadcast by

a Wi-Fi-enabled device. Information is passively collected from devices that have their Wi-Fi interface enabled, even if they are not connected to an access point. Panoptiphone uses this information to create a fingerprint of the device and empirically evaluate its uniqueness among a database of fingerprints. The user is then shown how much identifying information its device is leaking through Wi-Fi and how unique it is.

6.7. Formal and legal issues of privacy

Participant: Daniel Le Metayer.

- **Privacy by design** Based on our previous work on the use of formal methods to reason about privacy properties of system architectures, we have proposed a logic to reason about properties of architectures including group authentication functionalities. By group authentication, we mean that a user can authenticate on behalf of a group of users, thereby keeping a form of anonymity within this set. Then we show that this extended framework can be used to reason about privacy properties of a biometric system in which users are authenticated through the use of group signatures.
- **Privacy Risk Analysis** Privacy Impact Assessments (PIA) are recognized as a key step to enhance privacy protection in new IT products and services. They will be required for certain types of products in Europe when the future General Data Protection Regulation becomes effective. From a technical perspective, the core of a PIA is a privacy risk analysis (PRA), which has so far received relatively less attention than organizational and legal aspects of PIAs. We have proposed a rigorous and systematic methodology for conducting a PRA and illustrated it with a quantified-self use-case.

  The smart grid initiative promises better home energy management. However, there is a growing concern that utility providers collect, through smart meters, highly granular energy consumption data that can reveal a lot about the consumer's personal life. This exposes consumers to a large number of privacy harms, of various degrees of severity and likelihood: surveillance by the government and law-enforcement bodies, various forms of discrimination etc. A privacy impact assessment is vital for early identification of potential privacy breaches caused by an IT product or service and for choosing the most appropriate protection measures. So, a data protection impact assessment (DPIA) template for smart grids has been developed by the Expert Group 2 (EG2) of the European Commission's Smart Grid Task Force (SGTF). To carry out a true privacy risk analysis and go beyond a traditional security analysis, it is essential to distinguish the notions of feared events and their impacts, called "privacy harms" here, and to establish a link between them. The Working Party 29 highlights the importance of this link in its feedback on EG2's DPIA. We have provided in [11] a clear relationship among harms, feared events, privacy weaknesses and risk sources and described their use in the analysis of smart grid systems.

  Although both privacy by design and privacy risk analysis have received the attention of researchers and privacy practitioners during the last decade, to the best of our knowledge, no method has been documented yet to establish a clear connection between these two closely related notions. We have proposed a methodology to help designers select suitable architectures based on an incremental privacy risk analysis. The analysis proceeds in three broad phases: 1) a generic privacy risk analysis phase depending only on the specifications of the system and yielding generic harm trees; 2) an architecture-based privacy risk analysis that takes into account the definitions of the possible architectures of the system and yields architecture-specific harm trees by refining the generic harm trees and 3) a context-based privacy risk analysis that takes into account the context of deployment of the system (e.g., a casino, an office cafeteria, a school) and further refines the architecture-specific harm trees to yield context-specific harm trees which can be used to take decisions about the most suitable architectures. To illustrate our approach, we have considered the design of a biometric access control system. Such systems are now used commonly in many contexts such as border security controls, work premises, casinos, airports, chemical plants, hospitals, schools, etc. However, the collection, storage and processing of biometric data raise complex privacy issues. To deal with these privacy problems in biometric access control, a wide array of dedicated techniques (such as

secure sketches or fuzzy vaults) as well as adaptations of general privacy preserving techniques (such as encryption, homomorphic encryption, secure multi-party computation) have been proposed. However, each technique solves specific privacy problems and is suitable in specific contexts. Therefore, it is useful to provide guidance to system designers and help them select a solution and justify it with respect to privacy risks. We have used as an illustration of context a deployment in casinos. The verification of the identities of casino customers is required by certain laws (to prevent access by minors or individuals on blacklists) which can justify the implementation of a biometric access control system to speed up the verification process.

6.8. Building blocks

Participants: Marine Minier, Vincent Roca.

- **Symmetric cryptography**

  In [7], we introduce Constraint Programming (CP) models to solve a cryptanalytic problem: the chosen key differential attack against the standard block cipher AES. The problem is solved in two steps: In Step 1, bytes are abstracted by binary values; In Step 2, byte values are searched. We introduce two CP models for Step 1: Model 1 is derived from AES rules in a straightforward way; Model 2 contains new constraints that remove invalid solutions filtered out in Step 2. We also introduce a CP model for Step 2. We evaluate scale-up properties of two classical CP solvers (Gecode and Choco) and a hybrid SAT/CP solver (Chuffed). We show that Model 2 is much more efficient than Model 1, and that Chuffed is faster than Choco which is faster than Gecode on the hardest instances of this problem. Furthermore, we prove that a solution claimed to be optimal in two recent cryptanalysis papers is not optimal by providing a better solution.

  Using dedicated hardware is common practice in order to accelerate cryptographic operations: complex operations are managed by a dedicated co-processor and RAM/crypto-engine data transfers are fully managed by DMA operations. The CPU is therefore free for other tasks, which is vital in embedded environments with limited CPU power. In this work we discuss and benchmark XTS-AES, using either software or mixed approaches, using Linux and dm-crypt, and a low-power At-mel(tm) board. This board featurs an AES crypto-engine that supports ECB-AES but not the XTS-AES mode. We show that the `dm-crypt` module used in Linux for full disk encryption has limitations that can be relaxed when considering larger block sizes. In particular we demonstrate in [14] that performance gains almost by a factor two are possible, which opens new opportunities for future use-cases.

6.9. Other results

Participants: Mathieu Cunche, Vincent Roca.

- **Error-correcting codes**

  Recent work have shown that Reed-Muller (RM) codes achieve the erasure channel capacity. However, this performance is obtained with maximum-likelihood decoding which can be costly for practical applications. In [12], we propose an encoding/decoding scheme for Reed-Muller codes on the packet erasure channel based on Plotkin construction. We present several improvements over the generic decoding. They allow, for a light cost, to compete with maximum-likelihood decoding performance, especially on high-rate codes, while significantly outperforming it in terms of speed.

  In [3], we provide fundamentals in the design and analysis of Generalized Low Density Parity Check (GLDPC)-Staircase codes over the erasure channel. These codes are constructed by extending an LDPC-Staircase code (base code) using Reed Solomon (RS) codes (outer codes) in order to benefit from more powerful decoders. The GLDPC-Staircase coding scheme adds, in addition to the LDPC-Staircase repair symbols, extra-repair symbols that can be produced on demand and in large quantities, which provides small rate capabilities. Therefore, these codes are extremely flexible as they can be tuned to behave either like predefined rate LDPC-Staircase codes at one extreme, or like a single RS code at another extreme, or like small rate codes. Concerning the code design,

we show that RS codes with " quasi " Hankel matrix-based construction fulfill the desired structure properties, and that a hybrid (IT/RS/ML) decoding is feasible that achieves Maximum Likelihood (ML) correction capabilities at a lower complexity. Concerning performance analysis, we detail an asymptotic analysis method based on Density evolution (DE), EXtrinsic Information Transfer (EXIT) and the area theorem. Based on several asymptotic and finite length results, after selecting the optimal internal parameters, we demonstrate that GLDPC-Staircase codes feature excellent erasure recovery capabilities, close to that of ideal codes, both with large and very small objects. From this point of view they outperform LDPC-Staircase and Raptor codes, and achieve correction capabilities close to those of RaptorQ codes. Therefore all these results make GLDPC-Staircase codes a universal Application-Layer FEC (AL-FEC) solution for many situations that require erasure protection such as media streaming or file multicast transmission.

PROSECCO Project-Team

7. New Results

7.1. Verification of Security Protocols in the Symbolic Model

Participants: Bruno Blanchet, Marc Sylvestre.

security protocols, symbolic model, automatic verification 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 [27] does not come with proofs, and detailed proofs for the results in this paper were never published. Martín Abadi, Bruno Blanchet, and Cédric Fournet wrote detailed proofs of all results of this paper. This work appears as a research report [21] and is submitted to a journal.

Stéphanie Delaune, Mark Ryan, and Ben Smyth [39] introduced the idea of swapping data in order to prove observational equivalence. For instance, ballot secrecy in electronic voting is formalized by saying that A voting a and B voting b is observationally equivalent to (indistinguishable from) A voting b and B voting a. Proving such an equivalence typically requires swapping the votes. However, Delaune et al's approach was never proved correct. Bruno Blanchet and Ben Smyth filled this gap by formalizing the approach and providing a detailed soundness proof [12], [23]. This extension is implemented in ProVerif. Moreover, Marc Sylvestre implemented a graphical display of attacks in ProVerif. The extended tool is available at http://proverif.inria.fr.

Bruno Blanchet wrote a survey on ProVerif, available both as a book and as a journal paper [3].

7.2. Verification of Security Protocols in the Computational model

Participant: Bruno Blanchet.

security protocols, computational model, verification Bruno Blanchet implemented extensions of his computational protocol verifier CryptoVerif. In particular, the tool collects more precise information at each program point, in order to improve the simplification of cryptographic games and the proof of correspondence assertions (authentication). For instance, this extension allows one to prove injective correspondences for protocols with a replay cache. Another extension provides a query to show that several variables are independent secrets. The extended tool is available at http://cryptoverif.inria.fr.

7.3. Verification of Avionic Security Protocols

Participant: Bruno Blanchet.

security protocols, symbolic model, computational model, verification Within the ANR project AnaStaSec, Bruno Blanchet studied an air-ground avionic security protocol, the ARINC823 public key protocol [24]. He verified this protocol both in the symbolic model of cryptography, using ProVerif, and in the computational model, using CryptoVerif. While this study confirmed the main security properties of the protocol (entity and message authentication, secrecy), he found several weaknesses and imprecisions in the standard. He proposed fixes for these problems. He delivered this work to the ANR and he plans to submit it for publication next year.

7.4. The F* programming language

Participants: Alejandro Aguirre, Danel Ahman [University of Edinburgh], Benjamin Beurdouche, Karthikeyan Bhargavan, Antoine Delignat-Lavaud [Microsoft Research], Cédric Fournet [Microsoft Research], Catalin Hritcu, Chantal Keller [Université Paris-Sud], Kenji Maillard, Guido Martínez, Gordon Plotkin, Samin Ishtiaq [Microsoft Research], Markulf Kohlweiss [Microsoft Research], Jonathan Protzenko [Microsoft Research], Tahina Ramananandro [Microsoft Research], Aseem Rastogi [Microsoft Research], Nikhil Swamy [Microsoft Research], Peng Wang [MIT], Santiago Zanella-Béguelin [Microsoft Research], Jean Karim Zinzindohoué.

F* is a new higher order, effectful programming language (like ML) designed with program verification in mind. Its type system is based on a core that resembles System Fω (hence the name), but is extended with dependent types, refined monadic effects, refinement types, and higher kinds. Together, these features allow expressing precise and compact specifications for programs, including functional correctness properties. The F* type-checker aims to prove that programs meet their specifications using an automated theorem prover (usually Z3) behind the scenes to discharge proof obligations. Programs written in F* can be translated to OCaml, F#, or JavaScript for execution.

We published a paper on the design, implementation, and formal core of F* at POPL 2016 [20]. A first significant improvement on this design will appear at POPL 2017 under the name of "Dijkstra Monads for Free" [6]. Also significant work was put into extracting a subset of F* to C; we submitted a paper on this to PLDI 2017. F* is being developed as an open-source project at GitHub: https://github.com/FStarLang and the official webpage is at http://fstar-lang.org. We released several beta versions of the software this year.

7.5. Dependable Property-Based Testing

Participants: Maxime Dénès [Inria Sophia-Antipolis], Diane Gallois-Wong [ENS and Inria Paris], Catalin Hritcu, John Hughes [Chalmers University], Leonidas Lampropoulos [University of Pennsylvania], Zoe Paraskevopoulou [Princeton University], Benjamin Pierce [University of Pennsylvania], Li-Yao Xia [ENS and Inria Paris].

This year we finally released the Luck programming language for property-based generators (https://github.com/QuickChick/Luck); a paper on this is about to appear at POPL 2017 [18]. We also improved a previous case study on testing information-flow control mechanisms and published a journal paper on this at JCS [1]. Finally, we kept improving the QuickChick testing plugin for Coq (https://github.com/QuickChick/QuickChick), in particular by automatically producing generators from algebraic datatype definitions.

7.6. Micro-Policies and Secure Compilation

Participants: Arthur Azevedo de Amorim [University of Pennsylvania], André Dehon [University of Pennsylvania and Draper Labs], Catalin Hritcu, Yannis Juglaret, Boris Eng, Benjamin Pierce [University of Pennsylvania], Howard Shrobe [MIT], Stelios Sidiroglou-Douskos [MIT], Greg Sullivan [Draper Labs], Andrew Tolmach [Portland State University].

This year we obtained a new ERC Starting Grant on secure compilation using micro-policies; the grant will start in January 2017. Our work was focused on laying the foundations for this long-term research direction. Preliminary work on this appeared at CSF 2016 [17]. In addition, an improved version of our paper on micro-policies for information flow-control appeared at JFP [1]. Finally, we were part of Draper Labs' patent application on "Techniques for Metadata Processing", as developed jointly in the micro-policies project.

7.7. miTLS: Proofs for TLS 1.3

Participants: Karthikeyan Bhargavan, Chris Brzuska [Technical University of Hamburg], Cedric Fournet [Microsoft Research], Matthew Green [Johns Hopkins University], Markulf Kohlweiss [Microsoft Research], Santiago Zanella-Béguelin [Microsoft Research], Jean Karim Zinzindohoué.

transport layer security, cryptographic protocol, verified implementation, man-in-the-middle attack, impersonation attack

We actively participated in the design of TLS 1.3, and worked on a verified implementation of TLS 1.0-1.3 in F*, called miTLS. miTLS is being actively developed on GitHub and we have submitted a paper on our verified implementation of the TLS 1.3 record layer. We published a paper on our overall verification methodology in the IEEE Security and Privacy journal.

Many recent attacks on TLS, discovered by us and others, have relied on *downgrading* a TLS connection and forcing it to use obsolete cryptographic constructions, even if the client and server support and prefer to use modern cryptography. We wrote a paper that showed that such downgrade weaknesses also exist in other protocols such as IPsec, SSH, and ZRTP. We formalized a notion of *downgrade resilience* and showed how it can be achieved in different circumstances. In particular we proved that a new downgrade protection mechanism in TLS 1.3, which was proposed by us, prevents a large class of downgrade attacks. This paper appeared in IEEE S&P (Oakland) 2016 [7].

7.8. Attacks on obsolete cryptography

Participants: Karthikeyan Bhargavan, Gaëtan Leurent.

transport layer security, cryptographic protocol, man-in-the-middle attack, impersonation attack

At NDSS 2016, we published a paper [10] describing a new class of attacks on the use of weak hash functions in popular key exchange protocols such as TLS, IKE, and SSH. One of these attacks, called SLOTH, demonstrated a practical attack on MD5-based client authentication in TLS. We responsibly disclosed this vulnerability, which resulted in security updates in various web browsers and servers. For example, SLOTH-related updates were released for Firefox, Java, RedHat Linux, and for all websites hosted by the Akamai content delivery network.

At CCS 2016, we published a paper [9] that described an attack, called Sweet32, that affects protocols that use block ciphers with short 64-bit blocks, such as Triple-DES and Blowfish. When more than a certain amount of data is sent using such ciphers, the attacker can exploit ciphertext collisions to reconstruct the secret plaintext. We showed how this vulnerability affects TLS and OpenVPN connections. Our findings led to security advisories for OpenVPN, OpenSSL, and all Apple products.

7.9. HACL*: Verified cryptographic library

Participants: Karthikeyan Bhargavan, Jean Karim Zinzindohoué, Marina Polubelova, Benjamin Beurdouche, Jonathan Protzenko [Microsoft Research].

HACL* is a verified cryptographic library written in F*. It implements modern primitives, including elliptic curves like Curve25519, symmetric ciphers like Chacha20, and MAC algorithms like Poly1305. These primitives are then composed into higher-level constructions like Authenticated Encryption with Additional Data (AEAD) and the NaCl API. All the code in HACL* is verified for memory safety, side channel resistance, and where applicable, also for functional correctness and absence of integer overflow. HACL* code is used as the basis for cryptographic proofs for security in the miTLS project.

In CSF 2016, we published a paper on a library of elliptic curves written in F* and compiled to OCaml. This library included the first verified implementations for multiple curves: Curve25519, Curve448, and NIST P-256. However, our code was not very fast. More recently, we worked on Kremlin, a compiler from F* to C that generates code which is as fast as state-of-the-art cryptographic libraries written in C. We have submitted a paper on the Kremlin compiler and its use in HACL*. All our code is being actively and openly developed on GitHub.

7.10. Design and Verification of next-generation protocols: identity, blockchains, and messaging

Participants: Harry Halpin, George Danezis [University College London], Carmela Troncoso [IMDEA].

We began work on designing substantial modifications to existing protocols, verifying pre-standard protocols, or creating entirely new standards for new areas. In these areas the fundamental protocols are often unstandardized and controlled by a few large companies (such as the case of identity-based authorization in terms of Google and Facebook's use of OAuth) and new protocols (such as the incompatible space of protocols around secure messaging given by applications such as WhatsApp, Signal, Telegram, and Viber). In some cases, these protocols do not support basic features needed for standardization, such as decentralization and federation. Therefore, in the first half of 2017, Harry Halpin worked with colleagues at IMDEA and University College London in completing the first systematization of knowledge of decentralization, submitted to PETS 2017, and presented preliminary results in "The Responsibility of Open Standards" paper at the HotPETS 2016 workshop as well as in the First Monday journal.

One of the most important protocols in the entire Web is the OAuth protocol, yet it has suffered from a number of dangerous security and privacy issues. Previously formally analysed by Prosecco, one of the larger problems facing this widely deployed protocol is the lack of privacy. Whenever a user log-ins into a website via Google or Facebook Connect (their identity provider), and then authorizes the flow of data between that website and the identity provider. However, the identity provider then gains knowledge of the every single visit that their users make to other websites that request their data, in addition to the data that the identity provider stores itself. Using a new blind signature scheme based on Algebraic MACs, the new UnlimitID protocol makes the use of federated identity by a user at a website unlinkable to their identity provider, while still allowing websites to gain the advantage of single authenticated sign-on to a large identity to prevent spamming and abuse. This work was presented at the Workshop for Privacy in the Electronic Society at ACM CCS. Unlike previous work that requires substantial changes to both websites using OAuth and identity providers, by using the new W3C Web Crypto API (as analyzed by Halpin), this new protocol requires only changes to the identity provider and is do backwards-compatible with existing OAuth implementations. Microsoft has supported this work for possible future standardization in the OpenID Foundation.

In order to be decentralized, secure messaging requires an ability to discover key material and guarantee its integrity. Typically, today this is done via a single centralized and unstandardised service provider. In order to create an interoperable standard around secure messaging, key discovery needs to be decentralized. Blockchain-based approaches have been suggested in previous work in the security research community such as CONIKS, but have failed to take off due to the high deployment cost on centralized servers. We've designed a new protocol, ClaimChain, that builds on both existing work on blockchains while adding new optimizations and providing a decentralized logic based on Rivest and Lampson's SDSI to identify and discovery key material without a trusted third party. Joint work with CNRS to understand the social and economic considerations led to a publication in Internet Science and the existing design will be submitted to a top-notch security conference. Currently, we are discussing early use of this design with codebases used by secure messaging and email providers, and a security and privacy analysis of these codbases was published in CANS. Over the next year we plan for all of these protocols to have formally verified code for their cryptographic functionality and to present a design on how to integrate this work on key discovery into secure messaging with improved privacy and transcript consistency.

TAMIS Team

7. New Results

7.1. Results for Axis 1: Vulnerability analysis

Statistical model checking employs Monte Carlo methods to avoid the state explosion problem of probabilistic (numerical) model checking. To estimate probabilities or rewards, SMC typically uses a number of statistically independent stochastic simulation traces of a discrete event model. Being independent, the traces may be generated on different machines, so SMC can efficiently exploit parallel computation. Reachable states are generated on the fly and SMC tends to scale polynomially with respect to system description. Properties may be specified in bounded versions of the same temporal logics used in probabilistic model checking. Since SMC is applied to finite traces, it is also possible to use logics and functions that would be intractable or undecidable for numerical techniques.

Several model checking tools have added SMC as a complement to exhaustive model checking. This includes the model checker UPPAAL, for timed automata, the probabilistic model checker PRISM, and the model checker Ymer, for continuous time Markov chains. Plasma Lab [29] is the first platform entirely dedicated to SMC. Contrary to other tools, that target a specific domain and offer several analysis techniques, including basic SMC algorithms, Plasma Lab is designed as a generic platform that facilitates multiple SMC algorithms, multiple modelling and query languages and has multiple modes of use. This allows us to apply statistical model checking techniques to a wide variety of problems, reusing existing simulators. With this process we avoid the task of rewriting a model of a system in a language not ideally design to do it. This complex task often leads either to an approximation of the original system or to a more complex model harder to analyze. The task needed to support a new simulator is to implement an interface plugin between our platform Plasma Lab and the existing tool, using the public API of our platform. This task has to be performed only once to analyze all the systems supported by the existing simulator.

Plasma Lab can already be used with the PRISM language for continuous and discrete time Markov chains and biological models. During the last years we have developed several new plugins to support SystemC language [50], Simulink models [70], dynamic software architectures language [41], [14], and train interlocking systems [64]. They have been presented in recent publications.

[50] Transaction-level modeling with SystemC has been very successful in describing the behavior of embedded systems by providing high-level executable models, in which many of them have an inherent probabilistic behavior, i.e., random data, unreliable components. It is crucial to evaluate the quantitative and qualitative analysis of the probability of the system properties. Such analysis can be conducted by constructing a formal model of the system and using probabilistic model checking. However, this method is unfeasible for large and complex systems due to the state space explosion. In this paper, we demonstrate the successful use of statistical model checking to carry out such analysis directly from large SystemC models and allows designers to express a wide range of useful properties.

[70] We present an extension of the statistical model checker Plasma Lab capable of analyzing Simulink models.

[41], Dynamic software architectures emerge when addressing important features of contemporary systems, which often operate in dynamic environments subjected to change. Such systems are designed to be reconfigured over time while maintaining important properties, e.g., availability, correctness, etc. Verifying that reconfiguration operations make the system to meet the desired properties remains a major challenge. First, the verification process itself becomes often difficult when using exhaustive formal methods (such as model checking) due to the potentially infinite state space. Second, it is necessary to express the properties to be verified using some notation able to cope with the dynamic nature of these systems. Aiming at tackling these issues, we introduce

DynBLTL, a new logic tailored to express both structural and behavioral properties in dynamic software architectures. Furthermore, we propose using statistical model checking (SMC) to support an efficient analysis of these properties by evaluating the probability of meeting them through a number of simulations. In this paper, we describe the main features of DynBLTL and how it was implemented as a plug-in for PLASMA, a statistical model checker.

[14] The critical nature of many complex software-intensive systems calls for formal, rigorous architecture descriptions as means of supporting automated verification and enforcement of architectural properties and constraints. Model checking has been one of the most used techniques to automatically analyze software architectures with respect to the satisfaction of architectural properties. However, such a technique leads to an exhaustive exploration of all possible states of the system under verification, a problem that becomes more severe when verifying dynamic software systems due to their typical non-deterministic runtime behavior and unpredictable operation conditions. To tackle these issues, we propose using statistical model checking (SMC) to support the analysis of dynamic software architectures while aiming at reducing effort, computational resources, and time required for this task. In this paper, we introduce a novel notation to formally express architectural properties as well as an SMC-based toolchain for verifying dynamic software architectures described in π-ADL, a formal architecture description language. We use a flood monitoring system to show how to express relevant properties to be verified, as well as we report the results of some computational experiments performed to assess the efficiency of our approach.

[64] , accepted at HASE 2017 In the railway domain, an interlocking is the system ensuring safe train traffic inside a station by controlling its active elements such as the signals or points. Modern interlockings are configured using particular data, called application data, reflecting the track layout and defining the actions that the interlocking can take. The safety of the train traffic relies thereby on application data correctness, errors inside them can cause safety issues such as derailments or collisions. Given the high level of safety required by such a system, its verification is a critical concern. In addition to the safety, an interlocking must also ensure that availability properties, stating that no train would be stopped forever in a station, are satisfied. Most of the research dealing with this verification relies on model checking. However, due to the state space explosion problem, this approach does not scale for large stations. More recently, a discrete event simulation approach limiting the verification to a set of likely scenarios, was proposed. The simulation enables the verification of larger stations, but with no proof that all the interesting scenarios are covered by the simulation. In this paper, we apply an intermediate statistical model checking approach, offering both the advantages of model checking and simulation. Even if exhaustiveness is not obtained, statistical model checking evaluates with a parameterizable confidence the reliability and the availability of the entire system.

7.1.1. Verification of Dynamic Software Architectures

Participants: Axel Legay, Jean Quilbeuf, Louis-Marie Traonouez.

Dynamic software architectures emerge when addressing important features of contemporary systems, which often operate in dynamic environments subjected to change. Such systems are designed to be reconfigured over time while maintaining important properties, e.g., availability, correctness, etc. π-ADL is a formal, well-founded theoretically language intended to describe software architectures under both structural and behavioral viewpoints. In order to cope with dynamicity concerns, π-ADL is endowed with architectural level primitives for specifying programmed reconfiguration operations, i.e., foreseen, pre-planned changes described at design time and triggered at runtime by the system itself under a given condition or event. Additionally, code source in the Go programming language is automatically generated from π-ADL architecture descriptions, thereby allowing for their execution.

We have developed with Plasma Lab a toolchain [14] for verifying dynamic software architectures described in π-ADL. The architecture description in π-ADL is translated towards generating source code in Go. As π-ADL architectural models do not have a stochastic execution, they are linked to a stochastic scheduler parameterized by a probability distribution for drawing the next action. Furthermore, we use existing probability distribution

Go libraries to model inputs of system models as user functions. The program resulting from the compilation of the generated Go source code emits messages referring to transitions from addition, attachment, detachment, and value exchanges of architectural elements. Additionally we have introduced DynBLTL [41] a new logic tailored to express both structural and behavioral properties in dynamic software architectures.

We have developed two plugins atop the PLASMA platform, namely (i) a simulator plug-in that interprets execution traces produced by the generated Go program and (ii) a checker plugin that implements DynBLTL. With this toolchain, a software architect is able to evaluate the probability of a π-ADL architectural model to satisfy a given property specified in DynBLTL.

7.1.2. Statistical Model-Checking of Scheduling Systems

Participants: Axel Legay, Louis-Marie Traonouez.

Cyber-Physical Systems (CPS) are software implemented control systems that control physical objects in the real world. These systems are being increasingly used in many critical systems, such as avionics and automotive systems. They are now integrated into high performance platforms, with shared resources. This motivates the development of efficient design and verification methodologies to assess the correctness of CPS.

Schedulability analysis is a major problem in the design of CPS. Software computations that implements the commands sent to the CPS are split into a set of hard real-time tasks, often periodic. These tasks are associated to strict deadlines that must be satisfied. A scheduler is responsible for dispatching a shared resource (usually CPU computation time) among the different tasks according to a chosen scheduling policy. The schedulability analysis consists in verifying that the tasks always meet their deadlines.

Over the years, the schedulability of CPS have mainly been performed by analytical methods. Those techniques are known to be effective but limited to a few classes of scheduling policies. In a series of recent work, it has been shown that schedulability analysis of CPS could be performed with a model-based approach and extensions of verification tools such as UPPAAL. It shows that such models are flexible enough to embed various types of scheduling policies that go beyond those in the scope of analytical tools.

We have extended these works to include more complex features in the design of these systems and we have experimented the use of statistical model checking as a lightweight verification technique for these systems.

We also extended the approach to statistical model checking of products lines. Our first contribution has been to propose models to design software product lines (SPL) of preemptive real-time systems [25]. Software Product Line Engineering (SPLE) allows reusing software assets by managing the commonality and variability of products. Recently, SPLE has gained a lot of attention as an approach for developing a wide range of software products from non-critical to critical software products, and from application software to platform software products.

Real-time software products (such as real-time operating systems) are a class of systems for which SPLE techniques have not drawn much attention from researchers, despite the need to efficiently reuse and customize real-time artifacts. We have proposed a formal SPLE framework for real-time systems. It focuses on the formal analysis of real-time properties of an SPL in terms of resource sharing with time dependent functionalities. Our framework provides a structural description of the variability and the properties of a real time system, and behavioral models to verify the properties using formal techniques implemented in the tools UPPAAL symbolic model checker and UPPAAL statistical model checker. For the specification of an SPL, we propose an extension of a feature model, called Property Feature Model (PFM). A PFM explicitly distinguishes features and properties associated with features, so that properties are analyzed in the context of the relevant features. We also define a non-deterministic decision process that automatically configures the products of an SPL that satisfy the constraints of a given PFM and the product conditions of customers. Finally we analyze the products against the associated properties. For analyzing real-time properties, we provide feature behavioral models of the components of a scheduling unit, i.e. tasks, resources and schedulers. Using these feature behavioral models, a family of scheduling units of an SPL is formally analyzed against the designated properties with model checking techniques.

[25] This paper presents a formal analysis framework to analyze a family of platform products w.r.t. real-time properties. First, we propose an extension of the widely-used feature model, called Property Feature Model (PFM), that distinguishes features and properties explicitly. Second, we present formal behavioral models of components of a real-time scheduling unit such that all real-time scheduling units implied by a PFM are automatically composed to be analyzed against the properties given by the PFM. We apply our approach to the verification of the schedulability of a family of scheduling units using the symbolic and statistical model checkers of UPPAAL.

7.1.3. Model-based Framework for Hierarchical Scheduling Systems

Participants: Axel Legay, Louis-Marie Traonouez, Mounir Chadli.

In order to reduce costs in the design of modern CPS, manufacturers devote strong efforts to maximize the number of components that can be integrated on a given platform. This can be achieved by minimizing the resource requirements of individual components. A hierarchical scheduling systems (HSS) integrates a number of components into a single system running on one execution platform. Hierarchical scheduling systems have been gaining more attention by automotive and aircraft manufacturers because they are practical in minimizing the cost and energy of operating applications.

Several papers have proposed model-based compositional framework for HSS. In [4] we proposed a methodology for optimizing the resource requirement of a component of an HSS using model checking techniques. Our methodology consists of using a lightweight statistical model checking method and a costly but absolute certain symbolic model checking method that operates on identical models.

In another work [15] we have proposed stochastic extension of HSS that allows us to capture tasks whose real-time attributes, such as deadline, execution time or period, are also characterized by probability distributions. This is particularly useful to describe mixed-critical systems and make assumptions on the hardware domain. These systems combine hard real-time periodic tasks, with soft real-time sporadic tasks. Classical scheduling techniques can only reason about worst case analysis of these systems, and therefore always return pessimistic results. Using tasks with stochastic period we can better quantify the occurrence of these tasks. Similarly, using stochastic deadlines we can relax timing requirements, and stochastic execution times are used to model the variation of the computation time needed by the tasks. These distributions can be sampled from executions or simulations of the system, or set as requirements from the specifications. For instance in avionics, display components have lower criticality. They can include sporadic tasks generated by users requests. Average user demand is efficiently modeled with a probability distribution.

We have also developed a graphical high-level language to represent scheduling units and complex hierarchical scheduling systems. In order to bridge the gap between the formalisms, we exploit Cinco, a generator for domain specific modeling tools to generate an interface between this language and the one of UPPAAL. Cinco allows to specify the features of a graphical interface in a compact meta-model language. This is a flexible approach that could be extended to any formal model of scheduling problem.

[4] Compositional reasoning on hierarchical scheduling systems is a well-founded formal method that can construct schedulable and optimal system configurations in a compositional way. However, a compositional framework formulates the resource requirement of a component, called an interface, by assuming that a resource is always supplied by the parent components in the most pessimistic way. For this reason, the component interface demands more resources than the amount of resources that are really sufficient to satisfy sub-components. We provide two new supply bound functions which provides tighter bounds on the resource requirements of individual components. The tighter bounds are calculated by using more information about the scheduling system. We evaluate our new tighter bounds by using a model-based schedulability framework for hierarchical scheduling systems realized as UPPAAL models. The timed models are checked using model checking tools UPPAAL and UPPAAL SMC, and we compare our results with the state of the art tool CARTS.

[15] Over the years, schedulability of Cyber-Physical Systems (CPS) have mainly been performed by analytical methods. Those techniques are known to be effective but limited to a few classes of scheduling policies. In a series of recent work, we have shown that schedulability analysis of

CPS could be performed with a model-based approach and extensions of verification tools such as UPPAAL. One of our main contribution has been to show that such models are flexible enough to embed various types of scheduling policies that go beyond those in the scope of analytical tools. In this paper, we go one step further and show how our formalism can be extended to account for stochastic information, such as sporadic tasks whose attributes depend on the hardware domain. Our second contribution is to make our tools accessible to average users that are not experts in formal methods. For doing so, we propose a graphical and user-friendly language that allows us to describe scheduling problems. This language is automatically translated to formal models by exploiting a meta-model approach. The principle is illustrated on a case study.

7.1.4. Verification of Interlocking Systems

Participants: Axel Legay, Louis-Marie Traonouez, Jean Quilbeuf.

An interlocking is a system that controls the train traffic by acting as an interface between the trains and the railway track components. The track components are for example, the signals that allow the train to proceed, or the points that guide the trains from one track to another. The paths followed by the trains are called routes. Modern interlockings are computerized systems that are composed of generic software and application data.

We have proposed in collaboration with Université Catholique de Louvain and Alstom a method to automatically verify an interlocking using simulation and statistical model checking [64]. We use a simulator developed by Université Catholique de Louvain that is able to generate traces of the interlocking systems from a track layout and application data. This simulator is plug with Plasma Lab using a small interface developed with Plasma Lab's API. Then, the traces generated by the simulator have been used by Plasma Lab SMC algorithms to measure the correctness of the system. We have used Monte-Carlo and importance splitting algorithms to verify this system.

7.1.5. Advanced Statistical Model Checking

Participants: Axel Legay, Sean Sedwards, Louis-Marie Traonouez.

Statistical model checking (SMC) addresses the state explosion problem of numerical model checking by estimating quantitative properties using simulation. Rare events, such as software bugs, are often critical to the performance of systems but are infrequently observed in simulations. They are therefore difficult to quantify using SMC. Nondeterministic systems deliberately leave parts of system behaviour undefined, hence it is not immediately possible to simulate them. Our ongoing work thus pushes the boundaries of the cutting edge of SMC technology by focusing on rare event verification and the optimisation of nondeterminism.

7.1.5.1. Optimizing Nondeterministic Systems

[17] Probabilistic timed automata (PTA) generalize Markov decision processes (MDPs) and timed automata (TA), both of which include nondeterminism. MDPs have discrete nondeterministic choices, while TA have continuous nondeterministic time. In this work we consider finding *schedulers* that resolve all nondeterministic choices in order to maximize or minimize the probability of a time-bounded LTL property. Exhaustive numerical approaches often fail due to state explosion, hence we present a new lightweight on-the-fly algorithm to find near-optimal schedulers. To discretize the continuous choices we make use of the classical region and zone abstractions from timed automata model checking. We then apply our recently developed "smart sampling" technique for statistical verification of Markov decision processes. On standard case studies our algorithm provides good estimates for both maximum and minimum probabilities. We compare our new approach with alternative techniques, first using tractable examples from the literature, then motivate its scalability using case studies that are intractable to numerical model checking and challenging for existing statistical techniques.

7.1.5.2. Rare Event Verification

[3] Importance sampling is a standard technique to significantly reduce the computational cost of quantifying rare properties of probabilistic systems. It works by weighting the original distribution of the system to make the rare property appear more frequently in simulations, then compensating the resulting estimate by the weights. This can be done on the fly with minimal storage, but the challenge is to find *near optimal* importance sampling distributions efficiently, where optimal means that paths that do not satisfy the property are never seen, while paths that satisfy the property appear in the same proportion as in the original distribution.

Our approach uses a tractable cross-entropy minimization algorithm to find an optimal parameterized importance sampling distribution. In contrast to previous work, our algorithm uses a naturally defined low dimensional vector to specify the distribution, thus avoiding an explicit representation of a transition matrix. Our parametrisation leads to a unique optimum and is shown to produce many orders of magnitude improvement in efficiency on various models. In this work we specifically link the existence of optimal importance sampling distributions to time-bounded logical properties and show how our parametrisation affects this link. We also motivate and present simple algorithms to create the initial distribution necessary for cross-entropy minimization. Finally, we discuss the open challenge of defining error bounds with importance sampling and describe how our optimal parameterized distributions may be used to infer qualitative confidence.

[10] In this work we consider rare events in systems of Stochastic Timed Automata (STA) with time-bounded reachability properties. This model may include rarity arising from explicit discrete transitions, as well as more challenging rarity that results from the intersection of timing constraints and continuous distributions of time. Rare events have been considered with simple combinations of continuous distributions before, e.g., in the context of queueing networks, but we present an automated framework able to work with arbitrarily composed STA. By means of symbolic exploration we first construct a zone graph that excludes unfeasible times. We then simulate the system within the zone graph, avoiding "dead ends" on the fly and proportionally redistributing their probability to feasible transitions. In contrast to many other importance sampling approaches, our "proportional dead end avoidance" technique is guaranteed by construction to reduce the variance of the estimator with respect to simulating the original system. Our results demonstrate that in practice we can achieve substantial overall computational gains, despite the symbolic analysis.

[49] In this invited paper we outline some of our achievements in quantifying rare properties in the context of SMC. In addition to the importance sampling techniques described above, we also describe our work on importance *splitting*. Importance splitting works by decomposing the probability of a rare property into a product of probabilities of sub-properties that are easier to estimate. The sub-properties are defined by *levels* of a *score function* that maps states of the system$\times$property product automaton to values. We have provided the first general purpose implementation of this approach, using user-accessible "observers" that are compiled automatically from the property. These observers may be used by both fixed and adaptive level importance splitting algorithms and are specifically designed to make distribution efficient.

7.1.6. Side-channel Analysis of Cryptographic Substitution Boxes

Participants: Axel Legay, Annelie Heuser.

With the advent of the Internet of Things, we are surrounded with smart objects (aka things) that have the ability to communicate with each other and with centralized resources. The two most common and widely noticed artefacts are RFID and Wireless Sensor Networks which are used in supply-chain management, logistics, home automation, surveillance, traffic control, medical monitoring, and many more. Most of these applications have the need for cryptographic secure components which inspired research on cryptographic algorithms for constrained devices. Accordingly, lightweight cryptography has been an active research area over the last 10 years. A number of innovative ciphers have been proposed in order to optimize various performance criteria and have been subject to many comparisons. Lately, the resistance against side-channel attacks has been considered as an additional decision factor.

Side-channel attacks analyze physical leakage that is unintentionally emitted during cryptographic operations in a device (e.g., power consumption, electromagnetic emanation). This side-channel leakage is statistically dependent on intermediate processed values involving the secret key, which makes it possible to retrieve the secret from the measured data.

Side-channel analysis (SCA) for lightweight ciphers is of particular interest not only because of the apparent lack of research so far, but also because of the interesting properties of substitution boxes (S-boxes). Since the nonlinearity property for S-boxes usually used in lightweight ciphers (i.e., 4×4) can be maximally equal to 4, the difference between the input and the output of an S-box is much smaller than for instance for AES. Therefore, one could conclude that from that aspect, SCA for lightweight ciphers must be more difficult. However, the number of possible classes (e.g., Hamming weight (HW) or key classes) is significantly lower, which may indicate that SCA must be easier than for standard ciphers. Besides the difference in the number of classes and consequently probabilities of correct classification, there is also a huge time and space complexity advantage (for the attacker) when dealing with lightweight ciphers.

In [23], [67] we give a detailed study of lightweight ciphers in terms of side-channel resistance, in particular for software implementations. As a point of exploitation we concentrate on the non-linear operation (S-box) during the first round. Our comparison includes SPN ciphers with 4-bit S-boxes such as KLEIN, PRESENT, PRIDE, RECTANGLE, Mysterion as well as ciphers with 8-bit S-boxes: AES, Zorro, Robin. Furthermore, using simulated data for various signal-to-noise ratios (SNR) we present empirical results for Correlation Power Analysis (CPA) and discuss the difference between attacking 4-bit and 8-bit S-boxes.

Following this direction current studies evaluate and connect cryptographic properties with side-channel resistance. More precisely, in an ideal setting a cipher should be resilient against cryptanalyses as well as side-channel attacks and yet easy and cheap to be implemented. However, since that does not seem to be possible at the current level of knowledge, we are required to make a number of trade-offs. Therefore, we investigate several S-box parameters and connect them with well known cryptographic properties of S-boxes. Moreover, when possible we give clear theoretical bounds on those parameters as well as expressions connecting them with properties like nonlinearity and δ-uniformity. We emphasize that we select the parameters to explore on the basis of their possible connections with the side-channel resilience of S-boxes.

To this end, we divide the primary goal into several sub-problems. First, we discuss what is the maximal number of fixed points one can have in an optimal S-box. The question of the maximal number of fixed points for an optimal S-box is of interest on its own, but also in the side-channel context since intuitively an S-box with many fixed points should consume less power and therefore have less leakage. Moreover, the preservation of Hamming weight and a small Hamming distance between x and $F(x)$ are two more properties each of which could strengthen the resistance to SCA. Our findings show that notably in the case when exactly preserving the Hamming weight, the confusion coefficient reaches good value and consequently the S-box has good SCA resilience. We show that the S-boxes with no differences in the Hamming weight of their input and output pairs (and even, S-boxes F such that $F(x)$ have Hamming weight near the Hamming weight of x, on average) or S-boxes such that $F(x)$ lies at a small Hamming distance from x cannot have high nonlinearity (although the obtainable values are not too bad for $n = 4, 8$) and therefore are not attractive in practical applications. Note that an S-box with many fixed points is also a particular case of an S-box that preserves the Hamming weight/distance between the inputs and outputs. Furthermore, our study includes involutive functions since they have a particular advantage over general pseudo-permutations. In particular, not only from an implementation viewpoint but also their side-channel resilience is the same regardless if an attacker considers the encryption or decryption phase as well as attacking the first or the last round. Next, we find a theoretical expression connecting the confusion coefficient with that of preserving the Hamming weight of inputs and outputs.

In the practical part, we first confirm our theoretical findings about the connection between preserving Hamming weight and the confusion coefficient. Besides that, we give a number of S-box examples of size 4×4 intended to provide more insight into possible trade-offs between cryptographic properties and side-channel resilience. However, our study shows that mostly preserving Hamming weight might not automatically result in a small minimum confusion coefficient and thus in higher side-channel resistance. We therefore in

detail examine the influence of F on the confusion coefficient in general by concentrating on the input (in which key hypothesis are made) and the minimum value of the confusion coefficient. Following, we evaluate a number of S-boxes used in today's ciphers and show that their SCA resilience can significantly differ. Finally, we point out that non-involutive S-boxes might bring a significant disadvantage in case an attacker combines the information about F and F^{-1} by either targeting both first and last round of an algorithm or encryption and decryption.

[67] Side-channel Analysis of Lightweight Ciphers: Current Status and Future Directions

[23] Side-channel Analysis of Lightweight Ciphers: Does Lightweight Equal Easy?

7.1.7. Binary Code Analysis: Formal Methods for Fault Injection Vulnerability Detection

Participants: Axel Legay, Thomas Given-Wilson, Nisrine Jafri, Jean-Louis Lanet.

Formal methods such as model checking provide a powerful tool for checking the behaviour of a system. By checking the properties that define correct system behaviour, a system can be determined to be correct (or not).

Increasingly fault injection is being used as both a method to attack a system by a malicious attacker, and to evaluate the dependability of the system. By finding fault injection vulnerabilities in a system, the resistance to attacks or faults can be detected and subsequently addressed.

A process is presented that allows for the automated simulation of fault injections. This process proceeds by taking the executable binary for the system to be tested, and validating the properties that represent correct system behaviour using model checking. A fault is then injected into the executable binary to produce a mutant binary, and the mutant binary is model checked also. A different result to the validation of the executable binary in the checking of the mutant binary indicates a fault injection vulnerability.

This process has been automated with existing tools, allowing for easy checking of many different fault injection attacks and detection of fault injection vulnerabilities. This allows for the detection of fault injection vulnerabilities to be fully automated, and broad coverage of the system to be formally shown.

7.1.8. Security at the hardware and software boundaries

Participants: Axel Legay, Nisrine Jafri, Jean-Louis Lanet, Ronan Lashermes, Hélène Le Bouder.

7.1.8.1. IoT security

IoT security has to face all the challenges of the mainstream computer security but also new threats. When an IoT device is deployed, most of the time it operates in a hostile environment, i.e. the attacker can perform any attack on the device. If secure devices use tamper resistant chip and are programmed in a secure manner, IoT use low cost micro-controllers and are not programmed in a secure way. We developed new attacks but also evaluate how the code polymorphism can be used against these attacks. In [45] [27] we developed a template attack to retrieve the value of a PIN code from a cellphone. We demonstrated that the maximum trials to retrieve the four bytes of secret PIN is 8 and in average 3 attempts are sufficient. A supervised learning algorithm is used.

Often smart phones allow up to 10 attempts before locking definitely the memory. We used an embedded code generator [16], [45] dedicated to a given security function using a DSL to increase the security level of a non tamper resistant chip. We brought to the fore that a design of the software for protecting against fault attacks decreases the security against SCA. Fault attack is a mean to execute a code that is slightly different from the one that has been loaded into the device. Thus, to be sure that a genuine code cannot be dynamically transformed, one needs to analyze any possibility of a code to be transformed.

The work presented in [34] made possible to design an extremely effective architecture to achieve Montgomery modular multiplication. The proposed solution combines a limited resource consumption with the lowest latency compared with the literature. This allows to envisage new applications of asymmetric cryptography in systems with few resources. In order to find a cryptographic key using hidden channels, most attacks use the a priori knowledge of texts sent or received by the target. The proposed analysis presented in [28] does not use these assumptions. A belief propagation technique is used to cross the information obtained from leaked information with the equations governing the targeted algorithm.

7.1.8.2. Safe update mechanism for IoT

One of the challenges for IoT is the possibility to update the code through a network. This is done by stopping the system, loading the new version, verifying the signature of the firmware and installing it into the memory. Then, the memory must be cleaned to eliminate the code and the data of the previous version. Some IoT (sensor acquisition and physical system control) requires to never stop while executing the code. We have developed a complete architecture that performs such an update with real time capabilities. If one wants to use this characteristic in a real world it should pass certification. In particular he has to demonstrate that the system performs as expected. We used formal methods (mainly Coq) to demonstrate that the semantics of the code is preserved during the update. In [30], we paid attention to the detection of the Safe Update Point (SUP) because our implementation had some time an unstable behavior. We demonstrated that in a specific case, while several threads using code to be updated, the system enters into a deadlock. After discovering the bug, we patched our system.

7.1.8.3. Reverse engineering of firmware

Reverse engineering has two aspects; code reverse for which the literature is abundant and data reverse i.e. understanding the meaning of a structure and its usage has been less studied. The first step in reverse engineering consists in getting access of the code. In the case of romized code in a SoC, the access to the code is protected by a MMU mechanism and thus is an efficient mitigation mechanism against reverse engineering. In [8], [2] and [33] we set up several attacks to get access to the code even in presence of a MMU. The attack in [8] uses a vulnerability in the API where an object can be used instead of an array. This allows to read and write the code area leading to the possibility to execute arbitrary code in memory. In [33], we use the attack tree paradigm to explore all the possibilities to mount an attack on a given product. In [2], we used a ROP (Return Oriented Programming) attack to inject a shell code in the context of another application. Due to the fact that the shell code is executed in the context of the caller, the firewall is unable to detect the access to the secure container of the targeted application. This allows us to retrieve the content of the secure containers.

Once the dump has been obtained, one can try to retrieve code and data. Retrieving code is not obvious but several tools exist to help the analyst. These tools require that all the ISA (Instruction Set Architecture) is known. Sometime, the ISA is not known and in particular when one wants to obfuscate the code, he can use a virtual machine to execute dedicated byte code. In [32], we developed a methodology to infer the missing byte code, then we execute a data type inference to understand the memory management algorithm.

7.2. Results for Axis 2: Malware analysis

The detection of malicious programs is a fundamental step to be able to guarantee system security. Programs that exhibit malicious behavior, or *malware*, are commonly used in all sort of cyberattacks. They can be used to gain remote access on a system, spy on its users, exfiltrate and modify data, execute denial of services attacks, etc.

Significant efforts are being undertaken by software and data companies and researchers to protect systems, locate infections, and reverse damage inflicted by malware. Malware analysis can be divided in the following three main problems:

7.2.1. Malware Detection

Participants: Axel Legay, Fabrizio Biondi, Olivier Decourbe, Mike Enescu, Thomas Given-Wilson, Annelie Heuser, Nisrine Jafri, Jean-Louis Lanet, Jean Quilbeuf.

Given a file or data stream, the malware detection problem consists of understanding if the file or data stream contain traces of malicious behavior. For binary executable files in particular, this requires reverse engineering the file's behavior to understand if it is malicious. The main reverse engineering techniques are categorized as:

Static Analysis This refers to techniques that analyze the file without executing it. It includes disassembling the file's executable code and analyzing other static features of the binary, like its import/export table, hash, etc. The file's control flow and system flow graphs can be retrieved statically (unless they are obfuscated; see below) and used to guide the exploration of the file's semantics in the search of

malicious behavior. Information flow can be tracked since hostile applications often try to transmit private information to distant servers (this form of malware are now widely spread in the mobile world). The challenge consists in detecting into a file that a private information does not leak to the external world. The verification can be done statically, dealing with storage channel (implicit or explicit), but not with side channel.

Dynamic Analysis This refers to techniques that actually executed the file in a sandbox (usually a virtualized environment) and analyze its interaction with the sandbox. This technique is effective in understanding the file's actual interactions with the system, making it easy to detect malicious behavior. However, malware often implements sandbox detection techniques to detect when it is being run in a virtualized environment, when functions or system calls are hooked by the analyst, or when the sandbox does not look like a normal user's machine (e.g. because it does not contain any document). Dynamic tracking of information flow makes it possible to cope with side channel attacks. With temporal side channel, the challenge lies in the potential declassification procedure used by malware to escape the analysis. We extend the TaintDroid framework to cope with native code invocation [47]. This approach reduces the false positive warning drastically. Recently we have extended this work to cope with timing side channels [under submission]. We are developing a new malware that declassifies the labels thanks to the audio system of the smart-phone. This is a joint work with Telecom Bretagne.

Hybrid Analysis This refers to technique that combine both static and dynamic behavior, i.e. both code analysis and execution. While more complex to implement, these techniques are able to overcome many of the shortcomings of full static and full dynamic analysis. The best example of a hybrid technique is concolic (a portmanteau for CONCrete + symbOLIC) analysis.

To contribute to concolic analysis, we are working on the state-of-the-art `angr` concolic execution engine to make it fast and efficient enough to analyze large executable malware files efficiently. We are improving `angr` 's parallelism and allowing it to precompute semantic stubs of function and system calls, allowing it to focus its analysis on the main file without having to branch in the rest of the operative system. We plan to contribute our improvements to the main `angr` branch, so that the whole community can benefit from them.

7.2.2. Malware Deobfuscation

Participants: Axel Legay, Fabrizio Biondi, Olivier Decourbe, Mike Enescu, Thomas Given-Wilson, Annelie Heuser, Nisrine Jafri, Jean-Louis Lanet, Jean Quilbeuf.

Given a file (usually a portable executable binary or a document supporting script macros), deobfuscation refers to the preparation of the file for the purposes of further analysis. Obfuscation techniques are specifically developed by malware creators to hinder detection reverse engineering of malicious behavior. Some of these techniques include:

Packing Packing refers to the transformation of the malware code in a compressed version to be dynamically decompressed into memory and executed from there at runtime. Packing techniques are particularly effective against static analysis, since it is very difficult to determine statically the content of the unpacked memory to be executed, particularly if packing is used multiple times. The compressed code can also be encrypted, with the key being generated in a different part of the code and used by the unpacking procedure, or even transmitted remotely from a command and control (C&C) server.

Control Flow Flattening This technique aims to hinder the reconstruction of the control flow of the malware. The malware's operation are divided into basic blocks, and a dispatcher function is created that calls the blocks in the correct order to execute the malicious behavior. Each block after its execution returns control to the dispatcher, so the control flow is flattened to two levels: the dispatcher above and all the basic blocks below.

To prevent reverse engineering of the dispatcher, it is often implemented with a cryptographic hash function. A more advanced variant of this techniques embed a full virtual machine with a randomly generated instruction set, a virtual program counted, and a virtual stack in the code, and uses the machine's interpreter as the dispatcher.

Virtualization is a very effective technique to prevent reverse engineering. To contrast it, we are implementing state-of-the-art devirtualization algorithms in `angr` , allowing it to detect and ignore the virtual machine code and retrieving the obfuscated program logic. Again, we plan to contribute our improvements to the main `angr` branch, thus helping the whole security community fighting virtualized malware.

Opaque Constants and Conditionals Reversing packing and control flow flattening techniques requires understanding of the constants and conditionals in the program, hence many techniques are deployed to obfuscate them and make them unreadable by reverse engineering techniques. Such techniques are used e.g. to obfuscate the decryption keys of packed encrypted code and the conditionals in the control flow.

We have proven the efficiency of dynamic synthesis in retrieving opaque constant and conditionals, compared to the state-of-the-art approach of using SMT (Satisfiability Modulo Theories) solvers, when the input space of the opaque function is small enough. We are developing techniques based on fragmenting and analyzing by brute force the input space of opaque conditionals, and SMT constraints in general, to be integrated in SMT solvers to improve their effectiveness.

7.2.3. Malware Classification

Participants: Axel Legay, Fabrizio Biondi, Olivier Decourbe, Mike Enescu, Thomas Given-Wilson, Annelie Heuser, Nisrine Jafri, Jean-Louis Lanet, Jean Quilbeuf.

Once malicious behavior has been located, it is essential to be able to classify the malware in its specific family to know how to disinfect the system and reverse the damage inflicted on it.

While it is rare to find an actually previously unknown malware, morphic techniques are employed by malware creators to ensure that different generations of the same malware behave differently enough than it is hard to recognize them as belonging to the same family. In particular, techniques based on the syntax of the program fails against morphic malware, since syntax can be easily changed.

To this end, semantic signatures are used to classify malware in the appropriate family. Semantic signatures capture the malware's behavior, and are thus resistant to morphic and differentiation techniques that modify the malware's syntactic signatures. We are investigating semantic signatures based on the program's System Call Dependency Graph (SCDG), which have been proven to be effective and compact enough to be used in practice. SCDGs are often extracted using a technique based on pushdown automata that is ineffective against obfuscated code; instead, we are applying concolic analysis via the `angr` engine to improve speed and coverage of the extraction.

Once a semantic signature has been extracted, it has to be compared against large database of known signatures representing the various malware families to classify it. The most efficient way to obtain this is to use a supervised machine learning classifier. In this approach, the classifier is trained with a large sample of signatures malware annotated with the appropriate information about the malware families, so that it can learn to quickly and automatically classify signatures in the appropriate family. Our work on machine learning classification focuses on using SCDGs as signatures. Since SCDGs are graphs, we are investigating and adapting algorithms for the machine learning classification of graphs, usually based on measures of shared subgraphs between different graphs.

In malware detection and classification, it is fundamental to have a false positive rate (i.e. rate of cleanware classified as malware) approaching zero, otherwise the classification system will classify hundred or thousands of cleanware files as malware, making it useless in practice. To decrease the false positive rate, the classifier is also trained with a large and representative database of cleanware, so that it can discriminate between signatures of cleanware and malware with a minimal false positive rate. We use a large database of malware and cleanware to train our classifier, thus guaranteeing a high detection rate with a small false positive rate.

7.2.4. Papers

This section gathers papers that are results common to all sections above pertaining to Axis 2.

[57] Black-box synthesis is more efficient than SMT deobfuscation on predicates obfuscated with Mixed-Boolean Arithmetics.

[66] Recently fault injection has increasingly been used both to attack software applications, and to test system robustness. Detecting fault injection vulnerabilities has been approached with a variety of methods, yielding varied results. This paper proposes a general process using model checking to detect fault injection vulnerabilities in binaries. The process is implemented and used to detect a variety of different kinds of fault injection vulnerabilities in binaries.

[59] Fault-injection exploits hardware weaknesses to perturbate the behaviour of embedded devices. Here, we present new model-based techniques and tools to detect such attacks developed at the High-Security Laboratory at Inria.

[52] We proposed to use a bare metal approach without virtualization and a method to let the system stop the execution while the malware has been deployed in memory.

[51] We present our framework to grab sample from the net, evaluate it on victim PC and detect its presence thanks to our counter measures.

[53] In this paper, two counter measures are presented. The first one is related with the mode ECB of the AES cryptographic algorithm and the second is related with the usage of the crypto API. We developed a cryptographic provider which intercepts the key generation and store it in a safe place. Then we are able to decipher any files that the malware should have encrypted.

7.3. Results for Axis 3: Building a secure network stack

7.3.1. Private set intersection cardinality

Participants: Jeffrey Burdges, Alvaro Garcia Recuero, Christian Grothoff.

We designed new efficient protocol for privacy-preserving signed set intersection cardinality using blinded BLS signatures over bilinear maps and demonstrated its utility in machine learning for abuse detection in decentralised online social networks. The paper was presented at DPM 2016 [21].

7.3.2. Cell tower privacy

Participants: Christian Grothoff, Neal Walfield.

We analyzed real-world mobility data based on cell tower traces, and illustrated how cell tower trace data can be used to identify patterns of life. We then used these results to predict future locations over a 24h period in 15 minute intervals with 80% accuracy [43].

7.3.3. Taler protocol improvements

Participants: Jeffrey Burdges, Florian Dold, Christian Grothoff, Marcello Stanisci.

We improved the Taler payment system protocol [13] to (1) reduce storage requirements for the exchange, which was the dominant cost, and (2) reduce security assumptions by avoiding the use of AES entirely.

We adapted the payment handshake to work even if JavaScript is disabled for the Web page, and adjusted the protocol to match discussions for future Web payment protocols from W3c. The protocol was extended with accounting functions to allow merchants to trace payments for their back office requirements. The user interface of the Taler wallet was streamlined, the wallet can finally get change, and the extension was made to work with Firefox. A public demonstrator was launched at https://demo.taler.net/.

7.4. Other research results: Information-Theoretical Quantification of Security Properties

Participants: Axel Legay, Fabrizio Biondi, Mounir Chadli, Thomas Given-Wilson.

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 information 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 information by combining this information with their prior knowledge of the system.

Armed with the produced output of the system, the attacker tries to infer information about the secret information that produced the output. The quantitative analysis we consider defines and computes how much information the attacker can expect to infer (typically measured in bits). This expected leakage 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 that don't it is imperative to be able to distinguish between the systems leaking very small amounts of secret information and systems leaking a significant amount of secret information, since only the latter are considered to pose a security vulnerability to the system.

Applied to shared-key cryptosystems, this approach allows precise reasoning about the information leakage of the secret key when the attacker knows the encoder function and information about the distribution of messages. In such scenarios, this work has generalised perfect secrecy, and so provides a more useful measure for unconditional cryptosystems (results that are safe against future advances in computing capabilities and theoretical breakthroughs in unsolved problems).

This work also explored scenarios where the attacker has less information about the cryptosystem; such as not knowing the encoder function, or not knowing the message distribution. Results here formalised that the attacker can never improve their attacks by having bad prior information, thus ensuring misinformation is always useful. Also, results show that the choice of encoder function may strengthen the cryptosystem against being learned by the attacker through observation. In particular, we showed that a well designed encoder function (represented as a matrix) has an infinitude of freedom for the attacker. Thus, the attacker cannot accurately learn all the secret information merely by observation.

There are several different scenarios where the attacker is trying to learn the secret information about the system. Here this is explored by considering what the secret information is, or equivalently, what prior knowledge the attacker has about the system.

Our new results in information leakage computation include implementing a hybrid precise-statistical computation algorithm for our QUAIL tool. The new algorithm bridges the gap between statistical and formal techniques by using static program analysis to extract structural information about the program to be analyze and decide whether each part of it would be analyzed more efficiently with precise or statistical analysis. Then each part is analyzed with the most appropriate technique, and all analyses are combined into a final result. This new hybrid method outperforms precise and statistical analysis in computation time and precision, and is a clear example of the advantages of combining precise and statistical techniques.We refer to the tools section for more details.

Additionally, we have considered how the scheduling of privileged and unprivileged processes on a shared memory could allow an unprivileged process to access confidential information temporarily stored in the memory by a privileged process. This is for instance the case in cache attacks. We have developed a general model of information leakage for scheduled systems. Our model considers a finer granularity than previous attempts on the subject, allowing us to schedule processes with small leakage, and schedule sets of processes that were considered unschedulable with no leakage by the state of the art.

[1] Preserving the privacy of private communication is a fundamental concern of computing addressed by encryption. Information-theoretic reasoning models unconditional security where the strength of

the results does not depend on computational hardness or unproven results. Usually the information leaked about the message by the ciphertext is used to measure the privacy of a communication , with perfect secrecy when the leakage is 0. However this is hard to achieve in practice. An alternative measure is the equivocation, intuitively the average number of message/key pairs that could have produced a given cipher-text. We show a theoretical bound on equivocation called max-equivocation and show that this generalizes perfect secrecy when achievable, and provides an alternative measure when perfect secrecy is not achievable. We derive bounds for max-equivocation for symmetric encoder functions and show that max-equivocation is achievable when the entropy of the ciphertext is minimized. We show that max-equivocation easily accounts for key re-use scenarios, and that large keys relative to the message perform very poorly under equivocation. We study encoders under this new perspective, deriving results on their achievable maximal equivocation and showing that some popular approaches such as Latin squares are not optimal. We show how unicity attacks can be naturally modeled, and how relaxing encoder symmetry improves equivocation. We present some algorithms for generating encryption functions that are practical and achieve 90 to 95% of the theoretical best, improving with larger message spaces.

[24] Analysis of a probabilistic system often requires to learn the joint probability distribution of its random variables. The computation of the exact distribution is usually an exhaustive precise analysis on all executions of the system. To avoid the high computational cost of such an exhaustive search, statistical analysis has been studied to efficiently obtain approximate estimates by analyzing only a small but representative subset of the system's behavior. In this paper we propose a hybrid statistical estimation method that combines precise and statistical analyses to estimate mutual information and its confidence interval. We show how to combine the analyses on different components of the system with different precision to obtain an estimate for the whole system. The new method performs weighted statistical analysis with different sample sizes over different components and dynamically finds their optimal sample sizes. Moreover it can reduce sample sizes by using prior knowledge about systems and a new abstraction-then-sampling technique based on qualitative analysis. We show the new method outperforms the state of the art in quantifying information leakage.

[12] The protection of users' data conforming to best practice and legislation is one of the main challenges in computer science. Very often, large-scale data leaks remind us that the state of the art in data privacy and anonymity is severely lacking. The complexity of modern systems make it impossible for software architect to create secure software that correctly implements privacy policies without the help of automated tools. The academic community needs to invest more effort in the formal modeling of security and anonymity properties, providing a deeper understanding of the underlying concepts and challenges and allowing the creation of automated tools to help software architects and developers. This research track provides numerous contributions to the formal modeling of security and anonymity properties and the creation of tools to verify them on large-scale software projects.

[62] High-security processes typically have to load confidential information, such as encryption keys or private data, into memory as part of their operation. In systems with a single shared memory, when high-security processes are switched out due to context switching, confidential information may remain in memory and be accessible to low-security processes. This paper considers this problem from the perspective of scheduling. A formal model supporting preemption is introduced that allows: reasoning about leakage between high-and low-security processes, and producing information-leakage aware schedulers. Several information-leakage aware heuristics are presented in the form of compositional pre-and postprocessors as part of a more general scheduling approach. The effectiveness of such heuristics is evaluated experimentally, showing them to achieve significantly better schedulability than the state of the art.