INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Project-Team CACAO

Curves, Algebra, Computer Arithmetic, and so On

Nancy - Grand Est

THEME SYM

Table of contents

1. Team

Research Scientist

Guillaume Hanrot [Team Leader, Research Director, INRIA, HdR]
Paul Zimmermann [Research Director, INRIA, HdR]
Pierrick Gaudry [Research Scientist, CNRS, HdR]
Emmanuel Thomé [Research Scientist, INRIA]
Jérémie Detrey [Research Scientist, INRIA, since September 1st]

Faculty Member

Marion Videau [assistant professor, Université Henri Poincaré ; on leave until January 2011]

Technical Staff

Philippe Théveny [ODL grant, INRIA]

PhD Student

Romain Cosset [INRIA/DGA grant, since September 1st ; defense planned in 2011]
Gaëtan Bisson [MESR grant, INPL, since September 1st ; defense planned in 2011]
Alexander Kruppa [CNRS grant, defense planned in 2009]
Damien Robert [MESR grant, UHP, defense planned in 2010]

Post-Doctoral Fellow

Nuno Franco [Auxiliar Professor, University of Evora, Portugal, until August 31st]

Administrative Assistant

Emmanuelle Deschamps

2. Overall Objectives

2.1. Introduction

The context of the research interests of the CACAO project-team goes with numbers and equations. We deal with mathematical objects of varying complexity, and strive for providing fast algorithms for manipulating them. In particular, *algebraic curves* over finite fields form a very important class of objects for our study, given their relevance to number theory and public-key cryptology.

The objectives of the CACAO project-team are along the following lines:

- Study arithmetic of curves of small genus, with a particular emphasis on applications to cryptology;
- Improve the efficiency and the reliability of arithmetics in a broad sense (i.e., the arithmetics of a wide variety of objects).

These two objectives interplay strongly. On the one hand, arithmetics are at the core of optimizing algorithms on curves, starting evidently with the arithmetic of curves themselves. On the other hand, curves can sometimes be a tool to solve some arithmetical problems as integer factorization.

To reach these objectives, we have isolated three key axes of work:

- **Algebraic Curves and Cryptology**: the main issue here is to investigate curves of small genus over finite fields (base field $\mathbb{F}_{p^n}$, for various p and n). The main tasks are to compute in the Jacobian of a given curve, to be able to check that this variety is suitable for cryptography (cardinality, smoothness test) and to solve problems in those structures (discrete logarithm). Applications go from number theory (integer factorization) to cryptography (an alternative to RSA).

- **Arithmetics**: Here, we consider algorithms dealing with multiple-precision integers, floating-point numbers, p-adic numbers and finite fields. For such basic data structures, we do not expect new algorithms with better asymptotic behavior to be discovered; however, since those are first-class objects in all our computations, any speedup is most welcome, even by a factor of 2. Since January 2007, CACAO has also been strongly involved in a project on the number field sieve (NFS), an integer factorization algorithm. We aim at developing an efficient implementation of the NFS, study its distribution, and fine-tune it in the currently "practical" range, i.e., 100-150 decimal digits.
- **Linear Algebra and Lattices**: solving large linear systems is a key point of factoring and of discrete logarithm algorithms, which we need to investigate if curves are to be applied in cryptology. Lattices are central points of the new ideas that have emerged over the very last years for several problems in computer arithmetic or discrete logarithm algorithms.

A line of research has been started in the Fall of 2008, complementing the first two axes above. Jérémie Detrey, hired as an INRIA Research Scientist, brings to the CACAO group expertise on hardware platforms such as Field Programmable Gate Arrays (FPGAs). Such devices have become a useful asset both in terms of cryptanalysis and cryptography.

2.2. Highlights of the year

The highlights for year 2008 in the CACAO project-team are:

- Two successes related to the CADO ANR project (started in January 2007), on the topic of the Number Field Sieve factoring algorithm. The CADO project now has a complete implementation of the algorithm. Furthermore, the organization of the CADO workshop in October 2008 was a real success, gathering more than 50 participants, including 30 from abroad;
- Pierrick Gaudry defended his habilitation thesis on Oct. 8th, 2008;
- the associate team ANC (Algorithms, Numbers, Computers) started with ANU (Australian National University), cf http://www.loria.fr/~zimmerma/anc.html;
- the release of GCC 4.3, which uses and thus requires the MPFR library, which is co-developped by the project-team.

3. Scientific Foundations

3.1. Scientific Foundations

3.1.1. Algebraic Curves and Cryptology

Though we are interested in algebraic curves by themselves, the applications to cryptology remain a motivation of our research, which is therefore especially focused on curves defined over finite fields.

In the mid-eighties, Koblitz [18] and Miller [20] proposed to use elliptic curves as a basis of public key cryptosystems. Indeed, the set of points on an elliptic curve is an abelian group, which is finite if the base field is a finite field. In this group, the discrete logarithm problem is thought to be difficult in general, in the sense that the best known algorithm to solve it has an exponential complexity. This has to be compared with the classical RSA algorithm, the security of which relies on the difficulty of factoring integers, but where the best known factoring algorithm has subexponential complexity. In practice, this means that the size of the parameters is much smaller for elliptic curve based cryptosystems than for classical ones.

More generally, for an algebraic curve over a finite field, there is a finite abelian group associated to it, called the Jacobian of the curve. Algebraic curves can be classified by their genus; the genus of a conic is zero and elliptic curves are curves of genus 1 (in that case, the Jacobian is isomorphic to the curve). As long as the genus is not too large, the discrete logarithm problem in the Jacobian of a curve is thought to be difficult in general, therefore one can also base cryptosystems on non-elliptic curves.

The main algorithmic tasks in relation to the use of curves in cryptography are the following:

1. Have an explicit description of the group and the group operation, as efficient as possible. The speed of ciphering and deciphering is indeed directly linked to the efficiency of the group operation.
2. Construct curves suitable for cryptographic use: the minimal requirement for the discrete logarithm to be difficult is to have a large prime factor in the group order. It is therefore necessary to compute the group order to check that property. This is what we call the *point counting task*.
3. Study the security of curve-based primitives. By this, since no general framework exists to assess that security, we mean undertake an as thorough as possible study of the security offered by those groups. The most standard way to do this is by trying to solve discrete logarithm problems in certain classes of curves.

3.1.2. Linear Algebra and Lattices

With "linear algebra and lattices", we denote two classes of problems of interest: computing vectors of the kernel of a large sparse matrix defined over a finite field, and studying algorithms to handle lattices that are given by a vector basis.

Huge linear systems are frequently encountered as last steps of "index-calculus" based algorithms for factoring or discrete logarithm computations. Those systems correspond to a particular presentation of the underlying group by generators and relations; they are thus always defined on a base ring which is $\mathbb{Z}$ modulo the exponent of the group, typically $\mathbb{Z}/2\mathbb{Z}$ in the case of factorization, $\mathbb{Z}/(q^n-1)\mathbb{Z}$ when trying to solve a discrete logarithm problem over $\mathbb{F}_{q^n}^*$. Those systems are often extremely sparse, so that specialized algorithms (Lanczós, Wiedemann) relying only on the evaluation of matrix-vector products essentially have a quadratic complexity, instead of being cubic with the classical Gaussian elimination.

The sizes of the matrices that are handled in record computations are such that they do not fit in the central memory of a single machine, even using a representation adapted to their sparse nature. Some parallelism is then required, yielding various difficulties that are different from the ones encountered in the classical linear algebra problems linked to numerical analysis. Specifically, dealing with matrices defined over finite fields precludes direct adaptation of numerical methods based on the notion of convergence and fixed-point theorems.

The second main topic is algorithmic lattice theory. Lattices are key tools in numerous problems in computer algebra, algorithmic number theory and cryptology. The typical questions one wants to solve are to find the shortest nonzero vector in a lattice and to find the closest lattice vector to a given vector. A more general concern is to find a better lattice basis than the one provided by the user; by "better" we mean that it consists of short, almost orthogonal vectors. This is a difficult problem in general, since finding the shortest nonzero vector is already NP-hard, under probabilistic reductions. In 1982, Lenstra, Lenstra, and Lovász [19] defined the notion of a LLL-reduced basis and described an algorithm to compute such a basis in polynomial time. Although not always sufficient, the LLL-reduction is sometimes enough for the application. Some stronger notions of reduction exist, such as Hermite-Korkine-Zolotarev (HKZ) reduction [16], which require exponential or super-exponential time but solve the shortest vector problem in an exact way. Schnorr [21] introduced a complete hierarchy of reductions ranging from LLL to HKZ both in quality and in complexity, the so-called k-BKZ reductions.

3.1.3. Arithmetics

We consider here the following arithmetics: integers, rational numbers, integers modulo a fixed modulus n, finite fields, floating-point numbers and p-adic numbers. We can divide those numbers in two classes: *exact numbers* (integers, rationals, modular computations or finite fields), and *inexact numbers* (floating-point and p-adic numbers).

Algorithms on integers (respectively floating-point numbers) are very similar to those on polynomials, respectively Taylor or Laurent series. The main objective in that domain is to find new algorithms that make operations on those numbers more efficient. These new algorithms may use an alternate number representation.

In the case of integers, we are interested in multiprecision arithmetic. Various algorithms are to be used, depending on the sizes of the objects, starting with the most simple "schoolbook" methods to the most advanced, asymptotically fast algorithms. The latter are often based on Fourier transforms.

The case of modular arithmetic and finite fields is the first where the representation of the elements has to be chosen carefully. Depending on the type of operations one wants to perform, one must choose between a classical representation, the Montgomery representation, a look-up table, a polynomial representation, a normal basis representation, ... Then appropriate algorithms must be chosen.

With p-adic numbers, we get the first examples of non-exact representations. In that setting, one has to keep track of the precision all along a computation. The mechanisms to handle that issue can vary: since the precision losses are not too difficult to control, one can work with a fixed global precision, or one can choose to have each element carrying its precision. Additionally, there are several choices for representing elements, in particular when dealing with algebraic extensions of the p-adics (ramified or unramified).

Last but not least, we are interested in the arithmetics of real numbers of floating-point type. Again, we have a notion of approximation. It is therefore necessary to decide of a *format* that defines a set of representable numbers. Then, when the result of an arithmetical operation on two representable numbers is not representable, one should define a way to *round* it to a meaningful representable number. The purpose of the IEEE-754 standard is to give a uniform answer to these questions in order to guarantee the reliability and portability of floating-point computations. The revised standard 754-2008 is no more restricted to the 4 basic field operations and the square root, but recommends correct rounding for some mathematical functions, and also recommends how to extend the default available formats. This leads to efficiency questions, in particular to guarantee that the result of an operation has been correctly rounded in arbitrary precision.

Within the context of integer arithmetic, we are also interested in putting the problem on its head, and notably by the study of the converse operation to integer multiplication, that is, integer factoring. Being the most competitive algorithm for this task, the Number Field Sieve algorithm comes naturally as a context where several parts of our work find a natural continuation, in all of the three axes above.

4. Application Domains

4.1. Application Domains

4.1.1. Cryptology.

The main application domain of our project-team is cryptology. Algebraic curves have taken an increasing importance in cryptology over the last ten years. Various works have shown the usability and the usefulness of elliptic curves in cryptology, standards (for instance, IEEE P1363 [17] and real-world applications (like the electronic passport).

We study the suitability of higher genus curves to cryptography (mainly hyperelliptic curves of genus two, three). In particular, we work on improving the arithmetic of those curves, on the point counting problem, and on the discrete logarithm problem.

We also have connections to cryptology through the study and development of the integer LLL algorithm, which is one of the favourite tools to cryptanalyze public-key cryptosystems. Examples are the cryptanalysis of knapsack-based cryptosystems, the cryptanalyses of some fast variants of RSA, the cryptanalyses of fast variants of signature schemes such as DSA or Elgamal, or the attacks against lattice based cryptosystems like NTRU. The use of floating-point arithmetic dramatically speeds up this algorithm, which renders the aforementioned cryptanalyses more feasible.

Finally, we are studying integer factoring algorithms which are of utmost importance for the evaluation of the security of the still widely used RSA cryptosystem. In the context of our ANR CADO grant, we are investigating the Number Field Sieve algorithm, which is the best known algorithm for factoring numbers of the kind used in practical RSA instances.

4.1.2. Computational Number Theory Systems.

We have strong ties with several computational number theory systems, and code written by members of the project-team can be found in the Magma, Pari/GP, and Sage software tools.

Magma[1] is the leading computational number theory software. It also has some features of computer algebra (algebraic geometry, polynomial system solving) but not all of what is expected of a computer algebra system. It is developed by the team of John Cannon in Sydney.

Pari/GP[2] is a computational number theory system which comes with a library which can be used to access Pari functions within a C program. It has originally been developed at the Bordeaux 1 University, and is currently maintained (and expanded) by Karim Belabas, from Bordeaux University. It is free (GPL) software. We sometimes use it for validation of our algorithms. Again, some code written by members of the project-team is incorporated into Pari.

Sage[3] is an open-source computer algebra system. Its development was initiated by William Stein (Univ. of Washington, Seattle). Instead of reinventing the wheel, Sage incorporates the most efficient open-source packages in each domain, for example SINGULAR, Pari/Gp, NTL, LINBOX, and the software tools MPFR and GMP-ECM developed by CACAO. Although quite new, there is already a strong community of active developers around SAGE. This system is a good alternative to Maple, Mathematica, and Magma to better disseminate our research in the future.

4.1.3. Arithmetics.

Another indirect transfer is the usage of MPFR in GFORTRAN (since 2004), and in GCC, up from version 4.3 (released in 2008). MPFR is currently used at compile-time, to convert expressions like $\sin(3.1416)$ into fixed-precision IEEE 754 formats, when the rounding mode can be statically determined. The MPFR library is also used by the CGAL library for computational geometry developed by the Geometrica project-team (INRIA Sophia Antipolis - Méditerranée).

5. Software

5.1. Introduction

A major part of the research done in the CACAO project-team is published within software. On the one hand, this enables everyone to check that the algorithms we develop are really efficient in practice; on the other hand, this gives other researchers — and us of course — basic software components on which they — and we — can build other applications.

5.2. MPFR

Keywords: *IEEE 754*, *arbitrary precision*, *correct rounding*, *floating-point number*.

Participants: Guillaume Hanrot, Philippe Théveny, Paul Zimmermann [contact].

MPFR is one of the main pieces of software developed by the CACAO team. Since end 2006, with the departure of Vincent Lefèvre to ENS Lyon, it has become a joint project between CACAO and the ARENAIRE project-team (INRIA Grenoble - Rhône-Alpes). MPFR is a library for computing with arbitrary precision floating-point numbers, together with well-defined semantics, and is distributed under the LGPL license. In particular, all arithmetic operations are performed according to a rounding mode provided by the user, and all results are guaranteed correct to the last bit, according to the given rounding mode.

[1] http://magma.maths.usyd.edu.au/magma/
[2] http://pari.math.u-bordeaux.fr
[3] http://sagemath.org

Several software systems use MPFR, for example: the GCC and GFORTRAN compilers; the SAGE computer algebra system; the KDE calculator Abakus by Michael Pyne; CGAL (Computational Geometry Algorithms Library) developed by the Geometrica project-team (INRIA Sophia Antipolis - Méditerranée); Gappa, by Guillaume Melquiond; Genius Math Tool and the GEL language, by Jiri Lebl; Giac/Xcas, a free computer algebra system, by Bernard Parisse; the iRRAM exact arithmetic implementation from Norbert Müller (University of Trier, Germany); the Magma computational algebra system; and the Wcalc calculator by Kyle Wheeler.

The main developments in 2008 were: the release of MPFR 2.3.1 on January, the release in March of GCC 4.3, which requires MPFR, the start in June of nightly tests using the PIPOL platform (also for MPC, with multiple platforms), and the release of MPFR 2.3.2 in September. A new release is planned in November, which will be called GNU MPFR, at the request of the Free Software Foundation. This release will also contain the dilogarithm function and a set of `printf` functions, which were developed in 2008.

All those developments were done in the context of the ODL (*Opération de Développement Logiciel*) MPtools, supported by INRIA from September 2007 to August 2009 (see http://www.loria.fr/~thevenyp/mptools.fr.html). A common retreat with the developers of MPFR and MPC was organized in September; during those two days, some plans were made for the future developments of those two libraries.

5.3. MPC

Keywords: *arbitrary precision*, *complex floating-point number*, *correct rounding*.
Participants: Philippe Théveny, Paul Zimmermann [contact].

MPC is a floating-point library for complex numbers, which is developed on top of the MPFR library, and distributed under the LGPL license. It is co-written with Andreas Enge (TANC project-team, INRIA Futurs Saclay). A complex floating-point number is represented by $x + iy$, where x and y are real floating-point numbers, represented using the MPFR library. The MPC library provides correct rounding on both the real part x and the imaginary part y of any result. MPC is used in particular in the TRIP celestial mechanics system developed at IMCCE (*Institut de Mécanique Céleste et de Calcul des Éphémérides*), and by the Magma computational number theory system.

In 2008, in the context of the MPtools project, the focus was made on making MPC more robust, and extend the list of available functions, to provide all functions of the C99 standard. For this purpose, the following functions were added: the (complex) logarithm, the trigonometric functions, the hyperbolic functions. Also, the installation of MPC is now using the `autotools` system. A new test suite was developed, which enables one to easily add new tests, and in particular all special values (NaN, infinities, zeroes) are now extensively tested using this new mechanism. A script determining the coverage of the source code by the tests has been designed. A new version, MPC 0.5, was released in September.

During the Sage Days 10 (October 10-15), Ph. Théveny wrote an `spackage` for MPC, and later on a Python interface to use MPC within Sage.

5.4. Gmp-Ecm

Participants: Pierrick Gaudry, Alexander Kruppa, Paul Zimmermann [contact].

GMP-ECM is a program to factor integers using the Elliptic Curve Method. Its efficiency comes both from the use of the GMP library, and from the implementation of state-of-the-art algorithms. GMP-ECM contains a library (LIBECM) in addition to the binary program (ECM). The binary program is distributed under GPL, while the library is distributed under LGPL, to allow its integration into other non-GPL software. For example, the Magma computational number theory software and the SAGE computer algebra system both use LIBECM.

From October 2005 to November 2008, there have been more than 9600 downloads. According to the "table of champions" maintained by Richard Brent[4], the ten largest ECM factors were found using GMP-ECM, including the current ECM record (67 digits). GMP-ECM is used by many mathematicians and computer scientists to factor integers.

[4] http://wwwmaths.anu.edu.au/~brent/ftp/champs.txt

In 2008, GMP-ECM 6.2 and 6.2.1 have been released, featuring a new algorithm by Alexander Kruppa and Peter Montgomery for the so-called Phase 2 of the $P+1$ and $P-1$ algorithms.

5.5. Local fields

Participant: Emmanuel Thomé [contact].

Mploc is a `C` library for computing in p-adic fields and their unramified extensions. The focus is mainly on $\mathbb{Z}_p$ for prime p, and unramified extensions of $\mathbb{Z}_2$. The ability to compute in these structures is important to several applications, such as point counting or building curves with a prescribed number of points.

The Mploc library is already distributed[5] and used, although several performance improvements are sought. The library presently gathers 8,000 lines of `C` source code.

5.6. Finite fields

Participants: Pierrick Gaudry, Emmanuel Thomé [contact].

`mp`$\mathbb{F}_q$ is (yet another) library for computing in finite fields. The purpose of `mp`$\mathbb{F}_q$ is not to provide a software layer for accessing finite fields determined at runtime within a computer algebra system like Magma, but rather to give a very efficient, optimized code for computing in finite fields precisely known at *compile time*. `mp`$\mathbb{F}_q$ is not restricted to a finite field in particular, and can adapt to finite fields of any characteristic and any extension degree. However, one of the targets being the use in cryptology, `mp`$\mathbb{F}_q$ somehow focuses on prime fields and on fields of characteristic two.

`mp`$\mathbb{F}_q$'s ability to generate specialized code for desired finite fields differentiates this library from its competitors. The performance achieved is far superior. For example, `mp`$\mathbb{F}_q$ can be readily used to assess the throughput of an efficient software implementation of a given cryptosystem. Such an evaluation is the purpose of the "EBats" benchmarking tool[6]. `mp`$\mathbb{F}_q$ entered this trend in 2007, establishing reference marks for fast elliptic curve cryptography: the authors improved over the fastest examples of key-sharing software in genus 1 and 2, both over binary fields and prime fields. These timings are now comparison references for other implementations [22].

The library's purpose being the *generation* of code rather than its execution, the working core of `mp`$\mathbb{F}_q$ consists of roughly 5,000 lines of Perl code, which generate most of the currently 13,000 lines of `C` code. `mp`$\mathbb{F}_q$ is distributed from http://mpfq.gforge.inria.fr/.

5.7. Polynomial arithmetic in characteristic 2

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

Gf2x is a set of programs for polynomial multiplication over the binary field, developed together with Richard Brent (Australian National University, Canberra, Australia). There are implementations of various algorithms corresponding to different degrees of the input polynomials. In the case of polynomials that fit into one or two machine-words, the schoolbook algorithm has been improved and implemented using SSE instructions for maximum speed. For small degrees, we switch to Karatsuba's algorithm and then to Toom-Cook's algorithm. These have been implemented using the most recent improvements. Finally, for very large degrees one has to switch to Fourier-transform based algorithms, namely Schönhage's or Cantor's algorithm. In order to choose between these two asymptotically fast algorithms, we have implemented and compared them. The GF2X package is distributed and maintained. It is available from http://wwwmaths.anu.edu.au/~brent/gf2x.html. Integration of this work within a more general software tool like NTL or Sage is a natural extension to the lifecycle of the GF2X package. To this end, latest updates to the GF2X package aim at making this inclusion easy. An article describing our improvements to the algorithms and their implementation has been presented at the ANTS VIII conference, see [7]. This work was part of the ANC associate team (see below).

[5] http://www.loria.fr/~thome/software/mploc
[6] http://www.ecrypt.eu.org/ebats/

5.8. MPQS

Participant: Paul Zimmermann.

MPQS is a program that factors integers using the Multiple Polynomial Quadratic Sieve, developed by Scott Contini and Paul Zimmermann. It is distributed under GPL from http://www.loria.fr/~zimmerma/software/, and now available within Maple (up from version 12), according to a license with Waterloo Maple Software.

6. New Results

6.1. Floating-Point Arithmetic

Participants: Guillaume Hanrot, Philippe Théveny, Paul Zimmermann.

In addition to the results mentioned above concerning MPFR, the revision of the IEEE-754 standard was finally completed in 2008 [14]. This revision incorporates in particular some recommendations for extended and extendable precisions (Section 3.7) and correctly rounded functions (Section 9.2), which are partly due to our work in collaboration with the Arenaire project-team.

Following an initial idea of Siegfried Rump, P. Zimmermann worked with him, Sylvie Boldo and Guillaume Melquiond on new portable and efficient algorithms to compute the predecessor and successor — `nextbelow` and `nextafter` in the IEEE-754 naming — in rounding to nearest [11].

6.2. Exact arithmetic

Participants: Pierrick Gaudry, Guillaume Hanrot, Alexander Kruppa, Emmanuel Thomé, Paul Zimmermann.

Gaudry, Thomé, Zimmermann, together with Richard Brent (Australian National University, Canberra) have worked on the multiplication of binary polynomials using a variety of techniques, from low-level optimizations to advanced asymptotically fast FFT algorithms. The outcome is a consistent improvement over existing software libraries, with for example a 10-fold speed-up over NTL-5.4.2, for multiplication of polynomials of 2^{28} bits. This work has been published in [7], and the corresponding source code is distributed under the GPL License[7].

Richard Brent and P. Zimmermann are collaborating on a book called "Modern Computer Arithmetic". A new version [1] has been published in 2008, in the context of the INRIA associate team[8] which started in 2008.

Another common project with Richard Brent is the search for primitive trinomials over $\mathbb{F}_2$. The search for primitive trinomials corresponding to huge Mersenne primes continued. For degree 30402457, we have found exactly one primitive trinomial (and its reciprocal):

$$x^{30402457} + x^{2162059} + 1,$$

and for degree 32582657, we have found exactly three:

$$x^{32582657} + x^{5110722} + 1, x^{32582657} + x^{5552421} + 1, x^{32582657} + x^{7545455} + 1.$$

All those primitive trinomials have been checked by Allan Steel using Magma. The paper [2] describing in detail the new algorithm has been published in a special issue of *Contemporary Mathematics*, and the computational results will appear in [3].

[7] http://wwwmaths.anu.edu.au/~brent/gf2x.html
[8] http://www.loria.fr/~zimmerma/anc.html

A collaboration with Jean-Louis Nicolas and Marc Deléglise (Univ. Lyon 1) on the computation of Landau's function finally resulted in a common publication [4].

6.3. Results on Lattices and Linear Algebra

Participant: Guillaume Hanrot.

Last year, Hanrot and Stehlé (ARENAIRE project-team, INRIA Grenoble - Rhône-Alpes) completed an analysis of Kannan's enumeration algorithm, which is the best deterministic algorithm for finding a shortest non-zero vector in a lattice, or a closest vector to a given point. They proved in particular that the complexity of the former problem is at most $d^{d/(2e)+o(d)}$ arithmetic operations on integers of polynomial size. The newer result is the proof that this complexity is also a *lower bound* [10].

6.4. Curve-related results

Participants: Gaëtan Bisson, Romain Cosset, Pierrick Gaudry, Guillaume Hanrot, Damien Robert, Emmanuel Thomé.

6.4.1. Curves for Factoring

Romain Cosset, as part of his M2 internship, and beginning of his PhD thesis, has worked on developing a genus-2 "hyperelliptic curve method" for integer factoring, as an extension to the well-known elliptic curve method. Using a wide variety of ideas, the new genus-2 method shows many very interesting traits, and as it turns out, is pretty likely to be a faster alternative to ECM (the comparison reference for the ECM algorithm being the state-of-the-art GMP-ECM implementation, developed within the project-team). A paper detailing this work is in preparation.

The collaboration started in 2007 between Alexander Kruppa and Peter Montgomery led to a new algorithm for the so-called Phase 2 of the $P+1$ and $P-1$ factoring algorithms. Given an element of $\mathbb{Z}/N\mathbb{Z}$ (or for P+1 of a quadratic extension) of not-too-large order modulo a prime factor p of N, it tries to discover p by evaluating polynomials at many points looking for a value $\equiv 0 \pmod{p}$. The new algorithm improves on previous work by reducing the cost of constructing the polynomials of degree d from $O(d(\log d)^2)$ to $O(d \log d)$, and using reciprocal Laurent polynomials to reduce memory use and cost of multiplication (via weighted transforms) by a factor of two. The implementation scales well on multi-processor machines. For the P+1 method a new record factor of 60 decimal digits was found. The article describing this algorithm was presented at the ANTS VIII conference and is published in [9].

6.4.2. Curves for Cryptography

Pierrick Gaudry and David Lubicz have worked on effective formulae for the group law of genus 2 curves. A complete proof of very efficient formulae for the Kummer surface in characteristic 2 was left open. Using the theory of algebraic Theta functions, this difficulty was overcome. A journal article has been submitted and is currently under revision [12].

During the visit of É. Schost in May-June, Schost and Gaudry achieved the computation of the cardinality of the Jacobian of a genus 2 curve over the field $\mathbb{F}_{2^{127}-1}$. This required about a month of computational effort, and could not have been done without the introduction of several new algorithms. In particular, higher-order 2^k-torsion and 3^k-torsion were successfully exploited. This work was presented as an invited talk at the ECC 2008 conference.

Subsequently to his Master's thesis work, Gaëtan Bisson has been working with Takakazu Satoh (Tokyo Institute of Technology) on improving pairing-friendly elliptic curves generation techniques; more specifically, they enabled known techniques to generate curves with discriminants larger than initially intended. They explain this enhancement in a paper that has been accepted at Indocrypt 2008 [6].

Damien Robert and David Lubicz have worked on explicit isogeny computation in genus 2. They have designed an algorithm, similar to the so-called Vélu's formulae for elliptic curves, that constructs an abelian variety isogenous to a given Jacobian of a curve of genus 2. The key tool is the use of Theta functions. An article will be written in 2009 describing this new technique.

6.4.3. Discrete Logarithm on Curves

The paper by Diem and Thomé, improving the complexity of the computation of discrete logarithms in jacobians of non-hyperelliptic curves, has finally been published [5].

Another contribution in the context of discrete logarithms has been obtained by Enge, Gaudry, and Thomé, following previous work by Enge and Gaudry. For a general curve of large enough genus g over a finite field q, the complexity of a discrete log computation is in $L_{q^g}(1/2)$, where $L()$ is the classical subexponential function. In 2007, Enge and Gaudry have shown that for plane curves having a particular shape of degrees in x and y, this complexity can been reduced heuristically to $L_{q^g}(1/3+\varepsilon)$, recovering the kind of complexity we have for integer factorization or discrete logarithms in finite fields. The newer work achieves the removal of this ε in the complexity, and broadens the spectrum of the attack.

6.5. Number Field Sieve-related results

Participants: Pierrick Gaudry, Alexander Kruppa, Emmanuel Thomé, Paul Zimmermann.

The CADO code base for integer factorization has reached an almost mature state, and has been used to perform several factorizations, in particular cofactors from the BMtR (Brent-Montgomery-te Riele) table, which consists of numbers of the form $a^n \pm 1$, with $13 \leq a \leq 99$. The largest CADO-NFS factorization as of November 2008 is a 148-digit number from aliquot sequence 1074.

Recent work includes an implementation by Alexander Kruppa of the P-1, P+1 and ECM factoring algorithms optimized for small input numbers as occur in the refactoring phase of NFS. To allow more than two "large primes" not exceeding L in the norms of relations, composites with three (or more) primes up to L must be factored quickly and composites with any primes greater than L discarded as soon as possible. P-1, P+1 and ECM with carefully chosen parameters and an early-abort strategy were found to perform well for this task. The siever is being modified to produce more accurate size estimates, so that fewer candidates enter the refactoring phase, which would otherwise become a bottleneck. Selection of near-optimal early-abort strategies is to be investigated, the results are planned to be submitted as a joint paper with Thorsten Kleinjung (EPFL) to Mathematics of Computation.

Following a previous line of research started in a 2007 paper by Joux, Naccache and Thomé, the same authors, together with Lercier, showed that the *static* Diffie-Hellman problem lends itself well to a Number Field Sieve-type attack whose complexity is far lower than that of the previously best known method. In particular, a static Diffie-Hellman challenge over the multiplicative group of the field $\mathbb{F}_{2^{1025}}$ has been broken. This work exists at the moment as a preprint [13].

6.6. Boolean functions for cryptography

In the context of the ANR project RAPIDE, Marion Videau and Cédric Lauradoux (who was postdoc at the University of Princeton at that time) studied a new family of Boolean functions that has good implementation properties. The new family, called *matriochka symmetric* is derived from the family of symmetric Boolean functions well known for being the only family with a linear gate complexity. The results concerning the new construction, the gate complexity bounds and the Walsh spectrum study for functions of small degrees have been presented in [8] during the conference ISIT2008 (2008 IEEE International Symposium on Information Theory, Toronto, Canada).

7. Other Grants and Activities

7.1. National Initiatives

7.1.1. ANR CADO (Crible algébrique, Distribution, Optimisation)

Participants: Pierrick Gaudry, Guillaume Hanrot, Alexander Kruppa, Emmanuel Thomé, Paul Zimmermann.

The team has obtained a financial support from the ANR ("programme blanc") for a project, common with the TANC project-team and the number theory team of the mathematics lab in Nancy (IECN). Its objective is to study the number field sieve algorithm. This grant has been running since January 2007, and will continue until the end of 2009.

We are working on several aspects of this factoring algorithm, that are linked to our main objectives. Among other things, we investigate the so-called "polynomial selection" phase, which could possibly be improved using some lattice reduction tools, we work on the parallelization (in a Grid context) of the linear algebra step, we also study the relation search phase, where the speed of the underlying arithmetic is crucial.

For all of that, it is important to us to have our own implementation. Therefore, our primary task during the first year was the development of a complete implementation of the Number Field Sieve, which contains a large number of sub-algorithms. The implementation is complete, but leaves still many places to be improved. Compared to existing implementations, the CADO implementation is already a reasonable player. Several factorizations have been completed using our implementations.

7.1.2. ANR RAPIDE (Design and analysis of stream ciphers dedicated to constrained environments)

Participants: Marion Videau, Guillaume Hanrot.

The project from "programme Sécurity Et Informatique 2006" involves the team together with SECRET (former CODES) project-team, XLIM lab from the university of Limoges and the CITI lab from INSA-Lyon. It has been running since January 2007 and will continue until the end of 2010.

The research project consists in the study and analysis, both from theoretical and practical points of view, of existing stream ciphers and new designs based on non-linear feedback shift registers.

Despite the departure of Marion Videau (on secondment to the cryptographic lab of the Central Information System Security Division), the coordination tasks are held by her from the team side.

7.2. International Initiatives

7.2.1. Collaboration with ANU

Participants: Joerg Arndt, Alexander Kruppa, Paul Leopardi, Richard Brent, Paul Zimmermann.

In the context of the "associate team" ANC (Algorithms, Numbers, Computers), which started in 2008 (http://www.loria.fr/~zimmerma/anc.html), between the CACAO project-team and the team of Richard Brent at the Australian National University (ANU), several visits were organized in 2008: J. Arndt (ANU) visited LORIA for one month in May, A. Kruppa and P. Zimmermann visited ANU for two weeks in June, P. Leopardi and R. Brent (ANU) visited LORIA for two weeks in October, where they attended the CADO-NFS and the Sage Days 10 workshops.

7.2.2. Collaboration with Tsukuba, Japan

Participants: Gaëtan Bisson, Guillaume Hanrot, Damien Robert.

G. Hanrot was an invited researcher in the Laboratory of Cryptography and Information Security (LCIS) of the University of Tsukuba, Japan, in February, for a 3 weeks duration. This visit was fruitful and will lead to the submission of an article to the Pairing conference in 2009.

Later this year, in the context of the AYAME Junior Program on the subject of "Software and Hardware Components for Pairing-Based Cryptography" between the CACAO project-team, the Arenaire Project-Team and the Laboratory of Cryptography and Information Security (LCIS) of the University of Tsukuba, Japan, G. Bisson and D. Robert also visited Tsukuba for three weeks in October-November 2008.

7.2.3. Other visits

Éric Schost from University of Western Ontario visited us in May-June, working mainly with P. Gaudry on efficient point counting in Jacobians of genus 2 curves.

Andreas Enge from LIX visited us in July, working with P. Gaudry and E. Thomé on paper [15]. The final, greatly improved version of this work will appear in Journal of Cryptology.

Alexander Kruppa visited Arjen Lenstra's group in Lausanne for four weeks in September. He met Lenstra's students and learned about their current research, in particular on using the Cell processor in the Sony Playstation 3 for high-performance computing in cryptography. He also discussed re-factoring in NFS sieving with Thorsten Kleinjung, and the two agreed on publishing a joint paper on the subject.

8. Dissemination

8.1. Scientific Animation

8.1.1. CACAO seminar

We have a seminar, where we have invited in 2007 the following speakers: Nicolas Julien, Aurélie Bauer, Guillaume Melquiond, Laurent Imbert, Nicolas Meloni, Mathieu Cluzeau, Joerg Arndt, Éric Schost, Marc Mezzarobba, Nicolas Estibals.

8.1.2. Conference organization

Emmanuel Thomé has co-organized the *Journées Nationales de calcul Formel*, that took place in Luminy in October.

In the context of the CADO ANR Grant, Pierrick Gaudry and Emmanuel Thomé have co-organized the CADO workshop which took place at LORIA on October 7,8,9th. This workshop was a real success and attracted more than 50 participants, including more than 30 from foreign countries. The CADO workshop hosted invited talks from Kazumaro Aoki, Dan Bernstein, Thorsten Kleinjung, Reynald Lercier, Peter Montgomery.

Paul Zimmermann has organized the Sage Days 10 in Nancy from October 10 to 15, see http://wiki.sagemath.org/days10. This workshop consisted of 3 "talk days" with invited and contributed talks, and 3 days of "coding sprints", where the participants discussed on the code in small groups. This workshop was a real success too, and attracted more than 70 participants, among them more than 30 students. This was by far the largest "Sage Days" edition so far.

8.2. Committees memberships

From Sept. 2008, G. Hanrot has been scientific delegate of INRIA-Nancy Grand Est. Before this date, he was scientific vice-delegate. From Oct. 2008, he has been vice-president of INRIA Evaluation Committee. He was a member of the hiring committee for CR2 at INRIA Rocquencourt in 2008. He was referee for the habilitation of David Lubicz (Univ. Rennes 1, external reviewer), and the PhD thesis of Nicolas Gama (École Normale Supérieure, external reviewer).

P. Gaudry is an appointed member of the Computer Science "Commissions de Spécialistes" from École normale supérieure. He was referee for the PhD theses of D. Mireles Morales (Royal Holloway University of London) and of M. Wagner (Technische Universtität Berlin). He was a PC member of the First international conference on Symbolic Computation and Cryptography (held in Beijing, China) and is a PC member of the CHiLE conference (to be held in Utalca, Chile in 2009) and of the Pairing conference (to be held in Stanford, USA in 2009).

P. Zimmermann is member of the program committee of the Arith'19 conference, to be held in Portland (Oregon) in 2009. He was member of the program committee of the Sage Days 10. He was member of the habilitation thesis (HDR) jury of Laurent Imbert (Univ. Montpellier), and of Pierrick Gaudry (Univ. Henri Poincaré Nancy 1). He was a member of the INRIA hiring committee for CR2 (Saclay - Île-de-France) and DR2.

8.3. Invited Conferences

P. Gaudry gave a one-hour invited talk for the mini-workshop "Algebraic Curves and Cryptography" in Ulm, Germany and a half-an-hour invited talk for the third "Franco-Japanese Computer Security Workshop" at LORIA.

G. Hanrot gave a one-hour invited talk on "Enumeration problems for lattices" at the mini-workshop of the Lareda ANR project.

G. Hanrot gave a three hours lecture on "Number theory and cryptography" at the First MITACS-INRIA Workshop on Foundations and Practice of Security in June in Montreal, Canada.

P. Zimmermann gave an invited talk at the Dagstuhl seminar "Numerical Validation in Current Hardware Architectures" in January, and at the "Central European Conference on Cryptography" in Graz (Austria) in July.

8.4. Teaching

E. Thomé gave three 3 hours lectures at MPRI (Master Parisien de Recherche en Informatique) about algorithmic number theory, in the Cryptology course.

E. Thomé supervised the Master 2 internship of Romain Cosset from MPRI (Master Parisien de Recherche en Informatique), on the topic of factorization with hyperelliptic curves (February-August).

E. Thomé gave 6 hours of Master 2 courses at Université Henri Poincaré on the topic of cryptology and computer networks.

P. Gaudry gave 15 hours of Master 1 courses at Université Henri Poincaré on the topic of cryptology.

P. Gaudry gave 3 hours of adult professional courses at CFSSI (*Centre de Formation en Sécurité des Systèmes d'Information*) on the topic of elliptic curves during a cryptology session.

P. Gaudry and G. Hanrot are members of the jury of "agrégation externe de mathématiques", a competitive exam to hire high school teachers.

G. Hanrot gave a four hours invited lecture on "Fast arithmetic for cryptography" at the Fall School "Referentiels de la cryptographie moderne" in Rabat, Morocco.

E. Thomé is a member of the jury of the competitive exam for the École polytechnique.

E. Thomé and P. Zimmermann supervised the L3-level internship of Élie de Panafieu from ENS Cachan, on the topic of polynomial selection in the Number Field Sieve algorithm (June-July).

M. Videau gave 3 hours of adult professional courses at CFSSI (*Centre de Formation en Sécurité des Systèmes d'Information*) on the topic of cryptology during an information security session.

M. Videau gave 6 hours of adult professional courses at CFSSI on the topic of algorithmic for cryptology and cryptanalysis of block ciphers during a cryptology session.

M. Videau gave 9 hours of adult professional courses at CFSSI (for a degree equivalent to Master 2) on the topic of stream ciphers, block ciphers and modes of operation.

M. Videau gave 12 hours of Master 1 level courses at ESIAL (engineering school of Université Henri Poincaré) on the topic of cryptology.

9. Bibliography

Major publications by the team in recent years

[1] R. P. BRENT, P. ZIMMERMANN. *Modern Computer Arithmetic*, In preparation, Version 0.2, 2008, http://wwwmaths.anu.edu.au/~brent/pub/pub226.html.

Year Publications

Articles in International Peer-Reviewed Journal

[2] R. P. BRENT, P. ZIMMERMANN. *A Multi-level Blocking Distinct Degree Factorization Algorithm*, in "Contemporary Mathematics", vol. 461, 2008, p. 47-58, http://hal.inria.fr/inria-00181029/en/.

[3] R. P. BRENT, P. ZIMMERMANN. *Ten new primitive binary trinomials*, in "Mathematics of Computation", 2008, http://hal.inria.fr/inria-00337525/en/.

[4] M. DELÉGLISE, J.-L. NICOLAS, P. ZIMMERMANN. *Landau's function for one million billions*, in "Journal de Théorie des Nombres de Bordeaux", 2009, http://hal.archives-ouvertes.fr/hal-00264057/en/.

[5] C. DIEM, E. THOMÉ. *Index calculus in class groups of non-hyperelliptic curves of genus three*, in "Journal of Cryptology", vol. 21, 2008, p. 593-611, http://hal.inria.fr/inria-00107290/en/.

International Peer-Reviewed Conference/Proceedings

[6] G. BISSON, T. SATOH. *More Discriminants with the Brezing-Weng Method*, in "9th International Conference on Cryptology in India - INDOCRYPT 2008, Inde Khargapur", 2008, http://hal.inria.fr/inria-00337358/en/.

[7] R. P. BRENT, P. GAUDRY, E. THOMÉ, P. ZIMMERMANN. *Faster Multiplication in GF(2)[x]*, in "ANTS-VIII Algorithmic Number Theory, ANTS-VIII Lecture notes in computer science, Canada Banff", A. J. VAN DER POORTEN, A. STEIN (editors), vol. 5011, Springer-Verlag, 2008, p. 153-166, http://hal.inria.fr/inria-00188261/en/.

[8] C. LAURADOUX, M. VIDEAU. *Matriochka symmetric Boolean functions*, in "IEEE International Symposium on Information Theory - ISIT 2008, Canada Toronto", IEEE, 2008, p. 1631-1635, http://hal.inria.fr/inria-00338085/en/.

[9] P. L. MONTGOMERY, A. KRUPPA. *Improved Stage 2 to P±1 Factoring Algorithms*, in "8th International Symposium on Algorithmic Number Theory - ANTS-VIII Algorithmic Number Theory Lecture Notes in Computer Science, Canada Waterloo", A. J. VAN DER POORTEN, A. STEIN (editors), vol. 5011, Springer, 2008, p. 180-195, http://hal.inria.fr/inria-00188192/en/.

Research Reports

[10] G. HANROT, D. STEHLÉ. *Worst-Case Hermite-Korkine-Zolotarev Reduced Lattice Bases*, RR-6422, Rapport de recherche, 2008, http://hal.inria.fr/inria-00211875/en/.

[11] S. RUMP, P. ZIMMERMANN, S. BOLDO, G. MELQUIOND. *Computing predecessor and successor in rounding to nearest*, Rapport de recherche, 2008, http://hal.inria.fr/inria-00337537/en/.

Other Publications

[12] P. GAUDRY, D. LUBICZ. *The arithmetic of characteristic 2 Kummer surfaces*, 2008, http://hal.inria.fr/inria-00266565/en/.

[13] A. JOUX, R. LERCIER, D. NACCACHE, E. THOMÉ. *Oracle-Assisted Static Diffie-Hellman Is Easier Than Discrete Logarithms*, 2008, http://hal.inria.fr/inria-00337753/en/.

References in notes

[14] *IEEE Standard for Floating-Point Arithmetic*, Revision of ANSI-IEEE Standard 754-1985, approved June 12, 2008: IEEE Standards Board, Technical report, n^{o} ANSI-IEEE Standard 754-2008, 2008.

[15] A. ENGE, P. GAUDRY. *An $L(1/3+\varepsilon)$ Algorithm for the Discrete Logarithm Problem for Low Degree Curves*, in "Eurocrypt 2007 Advances in Cryptology - EUROCRYPT 2007 Lecture Notes in Computer Science, Barcelona Espagne", M. NAOR (editor), Lecture Notes in Computer Science, vol. 4515, Springer, 2007, p. 379-393, http://hal.inria.fr/inria-00135324/en/.

[16] C. HERMITE. *Extraits de lettres de M. Hermite à M. Jacobi sur différents objets de la théorie des nombres, deuxième lettre*, in "Journal für die reine und angewandte Mathematik", vol. 40, 1850, p. 279–290.

[17] IEEE. *P1363: Standard specifications for public key cryptography*.

[18] N. KOBLITZ. *Elliptic curve cryptosystems*, in "Math. Comp.", n^{o} 48, 1987, p. 203–209.

[19] A. K. LENSTRA, H. W. LENSTRA, L. LOVÁSZ. *Factoring Polynomials with Rational Coefficients*, in "Mathematische Annalen", vol. 261, 1982, p. 515–534.

[20] V. S. MILLER. *Use of Elliptic Curves in Cryptography*, in "Advances in cryptology—CRYPTO 85, New York, USA", Lecture notes in computer science, vol. 218, Springer-Verlag, 1986, p. 417–426.

[21] C. P. SCHNORR. *A Hierarchy of Polynomial Lattice Basis Reduction Algorithms*, in "Theoretical Computer Science", vol. 53, 1987, p. 201–224.

[22] M. SCOTT. *New record breaking implementations of ECC on quadratic extensions using endomorphisms*, Invited talk at the ECC 2008 Conference. Utrecht, the Netherlands, Sep. 22-24, 2008., September 2008.