RESEARCH STATEMENT

My research interests lie in the field of Computational Number Theory, Cryptography, Information Security and Algorithms for finite field arithmetic. As a long-term research direction after I complete the doctoral thesis in mathematics, I want to continue doing research in Cryptography and Information Security. I am very interested in some topics related to cryptographic hard problems, for example discrete logarithms, factoring, and elliptic curves.

In joint work with Alexis Bonnecaze and Stéphane Ballet during my PhD study at Institute of Mathematics of Marseille (France), I have worked on the topic "Efficient arithmetic in finite fields based on algebraic curves".

1. Summary of research in Ph.D thesis

In order to keep pace with the development of digital technology and to meet the growing need for security, it is becoming increasingly important to be able to manage digital data quickly. These data are sometimes large numbers and it is essential that the basic operations on these large numbers are optimized. Arithmetic operations such as additions, multiplications and exponentiations are an important part of these basic operations.

For example, public-key cryptography uses these operations on numbers up to several thousand bits in size. Public-key algorithms are most often based on the computational complexity of "hard" problems, often from Number Theory. For instance, the hardness of RSA is related to the *integer factorization problem*, while Diffie–Hellman and DSA are related to the *discrete logarithm problem*. The security of elliptic curve cryptography is based on number theoretic problems involving *elliptic curves*. Recently, pairing-based cryptography is based on pairing functions that map pairs of points on an elliptic curve into a finite field, have been a very active area of research in cryptography. Pairings are useful in cryptography because if constructed properly, they can produce finite fields that are large enough to make the discrete logarithm problem hard to compute, but small enough to make computations efficient. Because of the difficulty of the underlying problems, most public-key algorithms involve operations such as *modular multiplication and exponentiation*. The efficiency of these algorithms (and the protocols that use them) depends on how quickly these operations are carried out.

In my thesis, I am interested in the *efficiency of multiplication in a finite field*. Let q be a power of a prime number p and n a positive integer, the multiplication of two elements of $\mathbb{F}_{q^n}$ can be seen as a multiplication of two polynomials f and g of degree $n-1$ with coefficients in $\mathbb{F}_q$ followed by a modular reduction. This reduction is done with the irreducible polynomial of degree n used to define the field extension. The classical method to multiply two polynomials of degree n costs $O(n^2)$ operations and a method due to Karatsuba allows the cost to be reduced to $O(n^{1.59})$. In many practical cases, Karatsuba's method is sufficient, but when the elements to be multiplied belong to a large field, it is imperative to use more efficient algorithms. Currently, the fastest algorithms are derived from the Discrete Fourier Transform and its efficient implementation the Fast Fourier Transform (FFT).

With FFT, the multiplication of f and g can be computed with $O(n \log n)$ operations. However, this method works only when the algebraic structure (FFT is generally defined over a ring) contains special roots of unity. Thus, FFT cannot be applied with just any ring. Schonhage-Strassen's algorithm solves the problem by using "virtual" roots of unity for a cost of $O(n \log n \log \log n)$. All multiplication algorithms perform addition, scalar multiplication and bilinear multiplication. These operations have different costs. The bilinear multiplication (i.e. multiplying two elements of $\mathbb{F}_q$ which depend on f and g, the elements of $\mathbb{F}_{q^n}$ being multiplied) is heavier than the scalar multiplication (i.e. multiplying a constant of $\mathbb{F}_q$ by an element of $\mathbb{F}_q$ which depends on the elements being multiplied) which is itself heavier than the addition. Thus, algorithms with low bilinear complexity are particularly interesting.

In 1988, generalizing interpolation algorithms on the projective line over $\mathbb{F}_q$ to algebraic curves of higher genus over $\mathbb{F}_q$, D.V. and G.V. Chudnovsky provided a method [5] which enabled to prove the *linearity* [2] of the bilinear complexity of multiplication in finite extensions of a finite field. This is the so-called Chudnovsky-Chudnovsky multiplication algorithm (or CCMA). This is an interpolation method based on algebraic curves which can be roughly explained as follows. The elements to be multiplied are seen as the evaluation of some functions f and g of a Riemann-Roch space of a divisor D on a certain point (also called a place) Q. Since this evaluation is designed to be an isomorphism, multiplying two elements of the field $\mathbb{F}_{q^n}$ is the same as multiplying two elements of the Riemann-Roch space. A second evaluation function evaluates f and g on enough points P_i so that, by interpolation, the product fg can be calculated. This second evaluation function can be represented by a matrix denoted T_D in the thesis. This is the first phase of the algorithm. The second phase, which involves an other matrix denoted by T_{2D}^{-1}, consists of transforming fg into an element of the field $\mathbb{F}_{q^n}$. Indeed, both matrices are linked and their coefficients depend on the choice of different parameters (the place Q, the Riemann-Roch space, the points used on the second evaluation, etc.). As with any interpolation, the number of points on which evaluations can be made is particularly important. To increase this number, it is possible either to increase the genus or to increase the degree of the points. It is also possible to use derivated evaluations.

Many studies, including recently [4], focused on the qualitative improvement of CCMA, but very little is known about the scalar complexity of this method and therefore its overall complexity. The problem of its scalar complexity was only addressed in 2015 by Atighehchi, Ballet, Bonnecaze and Rolland [1]. They proposed a new construction which slightly improved the scalar complexity even though the main objective of this work was not to optimize scalar complexity. The objective of their article was mainly to compare the total complexity of an algorithm due to Couveignes and Lercier [6] with that of a method using parallel calculations on an improvement of CCMA. Thus, there was no strategy dedicated to scalar optimization and in fact, the number of scalar multiplications was not significantly reduced in finite distance. Therefore, it has to be noted that so far, practical implementations of multiplication algorithms of type Chudnovsky over finite fields have failed to simultaneously optimize the number of scalar multiplications and bilinear multiplications.

It is known that Chudnovsky's method is competitive in terms of bilinear complexity but what about its total complexity?

The objective of my research is to *analyse this scalar complexity and possibly find*

Chudnowsky-type algorithms with an improved scalar complexity. It is easy to see that the scalar complexity depends on the coefficients of the two matrices involved in the method since each matrix will have to be multiplied by a vector which depends on the elements to multiply. If a coefficient is equal to 0 or 1 the multiplication (in $\mathbb{F}_q$) does not have to be done. In fact $0 \cdot x = 0$ and $1 \cdot x = x$. Thus, in order to control the number of scalar multiplications, the number of 0 and 1 in each matrix should be controlled. As a first step, we decided to focus on the number of zeros. This means that we want to make the two matrices as sparse as possible. If we denote by $N_{zero}(T_D)$ and $N_{zero}(T_{2D,n}^{-1})$ respectively the number of zeros in the first matrix and the number of zeros in the second one (actually, we just need to consider the first n rows of the matrix), we seek to maximize

$$N_z = 2N_{zero}(T_D) + N_{zero}(T_{2D,n}^{-1}).$$

Since these matrices are linked, the best value of $N_{zero}(T_D)$ may not lead to the best solution.

The first task was to define a strategy to optimize N_z. We chose to fix the divisor D as well as the points Q and P_i and to determine the set of bases of the Riemann-Roch space which lead to the best value $N_{zero}(T_D)$. We note that fixing a basis of the Riemann-Roch space also fixes the basis of F_{q^n}. This strategy is generic in the sense that when the divisor and points are fixed, there is no other possible strategy. The obtained results in thesis have been published in [3] (2019).

By an observation on connection between CCMA and an algebraic geometry code (Goppa code), I obtained an upper-bound on $N_{zero}(T_D)$:

$$N_{zero}(T_D) \leq n \cdot \deg D.$$

Among the best bases of the Riemann-Roch space $\mathcal{L}(D)$, we searched which ones gave the best value of $N_{zero}(T_{2D,n}^{-1})$. The theoretical and numerical results obtained, which depend only on the degree of D, suggest that this optimization strategy is also independent of the choice of the divisor D. Consequently, in my thesis I proposed an improved computationally-efficient algorithm for optimization of the scalar complexity of CCMA. We used our strategy to improve the scalar complexity of the Baum-Shokrollahi construction on $\mathbb{F}_{256}/\mathbb{F}_4$. We obtained $N_z = 52$ which corresponds to a gain of 27% over Baum-Shokrollahi's method. Moreover, for this field, our construction is better than any other known algorithm.

Besides, I also mentioned other strategies for optimization of the scalar complexity of CCMA. A specific case has been considered, that is the order of evaluation points P_i on the algebraic curve $\mathscr{C}$ over $\mathbb{F}_q$ constructed in CCMA have been permuted.

We have submitted all these results in [10] (2020).

2. Future Research Direction

As a specific case of multiplication, exponentiation in finite fields is an important operation. Especially, modular exponentiation which is a type of exponentiation performed over a modulus, is useful in the field of public-key cryptography. It is known that exponentiation in the finite field extension $\mathbb{F}_{q^n}/\mathbb{F}_q$ has a computational advantage of *normal basis.* In particular, they allow fast exponentiation by q since it corresponds to a cyclic shift of coordinates, and it can be computed in time $O(n)$.

The algorithm for exponentiation by arbitrary power in finite fields of normal basis based on the Chudnovsky-type multiplication algorithm have been considered in [1] (2017). However, optimization of scalar complexity of this algorithm are not yet studied. It might be one of further research directions for my PhD study.

My main research direction in future after PhD thesis is as follows.

Inverse operation of discrete exponentiation in $\mathbb{Z}_p^*$ (where p is prime number) is *discrete logarithm.* The *Discrete Logarithm Problem* (DLP) is defined as: given a group G, a generator g of the group and an element h of G, to find the discrete logarithm to the base g of h in the group G. The discrete logarithm problem is a fundamental problem underlying the security of many cryptographic systems.

After finishing my thesis, I intend to focus on studying more deeply the *Discrete Logarithm Problem.*

In [6] (2009), Couveignes and Lercier introduced interesting basis representations for finite field extensions: *Elliptic Basis* and *Normal Elliptic Basis.* The most recently, idea of using this basis for discrete logarithm problem has been considered independently and published by two research groups: group of Dvornicich Roberto, Schoof René and Guido Lido ([9]); and group of Thorsten Kleinjung and Benjamin Wesolowski. Especially, in June 2019 Kleinjung and Wesolowski have announced a fully provable quasi-polynomial time algorithm based on elliptic basis representation in a preprint appeared on the eprint archive ([8]).

In July 2019 there is a preprint on the eprint arXiv ([7]) of Antoine Joux and Cecile Pierrot which focuses on this topic. They investigate more on the algorithmic aspects of heuristic variants of discrete logarithm algorithms based on elliptic basis representation. Their key idea is to use *a different model of the elliptic curve used for the elliptic basis* that allows for a relatively simple adaptation of the techniques used with former Frobenius representation algorithms. This really appeals to my interests.

I am very interested in works on the topic about the discrete logarithm problem based on the elliptic basis representation of finite field extension, especially Antoine Joux's idea of using the *model of elliptic curve.* I have studied, investigated and focused on the elliptic curves, as well as algebraic curves in my PhD thesis. Moreover, in aim of finding a new method of approach for my problem in PhD thesis, I have also thought very much about the idea of using the elliptic basis (or normal elliptic basis) for finite field extensions which was introduced by Couveignes and Lercier ([6]). Therefore, I find myself to have good advantage and experience if working on the topic of DLP based on the elliptic representation.

Besides, I am very willing to do research and investigate more various topics in Cryptography if I have an opportunity to receive suggestions from professors in research group.

Bibliography

[1] Kevin Atighehchi, Stéphane Ballet, Alexis Bonnecaze, and Robert Rolland. Arithmetic in finite fields based on the Chudnovsky-Chudnovsky multiplication algorithm. *Mathematics of Computation*, 86(308):2975–3000, 2017.

[2] Stéphane Ballet. Curves with Many Points and Multiplication Complexity in Any Extension of $\mathbb{F}_q$. *Finite Fields and Their Applications*, 5:364–377, 1999.

[3] Stéphane Ballet, Alexis Bonnecaze, and *Thanh-Hung Dang*. On the scalar complexity of chudnovsky^2 multiplication algorithm in finite fields. In *Algebraic Informatics, CAI 2019, Lecture Notes in Computer Science, vol 11545*, pages 64–75. Springer Cham, 2019.

[4] Stéphane Ballet, Jean Chaumine, Julia Pieltant, Matthieu Rambaud, Hugues Randriambololona, and Robert Rolland. On the tensor rank of multiplication in finite extensions of finite fields and related issues in algebraic geometry, 2019.

[5] D. V. Chudnovsky and G. V. Chudnovsky. Algebraic complexities and algebraic curves over finite fields. *J. Complexity*, 4(4):285–316, 1988.

[6] Jean-Marc Couveignes and Reynald Lercier. Elliptic periods for finite fields. *Finite Fields and Their Applications*, 15(1):1–22, 2009.

[7] Antoine Joux and Cécile Pierrot. Algorithmic aspects of elliptic bases in finite field discrete logarithm algorithms. 2019. http://arxiv.org/abs/1907.02689.

[8] Thorsten Kleinjung and Benjamin Wesolowski. Discrete logarithms in quasi-polynomial time in finite fields of fixed characteristic. Cryptology ePrint Archive, Report 2019/751, 2019. https://eprint.iacr.org/2019/751.

[9] Guido Lido. Discrete logarithm over finite fields of ”small” characteristic. Master's Thesis, Universita di Pisa, 2016.

[10] *Thanh-Hung Dang*, Stéphane Ballet, and Alexis Bonnecaze. A note on improving scalar complexity of chudnovsky^2 multiplication algorithm in finite fields. Submitted, 2020.