DE CIFRIS SEMINARS

DE CIFRIS KOINE
Book Series
Volume II

DE CIFRIS SEMINARS

DE CIFRIS KOINE
Series Editorial Board

DE CIFRIS KOINE
Book Series

De Cifris Koine è una collana editoriale curata da De Cifris Press, marchio dell'associazione nazionale De Componendis Cifris dedicata allo studio e alla divulgazione della crittografia e delle discipline correlate.

Questa collana rappresenta un punto di riferimento per la comunità crittografica italiana, offrendo una panoramica delle ricerche e delle innovazioni nel campo. Attraverso la pubblicazione degli atti di conferenze e workshop, De Cifris Koine fornisce non solo approfondimenti scientifici, ma anche contributi divulgativi, mettendo in luce i progressi e le attività dei principali esponenti in questo ambito.

La serie abbraccia un ampio spettro di argomenti, estendendosi oltre la crittografia stessa per includere le sue molteplici applicazioni e intersezioni con altre discipline. Tra queste, si annoverano la teoria dei codici, vari rami della matematica come l'algebra, la teoria dei numeri e la geometria, l'informatica con un focus particolare sulla cybersecurity e sull'informatica teorica, nonché l'ingegneria elettrica, le telecomunicazioni, la storia e gli aspetti legali legati alla crittografia.

Gli articoli pubblicati in questa collana sono accettati in tre lingue: italiano, inglese e francese.

La periodicità della pubblicazione è trimestrale.

De Cifris Koine is a book series published by De Cifris Press, publishing house of the national association De Componendis Cifris, whose activities focus on cryptography and related topics. De Cifris Koine volumes form the voice of the Italian cryptographic community, as they collect communications from both scientific and educational events and summaries of papers of its members and of their activities. In particular, De Cifris Koine hosts conference and workshop proceedings, including short abstracts.

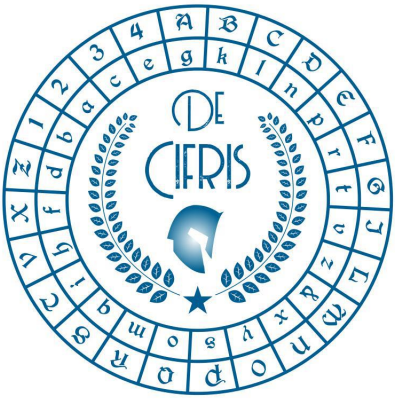
Topics covered in De Cifris Koine volumes relate to cryptography and its applications to and connections with other disciplines, as for example coding theory, maths (mainly algebra, number theory and geometry), computer science (mainly cyber security and theoretical computer science), electronic engineering, telecommunication engineering, history of cryptography and law.
Accepted articles are either in Italian, English or French.
Volumes are published quarterly.

La De Cifris Koine est une collection publiée par la De Cifris Press de l'association nationale italienne De Componendis Cifris. Elle est consacrée à l'étude et à la diffusion de la cryptographie et des disciplines connexes.

Cette collection est une référence importante pour la communauté cryptographique italienne, offrant une vue d'ensemble de la recherche et des innovations dans ce domaine. Grâce à la publication d'actes de conférences et de groupes de travail (workshops), la De Cifris Koine fournit non seulement des contributions scientifiques académiques, mais aussi des contributions à destination du grand public, mettant en lumière les progrès et les activités des principaux acteurs et des principales actrices du domaine.

Les articles de cette collection couvrent un large éventail de sujets allant de la cryptographie à ses nombreuses applications et intersections avec d'autres disciplines. On y retrouve notamment la théorie des codes, diverses branches des mathématiques telles que l'algèbre, la théorie des nombres et la géométrie, l'informatique, avec un accent sur la sécurité informatique et l'informatique théorique, ainsi que le génie électrique, les télécommunications et les aspects juridiques de la cryptographie. Les articles soumis à la De Cifris Koine sont acceptés en italien, anglais et français. La fréquence de publication est trimestrielle.

DE CIFRIS SEMINARS



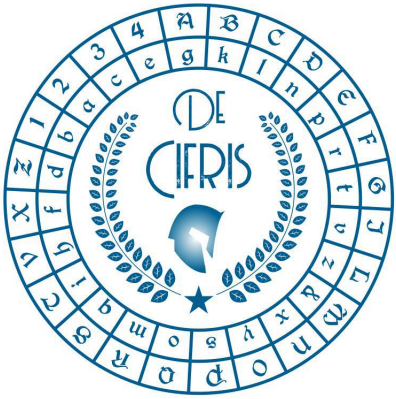
Edited by:

– *Danilo Bazzanella*,
Politecnico di Torino, Italy

– *Giulio Codogni*,
Università di Roma Tor Vergata, Italy

– *Nadir Murru*,
Università di Trento, Italy

– *Roberto Zunino*,
Università di Trento, Italy.



Pubblicazione trimestrale di proprietà dell'associazione nazionale di crittografia
De Componendis Cifris

Autorizzazione del Tribunale di Milano in data 23 - 02 - 2024
Num. R.G. 1315/2024 Num. Reg. Stampa 22
ISSN 3034-9796 - ISBN 979-12-81863-01-9

Editore: De Componendis Cifris APS.
Marchio Editoriale: De Cifris Press.
Direttore responsabile: Massimiliano Sala
Redazione: Antonino Alì, Nadir Murru
Luogo di pubblicazione: Via Gianfranco Zuretti 34 - 20125 Milano
e-mail: editorial@decifris.it
Stampa in proprio
Numero 2 - Pubblicato il 30 - 05 - 2024

PREFACE

De Componendis Cifris APS è stata costituita formalmente solo nel dicembre 2022, ma è attiva da molti anni con incontri ed eventi. Una delle sue iniziative è stata l'organizzazione di numerosi seminari. È quindi opportuno che un libro Koine raccolga alcuni dei nostri migliori seminari. Alcuni risultati possono essere obsoleti, ma gli articoli forniscono ancora diversi spunti preziosi in molte aree della ricerca crittografica, servendo al contempo da piacevole introduzione al campo specifico. Il volume si conclude con due contributi interessanti e originali.

Riteniamo che i curatori del libro, Danilo Bazzanella, Giulio Codogni, Nadir Murru e Roberto Zunino, abbiano fatto un lavoro pregevole nel selezionare sia un interessante campione di seminari sia i due invited paper, entrambi attualissimi: la crittografia post-quantistica basata su isogenesi e la fattorizzazione di interi.

De Componendis Cifris APS was formally established only in December 2022, but it has been active for many years in meetings and events. One of its initiatives has been the organization of many seminars. It is therefore fitting that a Koine book collects some of our best seminars. Some results may be outdated, but the papers still provide valuable insights into many areas of cryptographic research, while also serving as nice introductions. This Koine volume concludes with two interesting and original contributions.

We believe that the volume's editors, Danilo Bazzanella, Giulio Codogni, Nadir Murru and Roberto Zunino, have done a great job in selecting both an interesting sample of seminars and the two invited papers: (isogeny-based) post-quantum cryptography and integer factorization.

De Componendis Cifris APS n'a été officiellement créée qu'en décembre 2022, mais elle est active depuis de nombreuses années dans le cadre de réunions et d'événements. L'une de ses initiatives a été l'organisation de séminaires. Il est donc tout à fait approprié que un livre Koine rassemble quelques-uns de nos meilleurs séminaires. Certains des résultats peuvent être dépassés, mais les sujets traités fournissent toujours des aperçus précieux dans de nombreux domaines de la recherche cryptographique et servent également d'introduction agréable au domaine spécifique. Deux articles originaux et intéressants concluent le volume.

Nous pensons que les éditeurs du volume, Danilo Bazzanella, Giulio Codogni, Nadir Murru et Roberto Zunino, ont fait un excellent travail en sélectionnant un échantillon intéressant de séminaires et deux communications invitées: la cryptographie post-quantique basée sur les isogénies et la factorisation des nombres entiers.

Massimiliano Sala & Antonino Alì
Editor in Chief & Managing Editor
De Cifris Koine

Table of Contents

Part I
Introduction

Introduction to De Cifris Seminars

Among the objectives of the national association De Componendis Cifris is the aim of spreading the cryptographic culture in Italy. To achieve this purpose, in recent years it has promoted and coordinated a series of seminars held in various Italian universities. This volume is dedicated to collecting the extended abstracts of some seminars organised by members of universities participating in the association, while also containing two extensive research papers.

The seminars cover many of the most interesting topics in modern cryptography, dealing with both theoretical aspects and their applications. In the part of the volume dedicated to the theory, there are seven extended abstracts whose topics range over cryptanalysis, post-quantum cryptography, and symmetric cryptography.

Part I contains a foreword by some members of De Cifris, who are also professors and researchers in the fields covered in this volume.

Part II addresses some theoretical aspects of scheme design and cryptanalysis.
In particular, the extended abstract *"An overview on cryptanalysis of ARX ciphers"* (2019), by Stefano Barbero, describes some of the main cryptanalytic attacks against *ARX ciphers*, a class of symmetric-key algorithms only employing modular additions, bitwise rotations and exclusive-OR's.
Michela Ceria, Theo Moriarty and Andrea Visconti present *"Efficient cryptanalysis over multivariate Ore extensions"* (2020), an attack specialized to the protocol of Ore extensions exploiting advanced techniques of computational algebra.
Lorenzo Grassi, in *"Cryptanalysis of round-reduced AES"* (2019), describes some progress in the cryptanalysis of the block cipher *AES* (Advanced Encryption Standard), with ingenious techniques from differential cryptanalysis.
Finally, *"An investigation on integer factorization applied to public-key cryptography"* (2020), by Giordano Santilli provides an overview of the best factorization methods that can be exploited for attacking those public key cryptosystems whose security is based on the difficulty of the integer factorization problem.
The first extended abstract devoted to designing ciphers rather than breaking them is *"A Group Theoretical Approach for Symmetric Encryption"* (2020). The security of symmetric ciphers is related to the size of the symmetric group they generate. Riccardo Aragona presents some results on the size of the subgroup generated by these permutations.

De Cifris Koine – DE CIFRIS SEMINARS https://doi.org/10.69091/koine/vol-2-I01

"Shift Invariance in Symmetric Cryptography" (2019) by Luca Mariot is the extended abstract of a seminar titled "Boolean functions, S-Boxes and Evolutionary Algorithms" focused on the design of (symmetric) ciphers exploiting Boolean functions and cellular automata, as well as their related optimization problems which can be tackled by genetic programming.
Lastly, *"On adapting NTRU for Post-Quantum Public-Key Encryption"* (2020), by Simone Dutto, Guglielmo Morgari and Edoardo Signorini focusses on solving the problem by introducing a PKE scheme obtained from the KEM proposed in the NTRU submission at NIST.

Part III of the volume concerns applications, dealing with topics such as non-interactive proofs, machine learning, blockchain, applications of post-quantum cryptography, digital identity, and key derivation functions.
In particular, the paper *"Non-interactive time-based proof-of-stake finality"* (2019) by Giovanni Antino and Iris Dimmi provides a glimpse inside the novel blockchain technologies researched by AiliA SA for the Takamaka platform, investigating the design of a unique Proof of Stake protocol, and studying its high performance and robustness against Byzantine faults.
The next extended abstract concerns the seminar *"Privacy-preserving Information Sharing"* (2019) by Carlo Blundo focussing on a functionality of secure multiparty computation known as private set intersection (PSI). In short, PSI allows a client to know which elements of its own private set are members of the server's private set, without learning about the non-members.
The extended abstract *"The Links between Machine Learning and Blockchain"* (2020) by Andrea Gangemi is dedicated to presenting the possible applications of Machine Learning to Blockchain technology, such as the (partial) deanonymization of the cryptocurrencies' holders, the prediction of future prices of cryptocurrencies and the construction of consensus algorithms for the blockchain. Conversely, this paper also introduces some applications of Blockchain technology to Machine Learning, such as the decentralization of datasets.
"Post-quantum cryptography and the automotive industry" (2021) by Efstathia Katsigianni discusses the growing role of asymmetric encryption in the automotive industry and consequent threats posed by quantum computing. The author works for automotive industries, so she was able to give an up-to-date survey of the state of the art.
Alessandro Tomasi, in *"Digital Identity - modern tools and perspectives"* (2020), provides a quick overview of the main technologies and tools behind modern authentication infrastructures, taking into account both their technical limitations and relevant regulations.
Lastly, the abstract *"KDFs: an essential (and usually transparent) component of real-world applications"* (2019) by Andrea Visconti discusses both Key Derivation Functions and Hash functions, with a focus on real-world applications.

Part IV contains two research papers.
The first one is *"Isogenies Demystified"*, by Luca De Feo, providing an engaging introduction to the family of asymmetric cryptosystems based on isogenies. This fast-growing family is thought to be quantum-resistant; the analysis of its schemes often relies on the study of isogeny graphs, namely graphs whose vertices are elliptic curves and edges are isogenies. Isogeny graphs are also interesting from a theoretical point of view. This survey, like all the chapters of this book, was written in 2021, so it does not concern recent developments. In particular, it does not report, for example, on some recent attacks on SIKE or on the newly developed zero-knowledge proofs and signatures.
The second one, entitled *"Continued Fractions, Quadratic Fields, and Factoring: Some Computational Aspects"* by Michele Elia, is a research paper devoted to an original and promising factorisation method inspired by the famous SQUFOF (SQUare FOrms Factorisation, by Daniel Shanks). More precisely, M. Elia exploits the theory of continued fractions and quadratic fields to construct a novel factorisation method. Let N be the integer to factorise. He exhibits some special quadratic forms arising from the convergents of the continued fraction expansion of the square root of N. The author proves that some non-trivial factors of N can be found thanks to these quadratic forms, by evaluating the regulator of the related quadratic field (generated by the square root of N).

Danilo Bazzanella
Giulio Codogni
Nadir Murru
Roberto Zunino

Part II
Extended Abstracts: Theory

Group Theoretical Approach for Symmetric Encryption

Riccardo Aragona

Department of Engineering and Computer Science and Mathematics, University of L'Aquila, Italy
riccardo.aragona@univaq.it

A *block cipher* Φ is a family of key-dependent permutations

$$\{E_K \mid E_K : \mathcal{M} \to \mathcal{M},\ K \in \mathcal{K}\},$$

where $\mathcal{M}$ is the message space, $\mathcal{K}$ the key space, and $|\mathcal{M}| \leqslant |\mathcal{K}|$. The permutation E_K is called the *encryption function induced by the master key* K. The block cipher Φ is called an *iterated block cipher* if there exists an integer $r \geqslant 2$ such that for each $K \in \mathcal{K}$ the encryption function E_K is the composition of r *round functions*, that is, $E_K = \varepsilon_{1,K}\, \varepsilon_{2,K} \ldots \varepsilon_{r,K}$. To provide efficiency, each round function is itself the composition of a public component provided by the designers and a private component derived from the user key (by means of a public procedure known as *key-schedule*).

In the theory of modern iterated block cipher, two frameworks are mainly considered: *Substitution-Permutation Networks (SPN)*, and *Feistel Networks (FN)*. In both cases, the principles of confusion and diffusion suggested by Shannon [18] are implemented by considering each round function for an SPN and F-function for an FN as the composition of key-induced permutation as well as non-linear confusion layers (usually called *S-Boxes*) and linear diffusion layers, which are invertible in the case of SPNs and preferably (but not necessarily) invertible in the case of FNs.

Algebraic attacks might represent serious threats. It is possible to link some algebraic properties of confusion / diffusion layers and some algebraic weaknesses of the corresponding cipher. Firstly, in 1975 Coppersmith and Grossman [12] considered a set of functions which can be used to define a block cipher and, by studying the permutation group generated by those, they opened the way to a new branch of research focused on group-theoretical properties which can reveal weaknesses of the cipher itself. As it has been proved in [14], if such a group is too small, then the cipher is vulnerable to birthday-paradox attacks. Recently, in [7] the authors proved that if such group is contained in an isomorphic image of the affine group of the message space induced by a hidden sum, then it is possible to embed a dangerous trapdoor in it. More relevant in [17], Paterson built a DES-like cipher, resistant to

De Cifris Koine – DE CIFRIS SEMINARS –https://doi.org/10.69091/koine/vol-2-T01

both linear [15] and differential [5] cryptanalysis, whose encryption functions generate an imprimitive group and showed how the knowledge of this trapdoor can be turned into an efficient attack to the cipher. For this reason, a branch of research in symmetric cryptography is focused on studying the group generated by the encryption functions $\Gamma_\infty(\Phi) = \langle \varepsilon_{h,K} \mid 1 \leqslant h \leqslant r, K \in \mathcal{K} \rangle$ of a cipher Φ, and in particular, on showing that this group is primitive and not of affine type (see e.g. [2], [3], [4], [11], [10], [13], [19], [20], [21]).

For translation-based ciphers, which include the most common SPN ciphers, in [10] the authors provided two cryptographic conditions on S-Boxes (i.e., the weakly differential uniformity and the strongly anti-invariance) which guarantee the primitivity of the group generated by the round functions of the cipher. Furthermore, in [11], using the O'Nan-Scott classification of finite primitive groups, together with another cryptographic assumption, it has been proved that the group in question is the alternating group. Unfortunately, both of these results are not applicable to many lightweight ciphers. Motivated by this, in [2], joint work with Calderini, Tortora and Tota, we continue the study of the group generated by the round functions of an SPN. In particular, we prove the primitivity of the group Γ_∞ generated by the round functions of a translation-based cipher satisfying different cryptographic assumptions with respect to the result given in [10]. More precisely, we consider the (strong) differential uniformity which allows us to relax the hypothesis on the strongly anti-invariance. Then, we provide some additional conditions from which it follows that Γ_∞ is the alternating group. As an immediate consequence, we deduce that the round functions of some lightweight ciphers, such as PRESENT, RECTANGLE and PRINTcipher, generate the alternating group.

It is well-known that the non-linearity of the confusion layer is a crucial parameter for the security of the cipher. In particular, in order to prevent statistical attacks (e.g. differential [5] and linear [15] cryptanalysis), block ciphers' designers are very interested in invertible S-boxes reaching the best possible differential uniformity, which is two. Functions satisfying such property are called *almost-perfect non-linear* (APN) [16] and are extensively studied. Unfortunately, APN permutations are known only when the dimension s of the input space for the S-box is an odd number, except for the case of the Dillon's function ($s = 6$) [6], which nowadays represents the only isolated case [9]. It has been shown that no permutation with $s = 4$ is APN [8] and the problem is still without answers for $s \geqslant 8$. On the other hand, the cases when $s \in \{4, 8\}$ are the most used for implementation reasons. In [1], we propose a new general framework for block ciphers, called *wave cipher*, combining a typical structure of an SPN round function within a Feistel Network, which may feature injective APN S-Boxes of even size. We show that this framework produces provably secure ciphers, under some cryptographic assumptions, with respect to the imprimitivity attack. In particular we prove a group-theoretical result, which,

as a consequence, links the primitivity of the action of an SPN with that of an FN. From that follows the primitivity of the group generated by the round functions of a wave cipher. Instead, in order to prove the security of the given wave cipher with respect to other classical statistical attacks, it may be necessary to analyse the single instance under consideration.

References

1. R. Aragona, M. Calderini, R. Civino, M. Sala, and I. Zappatore. Wave-shaped round functions and primitive groups. *Advances in Mathematics of Communications*, 13(1):67–88, 2019.
2. R. Aragona, M. Calderini, A. Tortora, and M. Tota. On the primitivity of PRESENT and other lightweight ciphers. *Journal of Algebra and its Applications*, 17(6):1850115, 2020.
3. R. Aragona, A. Caranti, F. Dalla Volta, and M. Sala. On the group generated by the round functions of translation based ciphers over arbitrary fields. *Finite Fields and Applications*, 25:293–305, 2014.
4. R. Aragona, A. Caranti, and M. Sala. The group generated by the round functions of a GOST-like cipher. *Annali di Matematica Pura ed Applicata*, 196(1):1–17, 2016.
5. Eli Biham and Adi Shamir. Differential cryptanalysis of DES-like cryptosystems. *Journal of Cryptology*, 4(1):3–72, January 1991.
6. K. A. Browning, J. F. Dillon, M. T. McQuistan, and A. J. Wolfe. An APN permutation in dimension six. In *Finite Fields: Theory and Applications*, volume 518 of *Contemporary Mathematics*, pages 33–42, 2010.
7. M. Calderini, R. Civino, and M. Sala. On properties of translation groups in the affine general linear group with applications to cryptography. *Journal of Algebra*, 569:658–680, 2021.
8. M. Calderini, M. Sala, and I. Villa. A note on APN permutations in even dimension. *Finite Fields and Applications*, 46:1–16, 2017.
9. A. Canteaut, S. Duval, and L. Perrin. A generalisation of Dillon's APN permutation with the best known differential and nonlinear properties for all fields of size 2^{4k+2}. *IEEE Transactions on Information Theory*, 63:7575–7591, 2017.
10. A. Caranti, F. Dalla Volta, and M. Sala. On some block ciphers and imprimitive groups. *AAECC*, 20:339–350, 2009.
11. A. Caranti, F. Dalla Volta, and M. Sala. An application of the O'Nan-Scott theorem to the group generated by the round functions of an AES-like cipher. *Design, Codes and Cryptography*, 52(3):293–301, 2009.
12. D. Coppersmith and E. Grossman. Generators for certain alternating groups with applications to cryptography. *SIAM Journal on Applied Mathematics*, 29(4):624–627, 1975.
13. X. D. Hou. Affinity of permutations of $\mathbb{F}_2^n$. *Discrete Applied Mathematics*, 154(2):313–325, 2006.
14. Burton S. Kaliski Jr., Ronald L. Rivest, and Alan T. Sherman. Is the data encryption standard a group? (results of cycling experiments on DES). *Journal of Cryptology*, 1(1):3–36, January 1988.

15. Mitsuru Matsui. Linear cryptanalysis method for DES cipher. In Tor Helleseth, editor, *EUROCRYPT'93*, volume 765 of *LNCS*, pages 386–397. Springer, Heidelberg, May 1994.
16. Kaisa Nyberg. Differentially uniform mappings for cryptography. In Tor Helleseth, editor, *EUROCRYPT'93*, volume 765 of *LNCS*, pages 55–64. Springer, Heidelberg, May 1994.
17. Kenneth G. Paterson. Imprimitive permutation groups and trapdoors in iterated block ciphers. In Lars R. Knudsen, editor, *FSE'99*, volume 1636 of *LNCS*, pages 201–214. Springer, Heidelberg, March 1999.
18. Claude E. Shannon. Communication theory of secrecy systems. *Bell Systems Technical Journal*, 28(4):656–715, 1949.
19. R. Wernsdorf. The round functions of SERPENT generate the alternating group. Nist comment, NIST, 2000. available at `http://csrc.nist.gov/archive/aes/round2/\comments/20000512-rwernsdorf.pdf`.
20. Ralph Wernsdorf. The one-round functions of the DES generate the alternating group. In Rainer A. Rueppel, editor, *EUROCRYPT'92*, volume 658 of *LNCS*, pages 99–112. Springer, Heidelberg, May 1993.
21. Ralph Wernsdorf. The round functions of RIJNDAEL generate the alternating group. In Joan Daemen and Vincent Rijmen, editors, *FSE 2002*, volume 2365 of *LNCS*, pages 143–148. Springer, Heidelberg, February 2002.

An overview on cryptanalysis of ARX ciphers

Stefano Barbero

Department of Mathematical Sciences L. Lagrange, Politecnico of Torino, Italy
stefano.barbero@polito.it

The purpose of this overview is to describe the main cryptanalytic attacks currently used against ARX ciphers. ARX ciphers are block or stream ciphers described by easy algorithms with fast performance on PCs, compact implementation and essentially no risks of timing attacks. Considering n-bit strings they are defined using the natural bijection between $(\mathbb{F}_2)^n$ and $\mathbb{Z}/2^n\mathbb{Z}$

$$x = (x_{n-1}, x_{n-2}, \ldots, x_1, x_0) \leftrightarrow x_{n-1}2^{n-1} + x_{n-2}2^{n-2} + \cdots + x_1 2 + x_0$$

and three main operations: the addition mod 2^n indicated by $\boxplus$; the r–bit rotation, where

$$x \lll r = (x_{n-r-1}, x_{n-r-2}, \ldots, x_1, x_0, x_{n-1}, \ldots x_{n-r})$$

and

$$x \ggg r = (x_{r-1}, \ldots, x_1, x_0, x_{n-1}, x_{n-2}, \ldots, x_r)$$

respectively indicate a constant distance left rotation or right rotation of r bits ($r < n$) of a n-bit word $x = (x_{n-1}, x_{n-2}, \ldots, x_r, \ldots, x_0)$, and the XOR bitwise addition indicated by $\oplus$. Thanks to these operations and assuming constants included, the ARX schemes are functionally complete (see, e. g., [9]): every possible logic gate can be realized as a network of gates using ARX operations and constants. However, ARX ciphers have also some basic disadvantages: they are not best trade–off in hardware, although there are some different attempts of optimizations for various ARX ciphers; it is still not so completely clear which is their security against some cryptanalytic tools (e.g., [7, 8]) such as linear and differential cryptanalysis, it is also still not so clear which is their security against side channel attacks, i. e., attacks based on all hardware information detected from their implementation.

Perhaps, one of the most interesting cryptanalytic methods against ARX ciphers is Algebraic Cryptanalysis, which mainly consists of a deterministic key-recovery attack based on finding and solving an equation system, with coefficients in $\mathbb{F}_2$, that represents the encryption (or decryption) function (see, e. g., [2]). The set of indeterminates of the system represents a plaintext as well as a ciphertext, the internal states of the cipher, and the encryption/decryption key. The set of polynomial equations of the system arising from the ARX scheme can be solved using algorithms involving algebraic tools like Gröbner bases and SAT–solvers.

De Cifris Koine – DE CIFRIS SEMINARS –https://doi.org/10.69091/koine/vol-2-T02

In order to attack block ciphers like DES and FEAL, Matsui [11] developed Linear Cryptanalysis, which usefully applies when it is possible to find an "effective" linear expression describing a given cipher algorithm. In this case we obtain one or more linear relations among the parities of plaintext, ciphertext and the secret key. Let P, C and K denote the plaintext, the ciphertext and the key, respectively. Considering fixed bit locations $i_1, i_2, \ldots, i_a, j_1, j_2, \ldots, j_b, k_1, k_2, \ldots, k_c$, if we have with probability $p \neq \frac{1}{2}$ and effectiveness $|p - \frac{1}{2}|$ that

$$P[i_1, i_2, \ldots, i_a] \oplus C[j_1, j_2, \ldots, j_b] = K[k_1, k_2, \ldots, k_c],$$

for a randomly given plaintext P and the corresponding ciphertext C, we are able to determine one key bit $K[k_1, k_2, \ldots, k_c]$ with a straightforward algorithm, described by Matsui [11], based on the maximum likelihood method and using N plaintexts. The success rate of this method increases when N or $|p - \frac{1}{2}|$ does.

Another important cryptanalytic tool has been introduced by Biham and Shamir (see, e. g., [3], [4], and [5]), in order to attack the previous standard DES. The idea of their XOR–Differential Cryptanalysis is finding some couples of differentials $\Delta X = X' \oplus X''$, $\Delta Y = Y' \oplus Y''$, where a pair of plaintexts X', X'' yields to a pair of ciphertexts (or internal states of the cipher) Y', Y'' such that ΔY is not uniformly distributed. Finding one or more differentials can help to distinguish a ciphertext from randomness and to recover the (partial) key used in the cipher. Indeed, we may encrypt many pairs of chosen plaintexts having difference ΔX and try to decrypt the corresponding ciphertexts using all the possible subkeys to get the outputs (or internal states) Y. Checking the frequency of ΔY, we can select with high probability the correct subkey, observing that this frequency of ΔY must be close to the conjectured value of the probability $\mathbb{P}[\Delta Y | \Delta X]$.

Resting on the same core ideas, Additive Differential Cryptanalysis focuses on the difference of two outputs of a standard ARX operation taking into account the effects of modular addition. Since a standard ARX operation is defined as

$$ARX(a, b, d, r) = ((a \boxplus b) \lll r) \oplus d$$

where a, b, d are n–bit vectors, fixing the additive differences $\Delta\alpha$, $\Delta\beta$, $\Delta\lambda$ and $\Delta\mu$, we may define the difference Δe between two outputs of ARX as

$$\Delta e = ARX(a \boxplus \Delta\alpha, b \boxplus \Delta\beta, d \boxplus \Delta\lambda, r) \boxminus ARX(a, b, d, r).$$

Additive differences pass through modular addition with probability 1, thus we have $\Delta\gamma = \Delta\alpha \boxplus \Delta\beta$ and we may define the additive differential probability of ARX as

$$adp^{ARX} = (\Delta\gamma, \Delta\lambda \xrightarrow{r} \Delta\mu) = \frac{|\{(a \boxplus b, d) : \Delta e = \Delta\mu\}|}{|\{(a \boxplus b, d)\}|}.$$

As pointed out by Velichkov et alii [12], S–functions are the best tool one could use in order to estimate adp^{ARX}.

Another way to attack an ARX scheme is studying rotational differentials, i. e., investigating the propagation of rotations throughout the encryption steps of an ARX scheme applying Rotational-Differential Cryptanalysis. A rotation of r bits passes across XOR and across another bit rotation with probability 1, while the probability that it passes across a modular addition of two n–bit strings is given by $p_r = \frac{1}{4}(1 + 2^{r-n} + 2^{-r} + 2^{-n})$ (see, e. g., [6] and [10]). This probability is maximized to $2^{-1.415}$ when n is large and $r = 1$, therefore in an ARX scheme with q not chained modular additions, we can detect nonrandomness if $(p_1)^q > 2^{-n}$, i. e., if the implementation consists of $q < \frac{n}{1.415}$ not chained modular additions. The case of chained modular additions is deeply discussed in [10], where an algorithm for computing the rotational probability of ARX ciphers with chained modular additions is given and applied to mount rotational–differential attacks against BLAKE2 and a simplified version of Skein. The main obstacle in applying Rotational–Differential Cryptanalysis is the possible presence of constants in the ARX scheme. In this case one could try to use the Rotational–XOR Cryptanalysis with constants, recently developed by Ashur and Liu [1]. In particular the authors evaluate for large n the probability that the following equality holds

$$((x \oplus a_1) \boxplus (y \oplus b_1) \oplus \Delta_1) \lll 1 = ((x \lll 1) \oplus a_2) \boxplus ((y \lll 1) \oplus b_2) \oplus \Delta_2$$

where x, y are independent random n–bit strings and a_i, b_i, Δ_i, $i = 1, 2$ are constant n-bit strings. They use their results in order to find a 7–round distinguisher based on Rotational–XOR differences for SPECK32/64.

References

1. Tomer Ashur and Yunwen Liu. Rotational cryptanalysis in the presence of constants. *IACR Trans. Symm. Cryptol.*, 2016(1):57–70, 2016. `https://tosc.iacr.org/index.php/ToSC/article/view/535`.
2. G. V. Bard. *Algebraic Cryptanalysis*. Springer, 2009.
3. E. Biham and A. Shamir. *Differential Cryptanalysis of the Data Encryption Standard*. Springer, 1993.
4. Eli Biham and Adi Shamir. Differential cryptanalysis of DES-like cryptosystems. In Alfred J. Menezes and Scott A. Vanstone, editors, *CRYPTO'90*, volume 537 of *LNCS*, pages 2–21. Springer, Heidelberg, August 1991.
5. Eli Biham and Adi Shamir. Differential cryptanalysis of the full 16-round DES. In Ernest F. Brickell, editor, *CRYPTO'92*, volume 740 of *LNCS*, pages 487–496. Springer, Heidelberg, August 1993.
6. M. Daum. *Cryptanalysis of Hash Functions of the MD4-Family*. PhD thesis, Rhur Universität Bochum, 2005.
7. M. Hadian Dehkordi and R. Taghizadeh. Multiple differential-zero correlation linear cryptanalysis of reduced-round cast-256. *Journal of Mathematical Cryptology*, 11(2):55–62, 2017.

8. Rebecca E. Field and Brant C. Jones. Using carry-truncated addition to analyze add-rotate-xor hash algorithms. *Journal of Mathematical Cryptology*, 7(2):97–110, 2013.
9. Dmitry Khovratovich and Ivica Nikolic. Rotational cryptanalysis of ARX. In Seokhie Hong and Tetsu Iwata, editors, *FSE 2010*, volume 6147 of *LNCS*, pages 333–346. Springer, Heidelberg, February 2010.
10. Dmitry Khovratovich, Ivica Nikolic, Josef Pieprzyk, Przemyslaw Sokolowski, and Ron Steinfeld. Rotational cryptanalysis of ARX revisited. In Gregor Leander, editor, *FSE 2015*, volume 9054 of *LNCS*, pages 519–536. Springer, Heidelberg, March 2015.
11. Mitsuru Matsui. Linear cryptanalysis method for DES cipher. In Tor Helleseth, editor, *EUROCRYPT'93*, volume 765 of *LNCS*, pages 386–397. Springer, Heidelberg, May 1994.
12. Vesselin Velichkov, Nicky Mouha, Christophe De Cannière, and Bart Preneel. The additive differential probability of ARX. In Antoine Joux, editor, *FSE 2011*, volume 6733 of *LNCS*, pages 342–358. Springer, Heidelberg, February 2011.

Efficient cryptanalysis over multivariate Ore extensions

Michela Ceria[1], Theo Moriarty[2], and Andrea Visconti[3]

[1] Department of Mechanics, Mathematics and Management, Politecnico of Bari, Italy
[2] Department of Mathematics, University of Genova, Italy
[3] Department of Computer Science, University of Milano, Italy
michela.ceria@gmail.com, 5919@unige.it, andrea.visconti@unimi.it

Key exchange protocols are cryptographic protocols that allow two parties to share a common secret while communicating through an insecure channel. This enables the two parties to use a symmetric algorithm to communicate, without the high costs of transmitting securely the common cryptographic key.

In their milestone paper [8], Diffie and Hellman proposed a key exchange scheme, basing its security on the *Discrete Logarithm Problem* on the multiplicative group of integers modulo a prime p, then generalized to finite cyclic groups. Since then, many other algorithms in the same fashion have been developed by several authors and, in particular, in this paper we focus on those ones that are developed over noncommutative algebra frameworks.

The protocol described in [5] is the first one employing polynomial algebra; more precisely, it relies on evaluation of univariate polynomials on the elements of a public noncommutative ring R.

The first involved party, Alice, takes $a, b \in R$, $m, n \in \mathbb{N}$ and a secret polynomial $f \in \mathbb{Z}[x]$; then she sends to the second involved party, Bob, m, n, a, b and $A = f(a)^m b f(a)^n$. Bob selects secretly $h \in \mathbb{Z}[x]$ and sends to Alice $B = h(a)^m b h(a)^n$. The common secret is $S = f(a)^m B f(a)^n = h(a)^m A h(a)^n$.

We remark also that the most general setting employing polynomial algebra is the one defined in [10]; for other results in similar fashion and some cryptanalysis, see [7] and the references therein.

In this paper, we deal with a protocol [4] developed over multivariate Ore extensions [6], and we will present an attack to this protocol.

Given a (non-necessarily commutative) domain R, consider the left R-module of formal polynomials $R[Y]$. To make it a ring such that the multiplication of polynomials is associative and both-sided distributive and the degree of a product is equal to the sum of the degree of the factors, Ore [12] needed to define the product $Y \cdot r, r \in R$, requiring the existence of two maps $\alpha, \delta : R \to R$ such that

$$Y \cdot r = \alpha(r)Y + \delta(r) \text{ for each } r \in R\,.$$

De Cifris Koine – DE CIFRIS SEMINARS –https://doi.org/10.69091/koine/vol-2-T03

An *Ore extension* is given by $R[Y]$ with the ring structure such that

1. for each $r \in R$, $\alpha(r) = 0$ implies $r = 0$;
2. α is a ring endomorphism;
3. δ is an α-derivation of R, namely an additive map satisfying $\delta(rr') = \alpha(r)\delta(r') + \delta(r)r'$ for each $r, r' \in R$.

Such a structure is usually denoted by $R[Y; \alpha, \delta]$. Extending this idea, an *iterative Ore extension* is a ring defined as

$$R_n := R[Y_1; \alpha_1, \delta_1][Y_2; \alpha_2, \delta_2] \cdots [Y_n; \alpha_n, \delta_n],$$

where, for every $i > 1$, α_i is an endomorphism and δ_i an α_i-derivation of the iterative Ore extension

$$R_{i-1} := R[Y_1; \alpha_1, \delta_1] \cdots [Y_{i-1}; \alpha_{i-1}, \delta_{i-1}].$$

It is possible to extend α_i to an endomorphism of R_n and δ_i to an α_i-derivation in R_n, by setting $\alpha_i(Y_j) = Y_j$ and $\delta_i(Y_j) = 0$ for each $i \leqslant j \leqslant n$. Finally, a *multivariate Ore extension* is an iterative Ore extension which satisfies

- $\alpha_j\delta_i = \delta_i\alpha_j$, for each i, j, $i \neq j$;
- $\alpha_i\alpha_j = \alpha_j\alpha_i$, $\delta_i\delta_j = \delta_j\delta_i$ for $i < j \leqslant n$;
- $\alpha_j(Y_i) = Y_i, \delta_j(Y_i) = 0$ for $i \leqslant j \leqslant n$.

We can now describe Burger-Heinle's protocol. Consider a multivariate Ore extension T with constant subring R and three non-mutually commuting elements $L, P, Q \in \mathsf{T}$. Denoting

$$C = \{f \in R[t], \deg(f) > 0, f(0) \neq 0\},$$

we can define the sets $C_l := \{f(P), f \in C\}, C_r := \{f(Q), f \in C\}$. All these data are publicly available. Alice chooses secretly $(P_A, Q_A) \in C_l \times C_r$ and Bob does the same by choosing $(P_B, Q_B) \in C_l \times C_r$. Then, Alice sends $A = P_A L Q_A$ to Bob and gets $B = P_B L Q_B$ from him. The shared secret is $S = P_A B Q_A = P_B A Q_B$.
Note first that the protocol can be verbatim generalized to the context of *iterated Ore extensions with power substitutions* [11], as explicitly done in [7]; moreover, both the original and the extended protocols can be broken via the attack we are going to propose now, whose only ingredients are *Buchberger reduction* [1–3, 9] and *left/right divisibility*. We show now how to recover the polynomials $f, g \in R[t]$ such that $f(P) = P_A, g(Q) = Q_A$ from A. Let us write $g(t) = \sum_{i=a}^{d} c_i t^i = t^a \cdot \sum_{i=0}^{d-a} c_{a+i} t^i$, where $a := \min\{i : c_i \neq 0\} \leqslant d$, so that $Q_A = \sum_{i=a}^{d} c_i Q^i$ and, for a fixed term order on T, let us denote by $\mathbf{M}(Q)$ the leading monomial of Q and by $\mathbf{Q}(Q) := Q - \mathbf{M}(Q)$ the tail. We define a new variable U and, setting $\mathbf{M}(Q) \to \mathbf{Q}(Q) + U$ (so substituting

all occurrences of the leading monomial with the tail), we reduce A from the right $a+1$ times, getting

$$\begin{aligned} A = f(P)Lg(Q) &= f(P)L\sum_{i=a}^{d} c_i Q^i \\ &\to f(P)L\sum_{i=a+1}^{d} c_i Q^{i-a-1}U\cdot U^a + f(P)Lc_aU^a \\ &= XU\cdot U^a + YU^a, \end{aligned}$$

where $Y = f(P)Lc_a$ and $X = f(P)L\sum_{i=a+1}^{d} c_i Q^{i-a-1}$. This implies that

- we can find $f(P)$ by dividing Y by L from the right and we can get f by reducing with respect to P;
- we can get $\sum_{i=a+1}^{d} c_i Q^{i-a-1}$ by dividing X by Y from the left, and from that we can deduce g by reduction.

Since a is unknown, we still need a way to understand whether we performed the correct number of reduction steps, so that

$$Y := f(P)Lc_a$$

and

$$X := f(P)L\sum_{i=a+1}^{d} c_i Q^{i-a-1}.$$

For this aim we test whether L divides Y from the right: in the positive case, the attack can be concluded as above; otherwise, we continue by reducing until the positive case is reached. Note that, by symmetry, we can find $Lg(Q)$ and f.

We conclude by saying that [4] presents also a three-pass exchange protocol, which can be broken with a variation of the attack above, as examined in [7].

References

1. B. Buchberger. *Ein Algorithmus zum Auffinden der Basiselemente des Restklassenringes nach einem nulldimensionalen Polynomideal.* PhD thesis, Innsbruck, 1965.
2. B. Buchberger. Ein algorithmisches kriterium für die lösbarkeit eines algebraischen gleichungssystems. *Aequationes mathematicae*, 4:374–383, 1970.
3. B. Buchberger. Gröbner bases: An algorithmic method in polynomial ideal theory. In N. K. Bose, editor, *Multidimensional Systems Theory*, pages 184–232. Reider, 1985.
4. R. Burger and A. Heinle. A Diffie-Hellman-like key exchange protocol based on multivariate ore polynomials. arXiv preprint, 2014. `https://arxiv.org/pdf/1407.1270.pdf`.
5. Zhenfu Cao, Xiaolei Dong, and Licheng Wang. New public key cryptosystems using polynomials over non-commutative rings. Cryptology ePrint Archive, Report 2007/009, 2007. `https://eprint.iacr.org/2007/009`.
6. M. Ceria and T. Mora. Buchberger-Zacharias theory of multivariate Ore extensions. *Journal of Pure and Applied Algebra*, 221(12):2974–3026, 2017.
7. M. Ceria, T. Mora, and A. Visconti. Why you cannot even hope to use ore algebras in cryptography. *AAECC*, pages 1–16, 2021.
8. Whitfield Diffie and Martin E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.
9. P. Gordan. Neuer beweis des hilbertschen satzes über homogene funktionen. *Gottingen Nachrichten*, pages 240–242, 1899.
10. S. Kanwal, S. Inam, R. Ali, and S. Qiu. Two new variants of stickel's key exchange protocol based on polynomials over noncommutative rings, 2014.
11. B. Nguefack and E. Pola. Effective Buchberger-Zacharias-Weispfenning theory of skew polynomial extensions of restricted bilateral coherent rings. *Journal of Symbolic Computation*, 2019.
12. Oystein Ore. Theory of non-commutative polynomials. *Annals of Mathematics*, 34:480–508, 1933.

On adapting NTRU for Post-Quantum Public-Key Encryption

Simone Dutto[1], Guglielmo Morgari[2], and Edoardo Signorini[1,2]

[1] Department of Mathematical Sciences L. Lagrange, Politecnico of Torino, Italy
[2] Telsy Spa, Italy
simone.dutto@polito.it, guglielmo.morgari@telsy.it,
edoardo.signorini@polito.it

1 Introduction

The main research project on Post-Quantum Cryptography (PQC) is the NIST PQC Standardization Process [1], which focuses on selecting post-quantum Key Encapsulation Mechanisms (KEMs) and Digital Signature Schemes.

Public-Key Encryption (PKE) schemes will not be standardized since, in general, the submitted KEMs are obtained from PKE schemes and the inverse process is simple. However, there are cases for which this is not straightforward, like the NTRU submission [2].

This work focuses on solving this problem by introducing a PKE scheme obtained from the KEM proposed in the NTRU submission, while maintaining its IND-CCA2 security (indistinguishability under adaptive chosen ciphertext attack).

2 The NTRU submission

NTRU [2] is a finalist in the NIST PQC Standardization Process that presents a KEM in which the security is based on the Shortest Vector Problem (SVP) on the NTRU lattice [6] (several variations and generalization exist, see e.g. [8, 4, 5]).

The proposed KEM achieves IND-CCA2 security by exploiting an OW-CPA (one-wayness under chosen plaintext attack) PKE scheme with either of two sets of parameters: NTRU-HPS and NTRU-HRSS-KEM. Because of its larger range of addressed security levels, we focused on NTRU-HPS.
More precisely, let $\mathbb{Z}_q = \{-\frac{q}{2}, \ldots, 0, \ldots, \frac{q}{2} - 1\}$ and $\mathbb{Z}_3 = \{-1, 0, 1\}$, given

$$(n, q) \in \{(509, 2048), (677, 2048), (821, 4096)\}$$

corresponding to three different security levels, with $p = 3$ and $d = \frac{q}{8} - 2$, and considering $\phi_n(x) = 1 + x + \ldots + x^{n-1} \in \mathbb{Z}_3[x]$ or $\phi_n(x) \in \mathbb{Z}_q[x]$, all polynomials

De Cifris Koine – DE CIFRIS SEMINARS –https://doi.org/10.69091/koine/vol-2-T04

are represented as arrays in the three following rings:
- $R_q = \mathbb{Z}_q[x]/(x^n - 1)$,
- $S_q = \mathbb{Z}_q[x]/(\phi_n)$,
- $T = \mathbb{Z}_3[x]/(\phi_n)$.

Moreover, $T(d) = \left\{ \sum_{i=1}^{n-2} t_i x^i \in T \,|\, \#\{t_i = 1\} = \#\{t_i = -1\} = d/2 \right\}$.
The OW-CPA PKE scheme consists of the following algorithms.

$\underline{\mathsf{OWCPA.keygen}(seed)}$	$\underline{\mathsf{OWCPA.encrypt}(\mathbf{h}, (\mathbf{r}, \mathbf{m}))}$
1. $seed \to \mathbf{f} \in T$	1. return $\mathbf{c} = \mathbf{r} \cdot \mathbf{h} + \mathbf{m} \in R_q$
2. $seed \to \mathbf{g} \in T(d)$	$\underline{\mathsf{OWCPA.decrypt}((\mathbf{f}, \mathbf{f}_3, \mathbf{h}_q), \mathbf{c})}$
3. $\mathbf{f}_q = \mathbf{f}^{-1} \in S_q$	1. $\mathbf{a} = \mathbf{c} \cdot \mathbf{f} \in R_q$
4. $\mathbf{h} = 3\mathbf{g} \cdot \mathbf{f}_q \in R_q$	2. $\mathbf{m} = \mathbf{a} \cdot \mathbf{f}_3 \in T$
5. $\mathbf{h}_q = \mathbf{h}^{-1} \in S_q$	3. $\mathbf{r} = (\mathbf{c} - \mathbf{m}) \cdot \mathbf{h}_q \in S_q$
6. $\mathbf{f}_3 = \mathbf{f}^{-1} \in T$	4. if $(\mathbf{r}, \mathbf{m}) \in T \times T(d)$ return $(\mathbf{r}, \mathbf{m}, 0)$
7. return $\mathbf{h}$, $(\mathbf{f}, \mathbf{f}_3, \mathbf{h}_q)$	5. else return (0, 0, 1)

Since the message $m \in T(d)$ is ternary and constrained, and the security is only OW-CPA, this is not directly suitable for PKE.

3 Obtaining a PKE scheme from NTRU

In order to obtain a PKE scheme using the OW-CPA PKE scheme from NTRU [2], another work is considered: NTRUEncrypt [3], a first round submission that inspired NTRU-HPS in NTRU [2].

NTRUEncrypt exploits a padding function to encode the message in a ternary polynomial with at least 256 bits of entropy and a message masking to achieve IND-CPA security (indistinguishability under chosen plaintext attack). Then, the NAEP transformation [7] is adopted to obtain an IND-CCA2 PKE scheme.

3.1 Message padding function

The padding is an invertible map $\mathrm{Pad} : (\mathbb{Z}_{2^8})^L \times \{0,1\}^* \to T(d)$ with $\mathrm{Pad}(msg, seed) = m$, where the *seed* allows to add bits of entropy. Since in NTRU-HPS $m \in T(d)$, while in NTRUEncrypt it has no constraints, a new padding is required. Our definition uses an encoding function to obtain a ternary polynomial, and then exploits the *seed* to add bits of entropy and achieve the constraint.

Considering a bijection $\zeta : (\mathbb{Z}_2)^5 \to (\mathbb{Z}_3)^4$ with outputs among the permutations of the elements of the arrays $(0, 0, 1, -1)$, $(0, 1, 1, -1)$, $(0, 1, -1, -1)$, our encoding function is $\underline{\zeta} : (\mathbb{Z}_{2^8})^L \to (\mathbb{Z}_3)^{32\,L/5}$, with

$$\underline{\zeta}(m_1, \ldots, m_L) = \zeta\big(m_1[1{:}5]\big) \| \zeta\big(m_1[6{:}8] \| m_2[1{:}2]\big) \| \ldots \| \zeta\big(m_L[4{:}8]\big) .$$

In the encoded message

$$8 \cdot L/5 \leqslant \#\{1's\}, \#\{-1's\} \leqslant 16 \cdot L/5\,,$$

so that the maximum length of msg is $L \in \mathbb{N}$ such that 5 divides L and $16\,L/5 \leqslant d/2$, i.e. $L = 5\lfloor d/32 \rfloor$.

The last $r = n - 1 - 32\,L/5$ coefficients are generated through the *seed*, while reaching the constraint and adding at least 256 bits of entropy. In the worst cases, the missing 1's and -1's are $a = d/2 - 16\,L/5$ and $b = d/2 - 8\,L/5$ (or viceversa). Thus, the possible completions are $\binom{r}{a}\binom{r-a}{b}$ and the minimum entropy is

$$H_{\min} = \log_2\left(\binom{r}{a}\binom{r-a}{b}\right).$$

The obtained results are:
- for $n = 509, q = 2048, L = 35, H_{\min} = 301$;
- for $n = 677, q = 2048, L = 35, H_{\min} = 367$;
- for $n = 821, q = 4096, L = 75, H_{\min} = 399$.

Finally, Pad^{-1} takes the first $32\,L/5$ entries and applies the inverse of $\underline{\zeta}$. Its output is always a byte array of length L.

3.2 Message masking

As in NTRUEncrypt, the IND-CPA security is achieved by masking the message. However, the only way to mask $m = \mathrm{Pad}(msg, seed) \in T(d)$ while maintaining the constraint is to apply a permutation, which is not secure. Thus, the message is masked before the padding function using the digest of the required random polynomial $r \in T$, resulting in

$$m = \mathrm{Pad}(msg \oplus \mathrm{Hash}(r), seed) \in T(d)\,.$$

3.3 The PKE scheme

The algorithm for key generation is OW-CPA.keygen from NTRU.

To encrypt $msg \in (\mathbb{Z}_{2^8})^L$ using the public key h, the steps are:

- to sample $r \in T$, obtain $m = \mathrm{Pad}(msg \oplus \mathrm{Hash}(r), seed) \in T(d)$;
- to return $c = \text{OW-CPA.encrypt}(h, r, m)$.

To decrypt c with the secret key (f, f_3, h_q), the algorithm procceds as follows:

- $(r, m, fail) = \text{OW-CPA.decrypt}(f, f_3, h_q, c)$;
- if $fail = 0$ then return $msg = \mathrm{Pad}^{-1}(m) \oplus \mathrm{Hash}(r)$, else return $\perp$.

4 Conclusions

In this work a IND-CCA2 PKE scheme is obtained from the KEM in the NTRU [2] submission to the NIST PQC Standardization Process. Inspired by NTRUEncrypt, the NAEP transformation is used and two new functions are introduced: i) a padding function that adds more than 256 bits of entropy and encodes messages of 35 or 75 bytes depending on the security level; ii) a message masking that gives IND-CPA security to the resulting scheme. Performance and data-size of the obtained PKE scheme are analogous to those of the KEM in NTRU (benchmarks available in [9]), making the PKE scheme a valid post-quantum alternative.

References

1. G. Alagic, J. Alperin-Sheriff, D. Apon, D. Cooper, Q. Dang, J. Kesley, Y.-K. Liu, C. Miller, D. Moody, R. Peralta, R. Perlner, A. Robinson, and D. Smith-Tone. NIST Internal Report 8309: Status Report on the Second Round of the NIST Post-Quantum Cryptography Standardization Process. Technical report, NIST, 2020.
2. C. Chen, O. Danba, J. Hoffstein, A. Hülsing, J. Rijneveld, J. Schanck, P. Schwabe, W. Whyte, and Z. Zhang. NIST PQC Standardization Round 2 Submission: NTRU, Algorithm Specifications And Supporting Documentation. Technical report, NIST, 2019.
3. C. Chen, J. Hoffstein, W. Whyte, and Z. Zhang. NIST PQC Standardization Round 1 Submission - NTRUEncrypt, a lattice based encryption algorithm. Technical report, NIST, 2017.
4. Y. Doröz and B. Sunar. Flattening NTRU for evaluation key free homomorphic encryption. *Journal of Mathematical Cryptology*, 14(1):66–83, 2020.
5. J. Hoffstein, J. H. Silverman, W. Whyte, and Z. Zhang. A signature scheme from the finite field isomorphism problem. *Journal of Mathematical Cryptology*, 14(1):39–54, 2020.
6. Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. NTRU: A ring-based public key cryptosystem. In *Third Algorithmic Number Theory Symposium (ANTS)*, volume 1423 of *LNCS*, pages 267–288. Springer, Heidelberg, June 1998.
7. Nick Howgrave-Graham, Joseph H. Silverman, Ari Singer, and William Whyte. NAEP: Provable security in the presence of decryption failures. Cryptology ePrint Archive, Report 2003/172, 2003. `https://eprint.iacr.org/2003/172`.
8. G. De Micheli, N. Heninger, and B. Shani. Characterizing overstretched NTRU attacks. *Journal of Mathematical Cryptology*, 14(1):110–119, 2020.
9. VAMPIRE working groups. eBACS: ECRYPT Benchmarking of Cryptographic Systems - Measurements of key-encapsulation mechanisms, indexed by machine. Technical report, VAMPIRE: Virtual Applications and Implementations Research Lab, 2020.

Cryptanalysis of round-reduced AES

Lorenzo Grassi

Department of Computer Science, Ruhr University Bochum, Germany
lgrassi@science.ru.nl

AES is the best known and most widely used secret key cryptosystem, and determining its security is one of the most important problems in cryptanalysis. Since no known attack can break the full AES significantly faster than via exhaustive search, researchers have concentrated on attacks which can break reduced round versions of AES. While such attacks do not pose any practical threat to the AES, they give new insights in the cipher that is probably responsible for the largest fraction of encrypted data worldwide.

Such attacks are important for several reasons. First of all, they enable us to assess the remaining security margin of AES, defined by the ratio between the number of rounds which can be successfully attacked and the number of rounds in the full AES. In addition, there are many proposals for using reduced round AES (and especially its 4 or 5 rounds versions) as components in larger schemes, and thus successful cryptanalysis of these variants can be used to attack those schemes. Finally, new cryptanalysis techniques can enable us to develop new attack strategies which may become increasingly potent with additional improvements. In most of the cases, it took several years – and a series of subsequent improvements – from the invention of the technique until it was developed into its current form. As a concrete example, consider the impossible differential cryptanalysis on AES. When it was proposed in 2001 by Biham and Keller [2], the impossible differential attack could attack ("only") 5 rounds of AES and it was not competitive with respect to others attacks, as the integral one. It took approximately 6 years before that attack was extended and set up against 7-round AES-128 [15], becoming one of the few attacks on such number of rounds. Finally, only recently Boura *et al.* [5] improved it into its best currently known variant which breaks 7-round AES with an overall complexity of about 2^{107}.

In the follow-up we are going to recall some of our recent results regarding the cryptanalysis of round-reduced AES: among others, here we focus on the "multiple-of-8" [14] and on the mixture differential [10] properties, which are the starting point for new competitive secret-key distinguishers and key-recovery attacks.

De Cifris Koine – DE CIFRIS SEMINARS –https://doi.org/10.69091/koine/vol-2-T05

AES: Brief Description & State of the Art

AES [6] is a *Substitution-Permutation network* that supports key sizes of 128, 192, and 256 bits. The 128-bit plaintext initializes the internal state as a 4×4 matrix of bytes as values in the finite field $GF(2^8)$. Depending on the version of AES, N_r rounds are applied to the state, where $N_r = 10$ for AES-128, $N_r = 12$ for AES-192, and $N_r = 14$ for AES-256. An AES round $R(x) = K \oplus MC \circ SR \circ$ S-Box(x) applies four operations to the state matrix:

SubBytes (S-Box) – applying the same 8-bit to 8-bit invertible non-linear S-Box 16 times in parallel on each byte of the state;
ShiftRows (SR) – cyclic shift of each row to the left;
MixColumns (MC) – multiplication of each column by a constant 4×4 MDS matrix;
AddRoundKey (ARK) – XORing the state with a 128-bit subkey.

In the first round an additional AddRoundKey operation (using a whitening key) is applied, and in the last round the MixColumns operation is omitted.

State of the Art. Before our works, secret-key distinguishers[1] that are independent of the key were known for 2-, 3- and 4-round AES. They include truncated differential distinguishers for 2 and 3 rounds (see e.g. [13] for details), impossible differential distinguisher [2] for 4 rounds and integral distinguishers [7] for 3 and 4 rounds.

Providing a list of all possible key-recovery attacks on round-reduced AES present in the literature is out of our scope: here we limit ourselves to make few considerations. Many of the key-recovery attacks present in the literature are set up by extending the secret-key distinguishers just mentioned. Focusing on AES-128, concrete examples include the integral attacks on 6-round AES-128 [7, 6] and impossible differential attacks on 7-round AES-128 [2, 5]. Another powerful attack against AES is based on the Meet-in-the-Middle approach: it can cover up to 7-round AES-128 [8, 9], and combined with the bicycle technique [3] it covers all 10-round AES-128 with a computational cost a bit smaller than that required for brute force.

[1] In a secret-key distinguisher, the attacker is given access to both the cipher with a uniformly randomly chosen key and to a function that has been chosen uniformly at random from all invertible mappings from the plaintext space to the ciphertext space. The goal of the attacker is then to determine which of the two is the cipher and which is the random function.

Our Contribution

While in the last recent years cryptanalysis has mainly focused on maximizing the number of rounds that can be broken without exhausting the full codebook and key space, we considered a different point of view. Instead of focusing/improving already existing attacks, we have tried to propose *new* methods of cryptanalysis. Even if such new methods are not always more competitive than the ones already present in the literature, such new directions in cryptanalysis can be interesting from a research point of view in order to better understand the ciphers that are in used. Moreover, it is not possible to exclude a priori that, when such methods will reach their full potential, they can beat the existing attacks.

Multiple-of-8 Property and Truncated Differential for 5-round AES. Since the development of cryptanalysis of AES and AES-like constructions in the late 1990s, the set of inputs (or a subset of it) which differ only in one diagonal has special importance. It appeared in various (truncated) differential, integral, and impossible differential attacks (among others) for up to 4-round AES. For the first time in the literature, we proposed a precise theoretical analysis of the probability distribution of such set after 5-round AES.

In more details, given a diagonal set of 2^{32} plaintexts which differ only in one diagonal, we studied the probabilistic distribution of the number of different pairs of ciphertexts that are equal in one fixed anti-diagonal (or equivalently, that lie in certain subspaces) after 5 rounds of AES – denoted as "number of collisions" in the following. For the first time, we are able to show that independently of the secret key:

- the number of collisions is *always* a multiple of 8;
- the number of collisions is on average (a little) bigger compared to the case in which the ciphertexts are generated by a random permutation;
- besides the mean, also the variance of such a distribution is (much) higher than for a random permutation.

To show and prove these, we developed new theoretical approaches. Practical implementations and verification confirm our analysis.

Similar results can be exploited to set up new secret-key distinguishers, including the *"multiple-of-n" distinguisher* [14] and the *first truncated differential distinguisher based on the variance* [12], besides new key-recovery attacks. We remark that *these are the first secret-key distinguishers on 5-round AES which are independent of the secret key, improving over a 20 year old result on 4 rounds.*

Mixture Differential Cryptanalysis. At first it was not clear whether the "multiple-of-8" property/distinguisher could at all lead to attacks on AES which are competitive with respect to previously known results. In [10] appeared at FSE/-ToSC'19, we partially solved this question, by developing a new type of attacks and distinguishers – called "*mixture differential cryptanalysis*" – on round-reduced AES-like ciphers, a way to translate the (complex) "multiple-of-8" 5-round distinguisher into a simpler and more convenient one (though, on a smaller number of rounds). Given a pair of chosen plaintexts, the idea is to construct new pairs of plaintexts by mixing the generating variables of the original pair of plaintexts. In [10] we theoretically proved that for 4-round AES the corresponding ciphertexts of the original pair of plaintexts are equal in certain anti-diagonals (or equivalently, lie in a particular subspace) if and only if the corresponding pairs of ciphertexts of the new pairs of plaintexts have the same property. Such distinguisher has been recently revisited in [4], where its authors show that the above property is an immediate consequence of an equivalence relation on the input pairs, under which the difference at the output of the round function is invariant.

This secret-key distinguisher can be extended into a new key-recovery attack on 5-round AES-128 (and 6-round AES-128 [11]), which has been later on improved in [1], becoming the current attack with the lowest computational cost among the attacks currently present in the literature (that do not use adaptive chosen plaintexts/ciphertexts). By extending this technique to larger versions of AES, authors also obtained new attacks on AES-192 and AES-256 which have the best time complexity among all the attacks on 7-round AES which have practical data and memory complexities.

References

1. Achiya Bar-On, Orr Dunkelman, Nathan Keller, Eyal Ronen, and Adi Shamir. Improved key recovery attacks on reduced-round AES with practical data and memory complexities. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part II*, volume 10992 of *LNCS*, pages 185–212. Springer, Heidelberg, August 2018.
2. E. Biham and N. Keller. Cryptanalysis of Reduced Variants of Rijndael. unpublished, 2001.
3. Andrey Bogdanov, Dmitry Khovratovich, and Christian Rechberger. Biclique cryptanalysis of the full AES. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 344–371. Springer, Heidelberg, December 2011.
4. Christina Boura, Anne Canteaut, and Daniel Coggia. A general proof framework for recent AES distinguishers. *IACR Trans. Symm. Cryptol.*, 2019(1):170–191, 2019.
5. Christina Boura, Virginie Lallemand, María Naya-Plasencia, and Valentin Suder. Making the impossible possible. *Journal of Cryptology*, 31(1):101–133, January 2018.
6. J. Daemen and V. Rijmen. *The Design of Rijndael: AES - The Advanced Encryption Standard*. Springer, 2002.
7. Joan Daemen, Lars R. Knudsen, and Vincent Rijmen. The block cipher Square. In Eli Biham, editor, *FSE'97*, volume 1267 of *LNCS*, pages 149–165. Springer, Heidelberg, January 1997.
8. Patrick Derbez and Pierre-Alain Fouque. Exhausting Demirci-Selçuk meet-in-the-middle attacks against reduced-round AES. In Shiho Moriai, editor, *FSE 2013*, volume 8424 of *LNCS*, pages 541–560. Springer, Heidelberg, March 2014.
9. Patrick Derbez, Pierre-Alain Fouque, and Jérémy Jean. Improved key recovery attacks on reduced-round AES in the single-key setting. In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 371–387. Springer, Heidelberg, May 2013.
10. Lorenzo Grassi. Mixture differential cryptanalysis: a new approach to distinguishers and attacks on round-reduced AES. *IACR Trans. Symm. Cryptol.*, 2018(2):133–160, 2018.
11. Lorenzo Grassi. Probabilistic mixture differential cryptanalysis on round-reduced AES. In Kenneth G. Paterson and Douglas Stebila, editors, *SAC 2019*, volume 11959 of *LNCS*, pages 53–84. Springer, Heidelberg, August 2019.
12. Lorenzo Grassi and Christian Rechberger. New rigorous analysis of truncated differentials for 5-round AES. Cryptology ePrint Archive, Report 2018/182, 2018. `https://eprint.iacr.org/2018/182`.
13. Lorenzo Grassi, Christian Rechberger, and Sondre Rønjom. Subspace trail cryptanalysis and its applications to AES. *IACR Trans. Symm. Cryptol.*, 2016(2):192–225, 2016. `https://tosc.iacr.org/index.php/ToSC/article/view/571`.
14. Lorenzo Grassi, Christian Rechberger, and Sondre Rønjom. A new structural-differential property of 5-round AES. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *EUROCRYPT 2017, Part II*, volume 10211 of *LNCS*, pages 289–317. Springer, Heidelberg, April / May 2017.
15. R. C. Phan. Impossible differential cryptanalysis of 7-round Advanced Encryption Standard (AES). *Information Processing Letters*, 91(1):33–38, 2004.

Shift Invariance in Symmetric Cryptography

Luca Mariot

Semantics, Cybersecurity and Services Group, University of Twente, Netherlands
info@lucamariot.org

Introduction

Symmetric encryption is usually reputed to be the workhorse of cryptography. Indeed, the design of complex cryptographic protocols often relies upon the use of symmetric ciphers as secure building blocks to provide confidentiality. For this reason, in the last decades a large body of research has focused on the development of stream and block ciphers that are both secure and efficient. The concept of shift invariance plays a significant role in this respect. Loosely speaking, a transformation over a binary vector space is shift invariant if a translation of the input state results in the output state being translated by the same amount. This property yields several advantages when designing symmetric cryptographic primitives such as Boolean functions and S-boxes: as noted by Daemen et al. [4], shift invariant binary transformations can be realized by an array of interconnected processors, all of which implement the same Boolean function. This feature allows for very efficient implementations both in hardware and in software, and moreover the security analysis of the resulting primitive is often simplified.

The aim of this extended abstract is to give a brief overview on the main contributions in the literature concerning the use of shift invariant transformations to design symmetric cryptographic primitives. We start by introducing in Section 6 the related concept of cellular automata, the model of discrete dynamical systems embodying shift invariance. We then review the main results regarding the use of cellular automata in stream and block ciphers in Section 6.

Shift Invariant Transformations and Cellular Automata

In what follows, we adopt the usual notation for Boolean functions used in cryptography and coding theory; we refer the reader to [2] for more details.
Let $\mathbb{F}_2 = \{0, 1\}$ be the finite field with two elements, and for $n \in \mathbb{N}$ let $(\mathbb{F}_2)^n$ be the n-dimensional vector space over $\mathbb{F}_2$, or equivalently the set of all n-bit strings. Given $x = (x_0, \cdots x_{n-1}) \in (\mathbb{F}_2)^n$, the *shift* operator over x is defined as $\sigma(x) = (x_1, \cdots, x_{n-1}, x_0)$; in other words, σ rotates all coordinates of x one place

De Cifris Koine – DE CIFRIS SEMINARS –https://doi.org/10.69091/koine/vol-2-T06

to the left. A vectorial Boolean function $F : (\mathbb{F}_2)^n \to (\mathbb{F}_2)^n$ is called *shift invariant* if it commutes with σ, i.e. $\sigma(F(x)) = F(\sigma(x))$, for all $x \in \mathbb{F}_2^n$.

Cellular Automata (CA) are a computational model closely related to shift invariant transformations, in which the global output is given by evaluating in parallel a single local update rule over all components (or cells) arranged over an array. More formally, a one-dimensional CA of length n, diameter d, and local rule $f : (\mathbb{F}_2)^d \to \mathbb{F}_2$ is defined by a vectorial function $F : (\mathbb{F}_2)^n \to (\mathbb{F}_2)^n$ where for all $x \in (\mathbb{F}_2)^n$ and $0 \leqslant i \leqslant n-1$ the i-th component of the output $F(x)$ is defined as $F(x)_i = f(x_i, x_{i+1}, \cdots, x_{i+d-1})$. The reason why the cells are numbered from 0 to $n-1$ is that all indices are taken modulo n. Hence, a CA is composed of a circular binary vector where each cell, or coordinate, updates its state in parallel by applying the local rule f on the neighborhood formed by itself and the $d-1$ cells on its right, with periodic boundary conditions.

CA are actually an equivalent formulation of shift invariant transformations. In fact, one can show that if $F : (\mathbb{F}_2)^n \to (\mathbb{F}_2)^n$ is shift invariant then each of its coordinates is described by the same local update rule of diameter $d = n$. This is a consequence of the so-called *Curtis-Hedlund-Lyndon theorem*, for which the reader may find more details in [7].

CA-based design of Stream and Block Ciphers

Most of the literature related to CA focuses on their long-term behavior, by considering them as a particular kind of discrete dynamical systems. This approach was followed also in the earlier research effort on CA-based cryptography, such as Wolfram's pseudorandom number generator [12]. In Wolfram's idea, a CA equipped with a local rule of $d = 3$ variables was iterated several times starting from a random initial condition. In particular, Wolfram selected *rule 30*, defined as

$$f(x_0, x_1, x_2) = x_0 \text{ XOR } (x_1 \text{ OR } x_2)\,.$$

Due to the chaotic patterns emerging from the global behavior of *rule 30*, Wolfram proposed to use the sequence of states taken by the CA central cell as a keystream for a stream cipher. However, later research [10, 9] showed that rule 30 suffered from some cryptographic weaknesses and the resulting CA was thus unsuitable as a keystream generator. A more recent work [8] considered the search of rules of higher diameters and better cryptographic properties, but this research thread has somewhat faded over the last years.

Another perspective in CA-based cryptography, which proved to be more fruitful, is to focus only on the short-term behavior of a CA, i.e. to consider the vectorial Boolean function from a single CA iteration as an S-box to be used in the confusion layer of a block cipher. This research line was pioneered in the mid 90s by Daemen

et al. [4]. There, the authors showed that the CA local rule χ of $d = 3$ variables, defined as $\chi(x_0, x_1, x_2) = x_0$ XOR(x_1 AND (NOT(x_2))), resulted in an invertible shift invariant transformation $F : (\mathbb{F}_2)^n \to (\mathbb{F}_2)^n$ for any odd length n of the state array, a crucial property to enable decryption. Rule χ also turned out to have a simple description in terms of correlation and propagation characteristics, which made it interesting for cryptographic purposes. Indeed, this CA appeared in the confusion layer of several ciphers and hash functions, such as Panama [5]. Moreover, the CA rule χ is the only nonlinear component used in Keccak [1], the primitive based on the sponge construction that has been adopted as the SHA-3 standard for cryptographic hash functions. In this case, rule χ is applied on a CA of length $n = 5$, and the resulting S-box yields nonlinearity and differential uniformity values close to the optimal ones.

A more recent research thread considers the search of S-boxes arising from cellular automata both from the standpoint of their cryptographic properties and their implementation cost. Along this line, Picek et al. [11] used a metaheuristic optimization algorithm, namely Genetic Programming (GP), to search for CA-based S-boxes with sizes between 5×5 and 8×8. With this technique, the authors managed to find optimal S-boxes concerning the nonlinearity and differential uniformity properties up to size 7×7, and having hardware implementation costs similar to those of other state-of-the-art S-boxes. Mariot et al. [3] performed a systematic theoretical investigation of the cryptographic properties of CA-based S-boxes and developed a reverse-engineering approach based on GP to find what is the shortest CA rule resulting in a specific S-box.

Finally, more recently Gao et al. [6] proposed an S-box construction based on the iteration of CA with quadratic rules, i.e. those rules whose algebraic normal form has degree 2. In particular, the authors exploited the shift-invariance property of such CA to obtain a threshold implementation of the resulting S-boxes, thus providing a further countermeasure towards side-channel attacks.

References

1. G. Bertoni, J. Daemen, M. Peeters, and G. V. Assche. *The* Keccak *reference*, January 2011.
2. C. Carlet. *Boolean Functions for Cryptography and Coding Theory*. Cambridge University Press, 2020.
3. Jakobovic D., Leporati A., Mariot L., and Picek S. Cellular automata based S-boxes. *Cryptography and Communications*, 11(1):41–62, 2019.
4. J. Daemen, R. Govaerts, and J. Vandewalle. Invertible shift-invariant transformations on binary arrays. *Applied Mathematics and Computation*, 62(2):259–277, 1994.
5. Joan Daemen and Craig S. K. Clapp. Fast hashing and stream encryption with PANAMA. In Serge Vaudenay, editor, *FSE'98*, volume 1372 of *LNCS*, pages 60–74. Springer, Heidelberg, March 1998.
6. Si Gao, Arnab Roy, and Elisabeth Oswald. Constructing TI-friendly substitution boxes using shift-invariant permutations. In Mitsuru Matsui, editor, *CT-RSA 2019*, volume 11405 of *LNCS*, pages 433–452. Springer, Heidelberg, March 2019.
7. J. Kari. Theory of cellular automata: A survey. *Theoretical Computer Science*, 334(1-3):3–33, 2005.
8. A. Leporati and L. Mariot. Cryptographic properties of bipermutive cellular automata rules. *Journal of Cellular Automata*, 9(5-6):437–475, 2014.
9. B. Martin. A Walsh Exploration of Elementary CA Rules. *Journal of Cellular Automata*, 3(2):145–156, 2008.
10. Willi Meier and Othmar Staffelbach. Analysis of pseudo random sequence generated by cellular automata. In Donald W. Davies, editor, *EUROCRYPT'91*, volume 547 of *LNCS*, pages 186–199. Springer, Heidelberg, April 1991.
11. S. Picek, L. Mariot, B. Yang, D. Jakobovic, and N. Mentens. Design of S-boxes defined with cellular automata rules. In *Computing Frontiers Conference '17'*, page 409–414, 2017.
12. Stephen Wolfram. Cryptography with cellular automata. In Hugh C. Williams, editor, *CRYPTO'85*, volume 218 of *LNCS*, pages 429–432. Springer, Heidelberg, August 1986.

An investigation on integer factorization applied to public-key cryptography

Giordano Santilli

Department of Mathematics, University of Trento, Italy
giordano.santilli@unitn.it

Introduction

The Integer Factorization Problem (IFP) consists in finding the factorization of a given integer. This problem underlies many public-key cryptographic protocols such as RSA [6] or Rabin's cryptosystem [5]. In this context, we will consider semiprimes, i. e., $N = p \cdot q$, where p and q are primes and $p < q$. The most powerful classical factorization algorithm is the General Number Field Sieve (GNFS) [1] [4], [3] (see also [9]), which exploits the structure of the subring $\mathbb{Z}[\theta]$ of the ring of integers $\mathfrak{O}$ of the number field $\mathbb{Q}(\theta)$, where θ is linked to N. In particular, the main objects employed in GNFS are the first-degree prime ideals (FDPIs) of $\mathbb{Z}[\theta]$, namely the prime ideals $\mathfrak{p}$ such that $\mathcal{N}(\mathfrak{p}) = p$, where p is a prime and $\mathcal{N}$ denotes the norm of an ideal, defined as the number of elements in the quotient $\mathbb{Z}[\theta]/\mathfrak{p}$ (see [10] for further details). These ideals can be identified with pairs of integers using the following:

Theorem 1 *[1, p. 52] Let $f \in \mathbb{Z}[x]$ be a monic irreducible polynomial and θ one of its roots. Then, for every positive prime p there exists a bijection between*

$$\{(r, p) \,|\, r \in \mathbb{Z}_p \text{ and } f(r) \equiv 0 \bmod p\}$$

and

$$\{\mathfrak{p} \,|\, \mathfrak{p} \in \mathrm{Spec}(\mathbb{Z}[\theta]) \text{ and } \mathcal{N}(\mathfrak{p}) = p\}\,.$$

Remember also that given two ideals $\mathfrak{a}, \mathfrak{b} \subseteq \mathfrak{O}$, then $\mathfrak{a}|\mathfrak{b}$ if and only if $\mathfrak{b} \subseteq \mathfrak{a}$ ([10]).

In my Ph. D. thesis ([7]) I have studied several ways to attack IFP. In this abstract I will present two of them. The first one is a method developed by noting a pattern in the sequence of successive remainders (of a given integer). The second one is a procedure to speed up the search for FDPIs of a biquadratic extension, by considering the FDPIs of the rings of integers (of the underlying quadratic fields).

De Cifris Koine – DE CIFRIS SEMINARS –https://doi.org/10.69091/koine/vol-2-T07

Recursion in successive moduli

Let $m \in \mathbb{N}^+$ be such that $\left\lfloor \sqrt{\frac{N}{2}} \right\rfloor \leqslant m \leqslant \lfloor \sqrt{N} \rfloor$ and suppose that

$$\begin{cases} N \equiv a_0 \bmod m \\ N \equiv a_1 \bmod (m+1) \\ N \equiv a_2 \bmod (m+2), \end{cases} \qquad \text{where } a_0, a_1, a_2 \in \mathbb{N}^+ \text{are monotonic.}$$

Then, define $k := a_1 - a_0$ and w the smallest positive integer such that $w :\equiv a_2 - 2a_1 + a_0 \bmod (m+2)$. It can be proved, just using the properties of the floor and the ceiling functions (see e. g., [2, chapter 3]), that the remainder w assumes only fixed values:

Theorem 2 *In the above setting, if $N \geqslant 50$, then $w \in \{2, 4, 6\}$. Moreover, if there exists a suitable value of m such that $\left\lfloor \sqrt{\frac{N}{2}} + 1 \right\rfloor \leqslant m \leqslant \lfloor \sqrt{N} \rfloor - 1$, then $w = 4$.*

Therefore, since w is a fixed value, it is possible to obtain a formula for the successive remainders of N:

Theorem 3 *If $N \geqslant 50$, then for every $i \in \mathbb{N}$,*

$$N \equiv \left(a_0 + ik + w\frac{i(i-1)}{2} \right) \bmod (m+i).$$

Moreover if m is such that $\left\lfloor \sqrt{\frac{N}{2}} + 1 \right\rfloor \leqslant m \leqslant \lfloor \sqrt{N} \rfloor - 1$, then, for every $i \in \mathbb{N}$,

$$N \equiv \left(a_0 + ik + 2i^2 - 2i \right) \bmod (m+i).$$

The same result may be achieved considering the interpolating polynomial f of degree 2 such that $f(0) = a_0$, $f(1) = a_1$, $f(2) = a_2$. In fact it is possible to prove that in this case $N \equiv f(i) \bmod (m+i)$. However, as shown in [7], this formula does not give an immediate solution of the IFP. In fact, the following proposition holds.

Theorem 4 *Producing a factorization for N is equivalent to finding an integer $i \in \mathbb{N}^+$ for which $N \equiv \left(a_0 + ik + 2i^2 - 2i \right) \equiv 0 \bmod m + i$.*

First-degree prime ideals in biquadratic fields

The results presented in this section are published in [8]. Consider two irreducible polynomials $f_a(x) = x^2 - a$ and $f_b(x) = x^2 - b$ in $\mathbb{Z}[x]$ with $\alpha^2 = a$ and $\beta^2 = b$. Then, the biquadratic extension of degree 4 that contains both $\mathbb{Q}(\alpha)$ and $\mathbb{Q}(\beta)$ is $\mathbb{Q}(\theta) = \mathbb{Q}(\alpha+\beta)$ and the minimal polynomial of θ is $f_c(x) = x^4 - 2(a+b)x^2 + (a-b)^2$, ([11]). In this situation, it is possible to define a link between the FDPIs in $\mathbb{Z}[\theta]$ and those in $\mathbb{Z}[\alpha]$ and $\mathbb{Z}[\beta]$.

Theorem 5 *Let (r,p) be a FDPI of $\mathbb{Z}[\alpha]$ and (s,p) a FDPI of $\mathbb{Z}[\beta]$, then $(r+s,p)$ is a FDPI of $\mathbb{Z}[\theta]$. Vice-versa if (t,p) is a FDPI in $\mathbb{Z}[\theta]$ and either $p=2$ or $t\neq 0$, then there exists a unique pair $r,s\in\mathbb{Z}_p$, such that $t\equiv r+s \bmod p$ and (r,p) and (s,p) are FDPI in $\mathbb{Z}[\alpha]$ and $\mathbb{Z}[\beta]$, respectively.*

The notion of ideal divisibility may also be studied in terms of the FDPI in the quadratic extensions, due to the following:

Theorem 6 *Let $n,m\in\mathbb{Z}$ be such that $(m,n)=1$ and $m\neq 0$ and $I=\langle a+b\theta\rangle$, an ideal in $\mathbb{Z}[\theta]$. Then $I\cap\mathbb{Z}[\alpha]$ is a principal ideal and*

$$I\cap\mathbb{Z}[\alpha]=\left\langle n^2-bm^2+am^2+2nm\alpha\right\rangle.$$

We conclude with the following theorem.

Theorem 7 *Let n and m be coprime integers and $I=\langle n+m\theta\rangle$ be a principal ideal of $\mathbb{Z}[\theta]$. Suppose that (r,p) and (s,p) are FDPIs of $\mathbb{Z}[\alpha]$ and $\mathbb{Z}[\beta]$ respectively such that $(r,p)|I\cap\mathbb{Z}[\alpha]$ and $(s,p)|I\cap\mathbb{Z}[\beta]$. Then $(r+s,p)\subseteq\mathbb{Z}[\theta]$ divides I unless simultaneously $p\neq 2$, $n\equiv 0 \bmod p$ and $r\not\equiv -s \bmod p$. Vice-versa, if (t,p) is a FDPI of $\mathbb{Z}[\theta]$ that divides I and $t\neq 0$ if $p\neq 2$, then there exist two unique FDPIs $(r,p)\subseteq\mathbb{Z}[\alpha]$ and $(s,p)\subseteq\mathbb{Z}[\beta]$ such that $(r,p)|I\cap\mathbb{Z}[\alpha]$, $(s,p)|I\cap\mathbb{Z}[\beta]$ and $t\equiv r+s \bmod p$.*

Acknowledgements

The recursion in successive moduli was first noted by Matteo Piva, under the supervision of Massimiliano Sala, who was also the supervisor of my PHD thesis.

References

1. J. P. Buhler, H. W. Lenstra, and C. Pomerance. Factoring integers with the number field sieve. In *The development of the number field sieve*, volume 1554 of *Lecture Notes in Mathematics*, pages 50–94. Springer, 1993.
2. R. L. Graham, D. E. Knuth, and O. Patashnik. *Concrete Mathematics: A Foundation for Computer Science*. Addison Wesley, 1994.
3. A. K. Lenstra and H. W. Lenstra, editors. *The Development of the Number Field Sieve*, volume 1554 of *Lecture Notes in Mathematics*. Springer, 1993.
4. Arjen K. Lenstra, Hendrik W. Lenstra Jr., Mark S. Manasse, and John M. Pollard. The number field sieve. In *22nd ACM STOC*, pages 564–572. ACM Press, May 1990.
5. M. O. Rabin. Digitalized signatures. In *Foundations of Secure Computing*, pages 155–168. Academic Press, 1978.
6. Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the Association for Computing Machinery*, 21(2):120–126, February 1978.

7. G. Santilli. *An Investigation on Integer Factorization Applied to Public Key Cryptography*. PhD thesis, University of Trento, 2019.
8. G. Santilli and D. Taufer. First-degree prime ideals of biquadratic fields dividing prescribed principal ideals. *Mathematics*, 8(9):1433, 2020.
9. R. D. Silverman. Optimal parameterization of SNFS. *Journal of Mathematical Cryptology*, 1(2):105–124, 2007.
10. I. Stewart and D. Tall. *Algebraic Number Theory and Fermat's Last Theorem*. CRC Press, 2015.
11. K. S. Williams. Integers of biquadratic fields. *Canadian Mathematical Bulletin*, 13(4):519–526, 1970.

Part III
Extended Abstracts: Applications

Privacy-preserving Information Sharing

Carlo Blundo

Department of Business Sciences - Management and Innovation Systems, University of Salerno, Italy
cblundo@unisa.it

Electronic information availability is increasingly essential to our communities' functioning, and, in numerous circumstances, parties without complete mutual trust need to share data. This sharing naturally raises commensurate privacy concerns for the disclosure, and long-term safety, of sensitive contents. The research community has begun to develop a few cryptographic techniques for controlled (privacy-preserving) information sharing to address related security and privacy issues.

Privacy-preserving computation has originated in the 1980s with the seminal works by Yao for the two-party case [13], and by Goldreich, Micali, and Wigderson for the multiparty case [8]. Informally, n parties, each holding a value (say, i-th party holds the value x_i), want to compute a function of all the inputs cooperatively (i.e., $f(x_1, x_2, \ldots, x_n)$) while keeping, at the same time, all values secret. To this aim, parties run cryptographic protocols and exchange messages that convey in a concealed way the secret values. At the end of the protocol, each party, from the exchanged messages and their secret value, can compute $f(x_1, x_2, \ldots, x_n)$.

Privacy-preserving Information Sharing was analyzed under different security models. The adversary can follow a corruption strategy that can be either static or adaptive. In a static scenario, the adversary corrupts and controls a fixed number of parties before the protocol starts. Corrupted parties remain corrupted throughout, and uncorrupted (i.e., honest) parties remain uncorrupted. In an adaptive scenario, the adversary chooses which parties to attack while the protocol is running. The security model also has to consider the allowed adversarial behavior corresponding to the actions the adversary, through corrupted parties, can take during the protocol's execution to infringe the protocol's privacy. A *semi-honest* adversary (a.k.a., honest but curious or passive) does not deviate from the protocol but will attempt to learn all possible information from legitimately received messages. A *malicious* adversary (a.k.a., active) can arbitrarily deviate from the protocol specification changing the input and output or aborting. Finally, the security model also considers the computational complexity of the adversary. If the adversary is computationally unbounded, then there are no limits to their computational power. On the other hand, a (probabilistic) polynomial-time adversary is allowed to run

De Cifris Koine – DE CIFRIS SEMINARS –https://doi.org/10.69091/koine/vol-2-A02

in polynomial time (i.e., the adversary may only perform a polynomial amount of operations including, in case they are probabilistic, at most a polynomial number of coin-flips). The model also has to deal with the number of protocol instances that are run simultaneously. In the *stand-alone* scenario, only a single protocol execution takes place; while, in the *universal composability* scenario, many secure (and insecure) protocols are concurrently run and arbitrarily composed.

In the following, to simplify the formal definition of Privacy-preserving Information Sharing, we will consider two-party protocols secure against a static semi-honest polynomial-time adversary run within the stand-alone scenario. Interested readers can find a thorough and comprehensive study of efficient protocols and techniques for secure two-party computation in [9]; while the reader can refer to [5] for a comprehensive discussion of multiparty computation secure against computationally unbounded adversaries.

In the two-party setting, parties are usually referred to as *Client* (i.e., $\mathcal{C}$) and *Server* (i.e., $\mathcal{S}$). The function they want to compute can be represented as

$$f : \{0,1\}^* \times \{0,1\}^* \rightarrow \{0,1\}^* \times \{0,1\}^* ,$$

where $f = (f_{\mathcal{C}}, f_{\mathcal{S}})$. The pair $(f_{\mathcal{C}}, f_{\mathcal{S}})$ represents the output computed by $\mathcal{C}$ and $\mathcal{S}$, in other word, the *Client*, holding the value x interacting with the *Server* holding the value y, would like to compute the value $f_{\mathcal{C}}(x, y)$ while the *Server* gets $f_{\mathcal{S}}(x, y)$. The two-party protocol for computing f is denoted by π, while, for a party $\mathcal{P} \in \{\mathcal{C}, \mathcal{S}\}$, $view^{\pi}_{\mathcal{P}}(x, y)$ represents $\mathcal{P}$'s view during the execution of π on (x, y).
The view contains the party's input (x or y depending on $\mathcal{P}$), all the messages received by $\mathcal{P}$, and all the random values used by $\mathcal{P}$ during the protocol execution. Party $\mathcal{P}$'s output at the end of the execution of π on (x, y) is denoted by $output^{\pi}_{\mathcal{P}}(x, y)$. Party $\mathcal{P}$'s output can be computed from $\mathcal{P}$'s view.

If we assume that the function f is deterministic, then, following [9], a two-party protocol π computes f in a privacy-preserving manner if it is *correct*, that is

$$output^{\pi}_{\mathcal{C}}(x, y) = f_{\mathcal{C}}(x, y) \text{ and } output^{\pi}_{\mathcal{S}}(x, y) = f_{\mathcal{S}}(x, y)$$

and *secure*, that is there exist two probabilistic polynomial time algorithms $S_{\mathcal{C}}$ and $S_{\mathcal{S}}$ (referred to as *simulators*) such that

$$\{S_{\mathcal{C}}(x, f_{\mathcal{C}}(x, y))\}_{x,y\in\{0,1\}^*} \stackrel{c}{\equiv} \{view^{\pi}_{\mathcal{C}}(x, y)\}_{x,y\in\{0,1\}^*}$$

$$\{S_{\mathcal{S}}(y, f_{\mathcal{S}}(x, y))\}_{x,y\in\{0,1\}^*} \stackrel{c}{\equiv} \{view^{\pi}_{\mathcal{S}}(x, y)\}_{x,y\in\{0,1\}^*}$$

The symbol $\stackrel{c}{\equiv}$ indicates that the probability distributions induced on the sets on its left and right sides are *computationally indistinguishable*, that is, there is no efficient algorithm that can tell them apart. If there exists a simulator that can generate a party's view in the execution, then whatever can be computed by a party

participating in the protocol can be computed based on their input and output only. Hence, the adversary does not learn anything from the protocol execution beyond what they can derive from the input of the party they control and the protocol's output.

As a specific application of two-party privacy-preserving information sharing, we can consider the privacy-preserving computation of set intersection [1] (PSI). In this case, both $\mathcal{C}$ and $\mathcal{S}$ inputs are a set of elements, say $\mathcal{C}$ holds the set C and $\mathcal{S}$ holds the set S and $f_{\mathcal{C}}(C, S) = (C \cap S, |S|)$, while, $f_{\mathcal{S}}(C, S) = |C|$. Other variants exist, for instance:

- private set intersection cardinality (PSI-CA) [1, 7] where the client learns the size of the intersection, but not the elements in $C \cap S$;
- [4, 6] where parties' sets are certified in the sense that a trusted party ensures the inputs are valid and binds them to each participant;
- *Authorized* PSI and PSI-CA *Policy-Enhanced* PSI [12] that allows parties to privately compute set intersection while enforcing rich privacy policies semantics previously not possible with traditional PSI and Authorized PSI.

We refer the reader to [11] for a comprehensive comparison among PSI major protocol paradigms.

Despite its simple formulation PSI and PSI-CA protocols have several practical uses. They can be employed as building blocks in many privacy-preserving compelling applications. To cite just a few: in proximity testing [10], $\mathcal{C}$ and $\mathcal{S}$ can test if they are close to each other without either party revealing any additional information about their location; in testing human genomes [2], one can securely implement *in silico* some operations, such as Paternity Tests, Personalized Medicine, and Genetic Compatibility Tests, that are currently performed via *in vitro* methods (experiments demonstrate that the proposed techniques are feasible and practical); in estimating the *Jaccard* similarity index of two sets [3], private set intersection cardinality is used to add a privacy layer to many applications, including document similarity, biometric authentication, genetic tests, multimedia file similarity.

References

1. R. Agrawal, A. V. Evfimievski, and R. Srikant. Information sharing across private databases. In *2003 ACM SIGMOD international conference on Management of data*, pages 86–97, 2003.
2. Pierre Baldi, Roberta Baronio, Emiliano De Cristofaro, Paolo Gasti, and Gene Tsudik. Countering GATTACA: efficient and secure testing of fully-sequenced human genomes. In Yan Chen, George Danezis, and Vitaly Shmatikov, editors, *ACM CCS 2011*, pages 691–702. ACM Press, October 2011.
3. C. Blundo, E. De Cristofaro, and P. Gasti. EsPRESSo: Efficient privacy-preserving evaluation of sample set similarity. *Journal of Computational Security*, 22:355–381, 2014.
4. J. Camenisch and G. M. Zaverucha. Private intersection of certified sets. In *Financial Cryptography and Data Security '09*, volume 5628 of *LNCS*, pages 108–127, 2009.
5. R. Cramer, I. Damgård, and J. B. Nielsen. *Secure Multiparty Computation and Secret Sharing*. Cambridge University Press, 2015.
6. E. De Cristofaro, P. Gasti, and G. Tsudik. Fast and private computation of cardinality of set intersection and union. In *11th International Conference CANS 2012*, 2012.
7. Michael J. Freedman, Kobbi Nissim, and Benny Pinkas. Efficient private matching and set intersection. In Christian Cachin and Jan Camenisch, editors, *EUROCRYPT 2004*, volume 3027 of *LNCS*, pages 1–19. Springer, Heidelberg, May 2004.
8. Oded Goldreich, Silvio Micali, and Avi Wigderson. How to play any mental game or A completeness theorem for protocols with honest majority. In Alfred Aho, editor, *19th ACM STOC*, pages 218–229. ACM Press, May 1987.
9. C. Hazay and Y. Lindell. *Efficient Secure Two-Party Protocols: Techniques and Constructions*. Springer, 2010.
10. Arvind Narayanan, Narendran Thiagarajan, Mugdha Lakhani, Michael Hamburg, and Dan Boneh. Location privacy via private proximity testing. In *NDSS 2011*. The Internet Society, February 2011.
11. B. Pinkas, T. Schneider, and M. Zohner. Scalable private set intersection based on ot extension. *ACM Transactions on Privacy and Security 2018*, 2018.
12. Emil Stefanov, Elaine Shi, and Dawn Song. Policy-enhanced private set intersection: Sharing information while enforcing privacy policies. In Marc Fischlin, Johannes Buchmann, and Mark Manulis, editors, *PKC 2012*, volume 7293 of *LNCS*, pages 413–430. Springer, Heidelberg, May 2012.
13. Andrew Chi-Chih Yao. How to generate and exchange secrets (extended abstract). In *27th FOCS*, pages 162–167. IEEE Computer Society Press, October 1986.

Non-interactive time-based proof-of-stake finality

Giovanni Antino and Iris Dimmi

AiliA SA, Switzerland
giovanni.antino@takamaka.io, iris.dimmi@takamaka.io

Introduction

Takamaka [1, 7, 9] is a last generation Blockchain platform designed to bring to every business the opportunity to easily create and experiment with Blockchain solutions. Blockchain technology holds the potential to streamline business processes, enable new business models, enhance security, increase revenues and reshape industries. Takamaka is created to eliminate the barriers that are preventing enterprises from easily unlocking the potentials of blockchain technologies.

Takamaka is a new proof-of-stake blockchain (Proof of Stake is an emerging alternative to the more traditional Proof of Work [5]) developed using Java [8]. Two key concepts to the Takamaka blockchain are "epochs" and "slots". Every epoch is comprised of 24000 slots and each slot is the time alloted to a given active miner, alternatively called *node*, to generate a block and forward it to the rest of the network, allowing it to propagate and become part of the chain. Ideally, the network would have between 200 and 400 active miners. Slots are assigned to the miners using a heuristic algorithm based on the amount of stake they possess.

Another important concept is that of *stake*. Our blockchain has two tokens: *red tokens*, to which we will refer to as **TKR**, and *green tokens*, to which we will refer to as **TKG**. While both tokens can be equally used to pay for operations performed on the blockchain, only **TKG** enable mining, through transactions of "STAKE". Holders of **TKG**, also referred to as "stakeholders", decide which potential miner to trust with the generation of blocks by voting with the amount of **TKG** they posses.

In this document we present a time-based PoS protocol with the following aims:

- fixed transaction costs
- decentralisation of the network
- a tolerance of network interaction related problems higher than 33%, Byzantine generals problem
- eliminating competition based on computational power that is typical of PoW solutions by setting minimal requirements for mining nodes, effectively limiting the transaction throughput of the blockchain

De Cifris Koine – DE CIFRIS SEMINARS –https://doi.org/10.69091/koine/vol-2-A01

Prerequisites

These are the prerequisites deemed necessary (at the time of the writing) to promote blockchain adoption in enterprise environments.

- Global time accuracy between 1 and 5 seconds
- More than 50% of the stake controlled by honest miners that follow the protocol
- Accessibility to the network by the mining nodes for at least 50% of the duration of an epoch (4 days over an 8-day period)

Paper structure

After this introduction, Section *Notation and workflow of the Takamaka blockchain* describes the blockchain details, including: the actors of the PoS protocol, the network configuration, the scanning time and its Consent algorithm, as well as the slot allocation and the block weights.
The document concludes by identifying the conditions needed to reach finality.

Notation and workflow of the Takamaka blockchain

Actors of proof

- *Mining Nodes.* The main role of these servers is the creation of the blockchain itself. Each one of them has the right to: vote on smart contracts, select, validate and include transactions in blocks, assign rewards, assemble blocks.
- *Replica Nodes.* These servers propagate transactions/blocks while maintaining a complete copy of the network, but they cannot create new blocks.
- *Holder.* Holders are the owners of the tokens needed to operate on the chain.
- *Stake Holder.* Address holders of the tokens needed to select a replication node to a mining node. Green Token (**TKG**) are the tokens used to control the blockchain, similar to Cardano's ADA [3], and allow:
  - Gas transactions payment.
  - Staking on network nodes, therefore determining the miners.

  TKGs constitute the most important tokens for the blockchain since they are necessary for its proper functioning. Half of them are created at launch and the other half is created over time via mining in each block.
  A double-coin system is becoming widespread in blockchain technology (see e.g. Quadrans, [2]). In Takamaka, the main purpose of Red Tokens (**TKR**) is to pay for the execution of transactions and their inclusion in the blockchain. If the caller's address has a balance of **TKG** and **TKR**, then the first tokens used for payment are **TKR**. Only if they are depleted then the remaining costs are covered with **TKG**. All the **TKR** are declared in the 0 block of the chain

using a reserved transaction used exclusively in this block. The possession of these tokens does not allow staking and does not confer any control over the blockchain. The introduction of this token as an alternative method of paying to operate on the blockchain solves a problem unique to PoS. In a PoS with a single token, to operate one needs to pay fees using the very token that gives them control over it, thus losing control over the network the more they use it, which would incentivise NOT operating in the network to retain its control.

Network configuration

The node addresses taking part in the construction of the blockchain are categorized as:

$\mathcal{M}$ Main: Addresses that do not belong to any physical server. They work similarly to the DNS system and are needed as reference for the stakeholders.
$\mathcal{O}$ Overflow: Addresses that belong to physical servers, the nodes that form the network and generate the blocks.

Stakeholders delegate their voting power to an $\mathcal{M}$ address using a specific type of transaction, called "STAKE". Stakes are valid until the stakeholder performs a special type of transaction called "STAKE UNDO". All stakes done by the stakeholder up to this point in time are "deleted" and can be reassigned. An $\mathcal{M}$ is a valid objective for the staking only when it has been assigned at least one node of $\mathcal{O}$s. An $\mathcal{M}$ can have an unlimited number of $\mathcal{O}$, though in effect the blockchain is designed to have at most 400 active mining nodes. An $\mathcal{O}$ can be assigned only to one $\mathcal{M}$ per epoch.

Scanning time and Consent algorithm

To better understand the concepts expressed in this paragraph, the default values used to configure the blockchain **Takamaka** will be illustrated.

- *Slot.* Is the smallest time unit, it corresponds to a time period of 30 seconds. For any slot the Consent algorithm chooses only one miner. This miner is the only one entitled to generate an assigned block on that time frame. If for any reason the block is not created or it is rejected by the network, then the slot will be marked as "skipped" and not replaced.
- *Epoch.* Slot aggregation, in the default configuration 24000 slots correspond to one epoch.

Slot allocation

Every epoch, the blockchain executes a heuristic algorithm to distribute the slots for the next epoch among the nodes proportionally to their assigned stake. This process

has different phases and is executed once $\frac{2}{3}$ of the current epoch is completed. All the following steps must be strictly deterministic[1]:

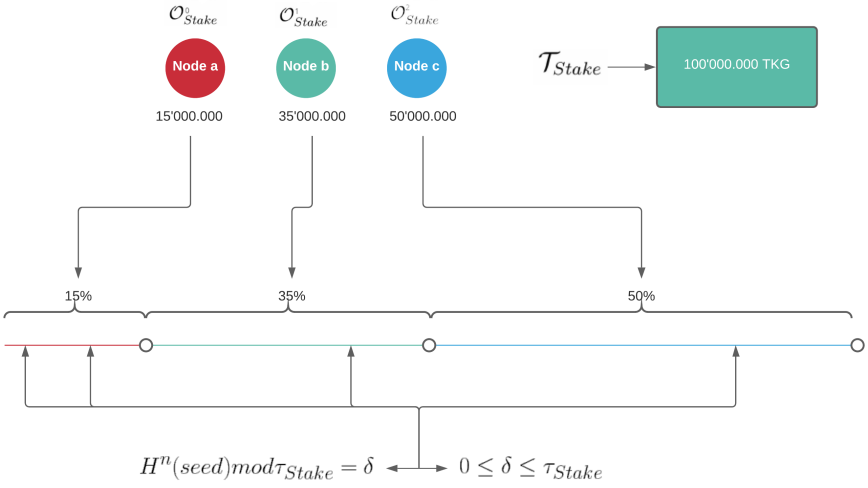


Fig. 1. Slot assignment Example - H: Hash function

- Identification of the block with the highest index number in the first third of an epoch: $\exists b_i \in Blocks \mid i = \max\left[0, \frac{\mathcal{T}_{SpE}}{3}\right)$.
- The *seed*[2] of the block, concatenated with the seeds of an arbitrary number of previous blocks, is passed to a hash function that creates a *ticket* (sequence of bits).
- A series of intervals, proportional to the stake, is generated and assigned sequentially to the $\mathcal{O}$ set of each $\mathcal{M}$. The resulting interval will be $[0, \text{totalStake} - 1]$.
- To determine the assignment of the i-th slot for the next epoch we proceed by first obtaining the result of the i-th iteration of the hash function. This value is transformed into its numeric counterpart and then we calculate its module base $\mathcal{T}_{SpE}$. The result from this last operation is a number in the interval $[0, \mathcal{T}_{SpE})$. The i-th slot will be assigned to the $\mathcal{O}$ owner of this interval. ***(Figure 1)***

[1] in this short introduction we do not detail the algorithms used for orders and decisions
[2] random sequence of bytes, unique to every block

Evaluation of the weight of a sequence of blocks

Takamaka adopts a strategy similar to that of Bitcoin [6], where the number of concatenated blocks is weighed on the complexity of the hash to determine which chain to extend. Takamaka nodes decide which sequence of blocks to extend by evaluating the weight of the preceding blocks. The weight of a block is determined by the total accepted stake divided by $\mathcal{T}_{SpE}$. For example, if $\mathcal{T}_{SpE} = 24000$ and the total accepted stake is 99000000, a single slot/block weight is 99000000/24000 = 4125. The nodes used for mining will always extend the heaviest chain available.

Finality

The proposed algorithm is somehow based on the sliding windows concept used in TCP/IP, where the block evaluation time window dynamically adapts to the network conditions.

Since slots are assigned by a heuristic algorithm, the resulting distribution does not match exactly the stake distribution and so it is necessary to correct the weight. Considering two overflow nodes $\mathcal{O}^1$ and $\mathcal{O}^2$ with the same amount of stake (e.g. 60000 **TKG**), we would expect the distribution function to assign them both the exact same number of slots. Unfortunately, when considering a small number of slots, like 60, heuristics can generate different results such as[3] $\mathcal{O}^1_{Slots} = 5000$ and $\mathcal{O}^2_{Slots} = 10000$. For this reason we introduced the concept of the "Expected theoretical slot weight" for a given i-th slot, assigned to the j-th overflow $\mathcal{W}^i_{O^j}$. This is a normalization of the slot weight. If we divide the stake assigned to the j-th overflow $\mathcal{O}^j_{Stake}$ by the total accepted stake for the epoch $\mathcal{T}_{Stake}$ and we multiply the resulting number by the total number of slots per epoch $\mathcal{T}_{SpE}$, then we get the theoretical number of slots that ought to be assigned to a node. Dividing this by $\mathcal{O}^j_{Slots}$, the actual number of slots assigned, we get the ratio by which the node's blocks will be multiplied to determine their weight. This is defined as $\mathcal{W}^i_{O^j}$ (e.g. 1).

$$\mathcal{W}^i_{O^j} = \frac{\mathcal{T}_{SpE}\frac{\mathcal{O}^j_{Stake}}{\mathcal{T}_{Stake}}}{\mathcal{O}^j_{Slots}} \tag{1}$$

Looking at the previous example and assuming a total accepted stake value, $\mathcal{T}_{Stake} = 18000$, and a number of slots per epoch, $\mathcal{T}_{SpE} = 24000$, we would get a weight multiplier of 1.6 for all the slots assigned to $\mathcal{O}^1$ in the corresponding epoch (e.g. 2).

$$\mathcal{W}^i_{O^1} = 1.6 = \frac{24000\frac{6000}{18000}}{5000} \tag{2}$$

[3] $\mathcal{O}^j_{Slots}$ is the number of assigned slots for the j-th overflow by the heuristic

Looking at the second node in our example $\mathcal{O}^2$, we get a slot weight multiplier of 0.8. (ex 3)

$$\mathcal{W}^i_{O^2} = 0.8 = \frac{24000\frac{6000}{18000}}{10000} \tag{3}$$

The "i" index identifies the epoch we are referring to. For the sake of simplification we will use an absolute index. When identifying the time after a certain event took place, we would usually say it happened a day and 12 hours after said event or 36 hours $(1 * 24 + 12)$ after. Assuming that the event in question is the creation of the Zero Block (epoch 0, slot 0), we can say that the epoch 10, slot 315 matches $i = 10 * 24000 + 315 = 24031$. Therefore, i is the absolute slot since the beginning of the blockchain. Given i, the reverse calculation is trivial, epoch $= \lfloor i/24000 \rfloor = 10$ and epoch $= i \bmod 24000 = 315$. We define the minimal temporal window where, at least theoretically, all nodes have the possibility to "vote" and create a block. With a maximum number of nodes set at 400, this window must not be smaller than 400 slots. Let us assume the "absolute" slot number to be 48401 (epoch 2, slot 401) and that the previous 48000 blocks were all correctly created and added to the chain. Starting from the 48000-th block, all miners become unavailable, with the exception of $\mathcal{O}^1$ and $\mathcal{O}^2$ and these nodes have the same configuration as in the previous examples 2 and 3. If we take the slot $n = 48000$ as a reference, n is final if the weight of the blocks in $[h = 48001, k = 48401]$ is greater than 50%+1 of the expected weight assuming all blocks in the interval are created and accepted. We define $\mathcal{WS}^k_h$ as the maximum weight for $[h, k]$,

$$\mathcal{WS}^k_h = \frac{\mathcal{T}_{Stake}}{k - h} \tag{4}$$

In our case $\mathcal{WS}^{48401}_{48001}$ with $\mathcal{T}_{Stake} = 18000$ is:

$$300 = \mathcal{WS}^{48401}_{48001} = \frac{18000}{24000} \cdot (48401 - 48001) \tag{5}$$

We can assume that in $[h, k]$ both $\mathcal{O}^1$ and $\mathcal{O}^2$ generated 134 blocks each, while no other nodes generated any block. We can define $\mathcal{WB}^k_h$ as the sum of the weight of the blocks in $[h, k]$. A block's weight equals that of its corresponding slot if it becomes part of the chain, otherwise it is zero. When the interval under consideration overlaps two epochs, the weight of the blocks generated by the same miner may vary. In that case the formula used to calculate the weight is given by the following formula (e.g. 6) where $\mathcal{N}(\mathcal{O})$ indicates the highest indexed overlow node. With 4 nodes $\mathcal{N}(\mathcal{O}) = 3$, since the indexes would be {0,1,2,3}.

$$\mathcal{WB}^k_h = \sum_k^h \left(\sum_{j=0}^{\mathcal{N}(\mathcal{O})} \mathcal{W}^i_{O^j} \right) \tag{6}$$

Thus, $\mathcal{WB}_{48001}^{48401} = 321.6$. The weight limit for the block n, index of the last block before the interval under consideration, to be final is defined as $\mathrm{W51}_h^k$. Given an interval $[h, k]$, $\mathrm{W51}_h^k$ indicates $50\% + 1$ of the total weight of the interval (e.g. 7).

$$W51_h^k = \frac{\mathcal{WS}_h^k}{(k-h)} \left(\frac{k-h}{2} + 1 \right) \tag{7}$$

For the considered example: $W51_h^k = 150.75 = 300/400 * (200 + 1)$. At this point we compare the resulting two weights. If the weight obtained by summing the generated slots is greater than $W51_h^k$ for the interval $[h, k]$, then slot n (the slot preceding the interval) can be considered final. (e.g. 8):

$$\mathcal{WB}_h^k \geqslant W51_h^k \tag{8}$$

In this situation, the block n is accepted as final.
Concluding with the example:

$$\mathcal{WB}_{48001}^{48401} \geqslant W51_h^k$$

and

$$321.6 \geqslant 150.75$$

As a result of its advanced finality protocol, Takamaka is very fast and can compete fairly with its current competitors (see e.g. [4]).

References

1. G. Antino and I. Dimmi. Non-interactive time-based proof of stake finality. Technical report, AiliA SA, 2020.
2. M. Battagliola, A. Flamini, R. Longo, A. Meneghetti, and M. Sala. Quadrans Blockchain -Yellow Paper v0.2. Technical report, Quadrans Foundation, 2021.
3. Cardano Foundation. Cardano. `https://cardano.org/`.
4. A. Meneghetti, T. Parise, M. Sala, and D. Taufer. A survey on efficient parallelization of blockchain-based smart contracts. *Annals of Emerging Technologies in Computing*, 3:9–16, 2019.
5. A. Meneghetti, M. Sala, and D. Taufer. A Survey on PoW-based Consensus. *Annals of Emerging Technologies in Computing*, 4:8–18, 2020.
6. S. Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008.
7. AiliA SA. Takamaka. `https://www.takamaka.io/`.
8. F. Spoto. Enforcing determinism of Java smart contracts. In *FC 2020 International Workshops*, pages 568–583. Springer, 2020.
9. N. Spoto, G. Antino, F. Pasetto, F. Tagliaferro, M. Carlini, A. Belvedere, F. Tai, and D. Cordioli. Takamaka White Paper v3.1. Technical report, AiliA SA, 2019.

The Links between Machine Learning and Blockchain

Andrea Gangemi

Department of Mathematical Sciences G. L. Lagrange, Politecnico of Torino, Italy
andrea.gangemi@polito.it

Starting from the end of the last century, Machine Learning (ML) and, more recently, blockchains are revolutionizing the world of technology. Machine Learning is a subset of Artificial Intelligence, and it refers to the study of computer algorithms which improve automatically, learning from the data a user provides. ML algorithms try to build statistical models based on known data, which are then used to predict a specific variable on future, unknown data [6].
The first blockchain was described by Satoshi Nakamoto in 2008 [9], in a famous paper that introduced Bitcoin, a decentralized digital currency. A lot of blockchains were built after that year and, in particular, Vitalik Buterin in 2013 first described the Ethereum blockchain [5]. The main applications of Ethereum are the so-called *smart contracts*, which are computer programs that automatically execute an action according to the terms of the contract.

At a high level, a blockchain can be considered as a distributed ledger, which first solved the problem of double-spending without a central authority. Every block contains a list of transactions, which represent the exchange of value between users. In the Bitcoin blockchain, a transaction is formed by a list of input/output addresses, which refer to previous output transactions, and a list of output addresses. Since there is not a user's balance, one of the output addresses is usually a change address and it belongs to the same user who started the transaction. A block is appended to the chain thanks to the work of miners, who invest their computational power to try to solve a cryptographic puzzle called Proof-of-Work (PoW).

The privacy of every user is guaranteed by the use of blockchain addresses, which provide pseudoanonimity [3]. This characteristic attracted people interested in illicit activities, since payments done through a blockchain cannot be immediately led back to the real identity of a user. To defend against this risk, researchers started applying ML algorithms on blockchain addresses, mainly Bitcoin, to classify between honest and dishonest users. In a nushell, this strategy exploits the information which can be found off-chain, which we now present in deatils.

De Cifris Koine – DE CIFRIS SEMINARS –https://doi.org/10.69091/koine/vol-2-A03

The strategy

Let us start with two examples. An exchange site must publish its address online to let people know about them, or sometimes an address can be advertised as a mistake on a forum or social network. Starting from a list of public addresses, and exploiting the structure of a transaction, it is possible to classify every user with a tag, representing what they use the blockchain for, such as: exchange, gambling, ransomware and so on.

Most works on this research area [11] imposed some strict hyphoteses, like having every user labeled with a single, specific tag and, on the other hand, having the list of input addresses all belonging to the same user. Based on this metric, clustering algorithms can be performed to obtain different groups containing addresses (which all belong to the same user). Starting from an analysis of the transaction outputs, empowered by the off-chain information, some of these groups can be labeled.

To classify the rest of the clusters, researchers have used ML algorithms: the clusters already labeled are divided into training set and test set, while several algorithms has been tried on these data and ranked thanks to the F1-score metric [6]. The best performing algorithm is probably the Gradient Boosting Classifier (GBC), which claims to identify correctly around 80% of the blockchain addresses [11].

Prediction of cryptocurrency prices

Machine Learning can also be used to predict future prices of cryptocurrencies, starting from past prices. To fight against their high volatility, studies consider the daily price as an average of the daily price fluctuations. Like in address clustering, past prices are divided into training and test set, and then different ML algorithms are ranked with the RMSE metric [6]. This time, two different algorithms showed interesting results: GBC for short-term predictions [2] and neural networks for long-term predictions [8].

A variant of these methods exploits the properties of Bitcoin transactions [1]. This novel approach counts day by day the number of transactions with i inputs and j outputs, and saves this number in a matrix. Then, it uses clustering techniques to group together these transactions with the cosine similarity metric, which measures how much two texts or, in this case, transactions, are close to each other. Finally, price prediction is conducted on a single cluster or on a small subset of them.

The study claims that this analysis (based on chosen subsets) gives better results for long-term predictions compared to the same analysis performed on the whole transaction set.

Other interactions between ML and blockchain

On the other side of the coin, a blockchain is also a tool which can help the growth of ML techniques. In fact, one of the main problems of Machine Learning is the centralization of datasets in some authorized servers. These servers act as trading centers, and they can be contacted by interested companies. Of course, a central authority represents a single point of failure, something that everyone would like to avoid. Researchers had the idea to exploit the blockchain properties to save datasets on their ledgers, in order to provide data all around the world in a decentralized fashion [10]. The described work uses heavily the smart contracts and for this reason it may be built on top of the Ethereum blockchain.

However, blockchains alone are not enough because a dataset is too large to be stored reliably on a block. The proposed solution utilizes also the InterPlanetary File System (IPFS), a protocol and peer-to-peer network for storing and sharing data in a distributed file system [7]. The dataset is saved into the IPFS server, while the hash of the dataset is recorded on the blockchain. The exchange of a dataset between the seller and an interested buyer occurs through a clever use of a smart contract, which is written by a mediator according to the requests of the two parties. This method has some flaws but in the future it could radically change how data are exchanged between different companies. Finally, Machine Learning may also be utilized to build a blockchain consensus algorithm. In fact, ML models are usually hard to solve but easy to verify, which is the key feature for a good consensus algorithm. The theoretical idea is known as Proof-of-Learning and was first described in [4]: it takes inspiration from Kaggle competitions, which allow the development of new performing algorithms in small amounts of time. Obviously, to utilize this protocol we have to build a new blockchain, suited for this task. On this blockchain, there would be three kinds of actors: suppliers, who propose new Machine Learning problems to the network, trainers, which develop their models, and validators, which test the models and propose the new blocks. Again, to avoid the space problem, suppliers publish their datasets on the IPFS server, and insert its digest on the blockchain.

References

1. C. G. Akcora, A. K. Dey, Y. R. Gel, and M. Kantarcioglu. Forecasting Bitcoin price with graph chainlets. In *Advances in Knowledge Discovery and Data Mining, PAKDD 2018*, LNCS, 2018.
2. L. Alessandretti, A. ElBahrawy, L. M. Aiello, and A. Baronchelli. Anticipating cryptocurrency prices using machine learning. *Complexity*, page 1–16, 2018.
3. A. Antonopoulos. *Mastering Bitcoin: Unlocking Digital Cryptocurrencies.* O'Reilly, 2017.

4. F. Bravo-Marquez, S. Reeves, and M. Ugarte. Proof-of-learning: A blockchain consensus mechanism based on machine learning competitions. In *2019 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPCON)*, 2019.
5. V. Buterin. Ethereum: A next-generation smart contract and decentralized application platform, 2013.
6. T. Hastie, R. Tibshirani, G. James, and D. Witten. *An Introduction to Statistical Learning with Applications in R.* Springer, 2013.
7. Protocol Labs. Ipfs. `https://ipfs.io/`.
8. Salim Lahmiri and Stelios Bekiros. Cryptocurrency forecasting with deep learning chaotic neural networks. *Chaos, Solitons & Fractals*, 118:35–40, 2019.
9. S. Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008.
10. W. Xiong and L. Xiong. Smart contract based data trading mode using blockchain and machine learning. *IEEE Access*, 7:102331–102344, 2019.
11. Hao Hua Sun Yin, Klaus Langenheldt, Mikkel Harlev, Raghava Rao Mukkamala, and Ravi Vatrapu. Regulating cryptocurrencies: A supervised machine learning approach to de-anonymizing the Bitcoin blockchain. *Journal of Management Information Systems*, 36(1):37–73, 2019.

Post-quantum cryptography and the automotive industry

Efstathia Katsigianni

IBM Deutschland GmbH, Germany
efstathia.katsigianni@escrypt.com

Asymmetric cryptography in automotive

In the last twenty years, vehicles have become increasingly complex and have obtained increasing connectivity capabilities. This has made a variety of applications possible, such as the access to the manufacturer's cloud services, fleet management, car-to-car and car-to-infrastructure communication, remote software update and remote diagnostics. All these interfaces have however in turn created a variety of new attack vectors for the connected vehicles: attackers can for example eavesdrop the communication or inject malware to a vehicle being in the field.

Cyber attacks of this kind can have very serious ramifications, such as the endangerment of human lives, compromised privacy and theft of intellectual property. The widespread use of the connectivity capabilities of modern vehicles therefore means that the establishment of security measures, with regards to the in-vehicle, onboard and online network, is of uttermost importance. Cryptography is as expected the cornerstone of every security solution, as by using well-established cryptographic schemes, one can achieve confidentiality (through encryption), ensure the integrity of information (using hash functions and message digests), as well as verify the authenticity of messages (by using digital signatures).

Digital Signatures using asymmetric cryptography are nowadays indeed an inherent part of applications in the embedded systems and the automotive industry. Applications such as secure over-the-air updates (SOTA) and Car-to-X communication are only some examples, which highlight the importance of establishing a secure communication channel and ensuring the messages are issued by a trusted sender (for example the unit-manufacturer).

Digital certificates are used extensively as a way of binding the identities of the communicating entities to the corresponding public keys. These are issued by corresponding authorities, which are parts of a common Public Key Infrastructure (PKI). Car manufacturers (Original Equipment Manufacturers - OEMs) nowadays operate individual PKIs, with different entities responsible for multiple use cases (e.g., diagnostics, testing and production), which might depend on a common or

De Cifris Koine – DE CIFRIS SEMINARS –https://doi.org/10.69091/koine/vol-2-A04

distinct Root certification authorities (Root CA(s)). Security in these cases nowadays depends upon the security of well-known algorithms, such as RSA, ECDSA and ECDH for signing and key generation. These in turn depend on hard-to-solve mathematical problems, namely prime factorization and solving discrete logarithms on elliptic curves.

The threat of quantum computers

Developments in the field of quantum computing have however shown that there is an increasing need for revising the algorithms used for digital signatures and key establishment, while symmetric block algorithms are still considered secure. Grover's algorithm (1996) [1] improves brute-force algorithms that check every possible key, providing a quadratic speed-up. This means that for example a brute-force attack on AES-128 with a cost of at most 2^{128} AES-operations on a classical computing system can be finished with about 2^{64} AES-operations on a quantum computer.

Shor's algorithm [5], [4] provides a way however to solve integer factorization and discrete logarithms in polynomial time with a quantum computer, while these problems have apparently exponential complexity with classical computers. Although there is still a lot of progress to be made in the development of suitable quantum computers, the time needed for finding suitable solutions and implementing them in fields like the automotive industry makes the transition to a quantum-world a very real and urgent matter. As per the often cited theorem of M. Mosca [3]: if the time needed for migrating to new solutions added to the time one product needs to be secure is greater than the time needed to compromise its security, then action has to be taken.

The cryptographic community has already considered this and there are already various standardization activities taking place, most notably the NIST Post Quantum Algorithm competition [2]. The quantum-secure algorithms submitted to the NIST competition fall into five categories:

- Hash-based signature schemes: relying on the security of the chosen hash function.
- Isogeny based: relying on the difficult mathematical problem of finding isogenies between special elliptic curves (SIDH/SIKE).
- Lattice based: relying on the shortest vector problem and learning with errors problem.
- Code based: the security of such systems is based on the hardness of inverting a random linear code.
- Multivariate-equations based: the security of these systems is based on the fact that solving multivariate quadratic systems of equations over finite fields is NP-hard.

As part of the NIST competition different algorithms will be chosen for signature generation, key establishment and public key encryption in the next couple of years. As recently as July 2020, this competition entered its third round, which includes seven finalist-algorithms, which will be considered for standardization in the next two years, and eight alternate algorithms, which may be standardized in the non-immediate future.

Challenges for the automotive industry

In the process of identifying suitable algorithms for all the relevant use cases in the automotive industry, there are different problems that have to be taken into consideration. One of them is related to resource constraints: many of the proposed schemes produce very large signatures or require very large key pairs and would therefore be not suited to be used in ECUs (electronic control units) with a limited amount of secure memory and computation resources. The security level offered by each algorithm is one more important factor that needs to be analyzed with respect to the needs of every individual use case.

A variety of issues need however to be considered even after suitable quantum-secure algorithms are chosen. A smooth transition from current systems to post-quantum enabled ones is necessary, not only for the ECUs of the future, but also the ones already in the field. For critical use cases, like online firmware updates, many ECUs in the field should be able to securely communicate with the backend systems of the OEM without big interruptions.

On the other hand, flexibility on the choice of algorithms (cryptographic agility) is necessary, as the security levels offered by different algorithms vary and as advances in computing and cryptographic research may soon make some standardized post-quantum algorithms obsolete. The integrity of algorithm selection needs to be in turn ensured (protection against downgrade attacks), while in-field updates of the used algorithms and parameters, e.g. the key lengths of the symmetric keys used, should also be possible.

At the same time, performance restrictions must be addressed by careful resource planning and by taking advantage of or extending hardware acceleration solutions. The communication protocols in use should be adjusted to handle post-quantum signatures and key exchange, while the available key and certificate management solutions must take the different requirements of the post-quantum algorithms into consideration.

Last but not least, new post-quantum enabled PKIs have to be put in place, which are compatible with the available PKIs and be able to handle classical certificates for older ECUs, while they are still valid. To this end those PKIs can be designed according to the parallel or the hybrid approach. The OEMs would then need to operate one classical and one post-quantum PKI in parallel, or use hybrid certificates, signed by two different public keys, one classical and one post-quantum.

Hybrid certificates have been widely studied in the last years, since they allow compatibility with targets that do not yet support post-quantum schemes.

Conclusion

In conclusion, the technological transformation originating from the eve of quantum computers poses a variety of challenges and risks for the automotive industry. Tackling these challenges requires a lot of preparation and careful ECU-design decisions by the OEMs. How the transition of ECUs in the field towards the post-quantum world can be achieved is a question that will soon need to be answered.

As part of the research project FLOQI, which is funded by the German Federal Ministry of Education and Research, all these challenges for the automotive industry are being considered and the project partners engage in the development of a quantum computer-resistant PKI. The goals of the FLOQI project include the specification of a PKI supporting both classical and quantum-computer-resistant algorithms and the choice of signature and key agreement algorithms suitable for use-cases in the automotive industry, the financial sector, e-governance and, obviously, the entire Industry 4.0.

References

1. L. K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-eighth Annual ACM Symposium on Theory of Computing*, pages 212–219, New York, NY, USA, 1996. ACM.
2. D. Moody, G. Alagic, D. C. Apon, D. A. Cooper, Q. H. Dang, J. M. Kelsey, Y. Liu, C. A. Miller, R. C. Peralta, R. A. Perlner, A. Y. Robinson, D. C. Smith-Tone, and J. Alperin-Sheriff. Status report on the second round of the NIST post-quantum cryptography standardization process. Technical report, NIST, 2020.
3. Michele Mosca. Cybersecurity in an era with quantum computers: Will we be ready? Cryptology ePrint Archive, Report 2015/1075, 2015. `https://eprint.iacr.org/2015/1075`.
4. P. W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal of Computation*, 26(5):1484–1509, 1997.
5. Peter W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *35th FOCS*, pages 124–134. IEEE Computer Society Press, November 1994.

Digital Identity - modern tools and perspectives

Alessandro Tomasi

Fondazione Bruno Kessler, Trento, Italy
altomasi@fbk.eu

Digital tools for the authentication of persons, processes, and legal entities are part of our critical infrastructure, enabling secure access, in presence or remotely, to essential public and private services.

NIST Digital Identity Guidelines [3] provide a robust baseline on the subject, establishing the notion of digital identity as a set of context-specific attributes; separating lifecycle stages, from proofing and enrollment to authentication and the management of authenticators; and introducing the notions of identity providers and relying parties in a federated context, where an identity asserted by one entity is relied upon by another, and the provider discloses information only after the establishment of a relationship of trust.

Two of the most important trends in digital identity are (a) the ubiquitous availability of strong cryptography for individuals, backed by hardware and used in secure protocols, and (b) a shift towards more open systems - interoperability, decentralization, and distribution. Federation is popular in an enterprise context, where a single trusted authority is relied upon to manage identities. It is also perceived as useful to individuals by reducing the number of passwords they need to store. However, passwords have been repeatedly shown to be a vulnerability in practice, and dramatic improvements in passwordless authentication flows have been made. Additionally, federated protocols have drawbacks when expressing long-term credentials about subjects, as they are generally managed by private enterprise, while losing access to a private enterprise account for any reason would make those credentials unverifiable.

Personal keys: ubiquity and interoperability. The Fast IDentity Online (FIDO) Alliance is developing an open ecosystem for standards-based and interoperable strong authentication solutions, designed to minimize password use and phishing threats. This requires both the provision of secure hardware authenticators capable of generating and storing asymmetric key pairs, such as YubiKeys[1], and the development of secure authentication protocols with which the hardware security modules can operate. Key enabling recent developments are the Client to Authenticator Protocol (CTAP[2]), which specifies how hardware authenticators communicate with

De Cifris Koine – DE CIFRIS SEMINARS –https://doi.org/10.69091/koine/vol-2-A05

[1] https://www.yubico.com/

[2] https://fidoalliance.org/specifications/

the client, together with the W3C Web Authentication specification (WebAuthn[3]), which defines a standard web API for online services.

Hardware security modules are increasingly available in, and compatible with, mobile phones. The ubiquitous availability of phones as HSM readers and biometric sensors, coupled with highly user-friendly clients, has actually made cryptography in particular, and multi-factor authentication in general, extremely widespread. Moreover, as hardware readers they are now capable of bridging the gap with more traditional devices such as smart cards. For example, the Italian Electronic Identity Card (CIE) 3.0 [5] is NFC enabled and supports a PIN-based authentication protocol. This can be leveraged to derive other identities with mobile-based authentication flows, such as the PosteID[4] SPID[5] scheme.

Identity as a public service and infrastructure: eIDAS. The eIDAS regulation [4] and its implementing technical framework [2] are explicitly designed to enable the cross-border interoperability and legal validity of individual national electronic identity (eID) schemes, allowing EU citizens to access public services in other member states, and enabling an ecosystem of private services to be built upon this public infrastructure.

A list of notified eIDAS schemes, including CIE 3.0 and SPID, is maintained by the eID User Community[6]. Concretely, eIDAS allows Relying Parties (RP) to receive assertions on a core attribute set [1] from the eIDAS attribute profile of eID bearers.

Decentralization and public registries. There exists an infrastructure allowing legal entities to be registered, resolved, authenticated, and have legally binding statements verified, instantly and remotely. This infrastructure relies on a combination of public registries and commercial trust service providers. Decentralized identity proposals lay foundations for a similar infrastructure for natural persons, independently of identities provided by private enterprises and tied, e.g. to a company account.

Website addresses are resolved through the Domain Name System (DNS), which translates human-readable URLs into machine-readable IPs. DNS is operated by 12 organizations[7] providing a public service for name resolution.

A resolved domain can be authenticated through X.509 certificates. These can be issued for Domain Validation (DV) to assert control over a domain, or as Extended Validation to associate a legal entity to the domain. Browsers and operating systems

[3] https://www.w3.org/TR/webauthn-2/

[4] `https://posteid.poste.it/`

[5] `https://www.spid.gov.it/`

[6] `https://ec.europa.eu/cefdigital/wiki/display/EIDCOMMUNITY/Overview+of+pre-notified+and+notified+eID+schemes+under+eIDAS`

[7] `https://root-servers.org/`

ship with trusted root certificates. EV checks rely on official online registries of company incorporation or registration. PSD2-compliant QCs may only be issued by Qualified Trust Service Providers, a registry of which is maintained by Open Banking Europe. For private persons, the most common equivalent is a phone number or email address, each of which has several available authentication protocols, from SMS and push messages to single-sign-on and one-time links. Holders of an eID do carry an official X.509 certificate asserting their identity, issued by their national identity provider, but there is no resolution service for public keys or identifiers associated to the public identity of national citizens, partly due to privacy concerns. Decentralized Identifiers have been proposed as a public resolution and registry service for personal identities. This would offer individuals a portable identity that is not tied to a single commercial service provider, and might enable services providing online signature and verification of identities and contracts. Decentralization extends to the entities that can issue signed assertions about DIDs in the Verifiable Credentials recommendation, which does not require federation. The eIDAS Bridge project developed for the European Blockchain Services Infrastructure enables VCs about DID subjects signed with eIDAS certificates to be verified.

References

1. eIDAS eID Technical Subgroup. eIDAS technical specifications - eIDAS SAML Attribute Profile. Technical report, European Council, July 2014. Accessed: 2024-07-22.
2. eIDAS eID Technical Subgroup. eIDAS technical specifications - eIDAS interoperability architecture v1.2. Technical report, European Council, September 2019. Accessed: 2024-07-22.
3. P. A. Grassi, M. E. Garcia, and J. L. Fenton. Digital Identity Guidelines. Technical report, NIST, June 2017. Accessed: 2024-06-28.
4. Official Journal of the European Union. Regulation 910/2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC. Technical report, European Parliament and European Council, 2014. Accessed: 2024-06-28.
5. `https://www.cartaidentita.interno.gov.it/`. Carta d'Identità Elettronica CIE 3.0 - Specifiche Chip. Technical report, Ministero dell'Interno, Istituto Poligrafo e Zecca dello Stato Italiano, November 2015. Accessed: 2024-06-28.

KDFs: an essential (and usually transparent) component of real-world applications

Andrea Visconti

Department of Computer Science, University of Milano, Italy
andrea.visconti@unimi.it

Nowadays, most of real-world applications use passwords or passphrases to protect users' data or to provide privileged access to specific resources. Unfortunately, user-chosen passwords are often short, lack of enough entropy, and must be securely stored in a database. If attackers are able to collect this data they might have access to detailed information on million of users. Looking at information on the internet — e.g. online articles, press reports, government news releases, and so on — it is not difficult to find examples of company data breaches (see Table 1). These data breaches usually are due to poor security, incorrect configuration of information systems, cyber attacks, insiders stealing or leaking data from their employers, and so on. A non-regulatory agency, such as the National Institute of Standards

Year	Organization	Records stolen	Source
2020	Instagram	200M	[17]
2020	Marriott International Inc.	5,2M	[1]
2020	Tetrad	120M	[14]
2019	Adobe Inc.	7,5M	[2]
2019	Microsoft	250M	[3]
2019	Bharti Airtel Limited	320M	[10]
2019	Capital One Financial Corp.	106M	[9]
2019	EasyJet	9M	[15]
2019	Just Dial Limited	100M	[6]
...	...	...	...

Table 1. A (non-exhaustive) list of data breaches

and Technology (NIST), develops and provides information security standards and guidelines for how public and private sector organizations in the United States assess and improve their ability to prevent, detect, and respond to cyber attacks. An example of this guidelines is NIST SP 800-63-B [7], in which NIST's researchers describe general requirements for authenticator types: memorized secret authenticators (password, PIN), look-up secret authenticators (physical/electronic records that store a set of secrets), single-factor/multi-factor OTP authenticators, ...

De Cifris Koine – DE CIFRIS SEMINARS –https://doi.org/10.69091/koine/vol-2-A06

Particular attention is paid to memorized secret authenticators, for which they suggest to implement controls to protect secrets against online guessing attacks. These controls suggest to avoid context-specific passwords, passwords derived from the previous ones, dictionary words, repetitive/sequential characters, and so on. In addition, NIST's researchers suggest to store memorized secret authenticators resistant to offline attacks, using a key derivation function (KDF) which inputs an hash function and a salt, among others. Doing so, they are suggesting that (a) symmetric encryption algorithms have to be avoided because users' secrets cannot be decrypted neither by users nor any system administrators; (b) identical, or similar, secrets must have different and unrelated digests stored into our database; (c) digests stored have not to provide information about the lengths of users' secrets.

The growing use of GPUs, FPGAs, and ASICs for brute-forcing users' secrets has made the selection of cryptographic algorithms a critical point, indeed, a good algorithm have to enforce a certain amount of computational cost on these devices. For this reason, we use key derivation functions to store secrets on our machines. The aim of key derivation functions is to slow attackers down as much as possible, introducing instructions that do nothing apart wasting CPU time and memory space to compute intermediate data. Even in very recent postquantum applications KDFs are necessary to guarantee security proofs ([11, 13]).

One of the most widely used key derivation functions is `PBKDF2`[12]. In order to secure passwords, `PBKDF2` has been involved in many real-world implementations and, among many, we can mention Android full disk encryption, LastPass, WPA/WPA2, GRUB2, LUKS, 1Password, EncFS, FileVault Mac OS X, Winrar, ... Notice that `PBKDF2` is not the only one. In the literature, we can also find scrypt and a number of participants to the password hashing competition: Argon2 ([16], the winner), Catena, Lyra2, yescrypt and Makwa (recommended algorithms).

In order to explain the behavior of a KDF, in this extended abstract, we will focus on `PBKDF2`. This KDF can derive secrets of arbitrary length, generating as many blocks T_i as we need (see equation 2). Each block T_i is computed iterating a pseudo-random function (PRF), for example HMAC. Therefore, $hLen$ — i.e. the length of each block T_i — is bounded by HMAC that, in turn, depends on the hash function adopted. `PBKDF2` inputs a random salt s, a secret p, a key length $dkLen$. Iterating the HMAC function c times, it outputs a derived key DK (see eq. 1).

$$\mathrm{DK} = \texttt{PBKDF2}(p, s, c, dkLen) \tag{1}$$

where DK is the concatenation of $\lceil dkLen/hLen \rceil$-blocks:

$$\mathrm{DK} = T_1 || T_2 || \ldots || T_{\lceil dkLen/hLen \rceil} \tag{2}$$

In order to slow down the attackers, we have to set the iteration count as large as possible, and in real-world applications this counter can vary considerably — i.e., 2,000 for WPA/WPA2 [8] and 2,500,000 for LUKS [4, 5].

References

1. A. Bera and S. Ganguli. Marriott says 5.2 million guests exposed in new data breach. *Reuters*, 2020. Accessed: 2024-06-28.
2. P. Bischoff. 7 million Adobe Creative Cloud accounts exposed to the public. *Comparitech*, 2019. Accessed: 2024-06-28.
3. I. Bonifacic. Microsoft accidentally exposed 250 million customer service records. *Engadget*, 2019. Accessed: 2024-06-28.
4. Simone Bossi and Andrea Visconti. What users should know about full disk encryption based on LUKS. In Michael Reiter and David Naccache, editors, *CANS 15*, LNCS, pages 225–237. Springer, Heidelberg, December 2015.
5. M. Broz, V. Matyas, O. Mosnáček, and A. Visconti. Examining PBKDF2 security margin — case study of LUKS. *Journal of Information Security and Applications*, 46:296–306, 2019.
6. ET Bureau. Data breach at JustDial leaks 100 million user details. *The Economic Times, Business News, Tech*, 2019. Accessed: 2024-06-28.
7. P. A. Grassi et al. NIST SP 800-63B. Technical report, NIST, 2016.
8. IEEE. IEEE Std 802.11 i-2004 Part 11: wireless LAN medium access control (MAC) and physical layer (PHY) specifications. Technical report, IEEE Computer Society, 2004.
9. R. McLaren. A hacker gained access to 100 million Capital One credit card applications and accounts. *CNN Business*, 2019. Accessed: 2024-06-28.
10. S. Nazmi. Indian Airtel: Bug meant users' personal data was not secure. *BBC News*, 2019. Accessed: 2024-06-28.
11. M. Qi. An efficient post-quantum kem from csidh. *Journal of Mathematical Cryptology*, 16(1):103–113, 2022.
12. RSA Labs. *PKCS #5 V2.1: Password Based Cryptography Standard*, 2012.
13. O. Taraskin, V. Soukharev, D. Jao, and J. T. LeGrow. Towards isogeny-based password-authenticated key establishment. *Journal of Mathematical Cryptology*, 15(1):18–30, 2021.
14. UpGuard Team. Household Names: How Tetrad Exposed Data on 120 Million Consumers. *UpGuard*, 2020. Accessed: 2024-06-28.
15. J. Wakefield. EasyJet admits data of nine million hacked. *BBC News, Technology report*, 2019. Accessed: 2024-06-28.
16. Jos Wetzels. Open sesame: The password hashing competition and Argon2. Cryptology ePrint Archive, Report 2016/104, 2016. `https://eprint.iacr.org/2016/104`.
17. D. Winder. 235 Million Instagram, TikTok and YouTube User Profiles Exposed In Massive Data Leak. *Forbes*, 2020. Accessed: 2024-06-28.

Part IV
Research Papers

Isogenies Demystified

Luca De Feo

IBM Research GmbH, Switzerland
aracne-2021@defeo.lu

Isogenies, what are they? Like the character in Alessandro Manzoni's novel, cryptographers encountering an isogeny in a textbook would have been justified, ten years ago, asking themselves "*chi era costui?*"[1] Elliptic curves, that, we know. But isogenies? The name sounds familiar, it must be one of those math things starting in iso-; but *chi diavolo era costui?*[2]

That is no longer true. Any self-respecting cryptographer nowadays must have at least a vague idea of what isogenies are and how they are used in cryptography. This survey is your guide to the supersingular isogeny galaxy [22].

Aufstieg und Fall der Elliptische Kurven Kryptografie

Elliptic curves are today a fact of life. I sometimes wonder what Euler would have thought of it. Not a single day goes without billions of elliptic curve operations being performed by servers, laptops, smartphones and even refrigerators throughout the world. I suppose Euler would have loved the fridge.

Why are elliptic curves so important in cryptography? One reason is that they are the closest thing we know to a *generic group*. What cryptographers ask from a group is: to be abelian, to be finite, to have efficient algorithms for testing membership, equality, and for evaluating the group operation. Any group well mannered enough to do exactly what is asked from it, and nothing more, is called *generic*.

The most important operation for a cryptographic group is *exponentiation*:

$$\begin{aligned} \exp_g : \mathbb{Z} &\to G, \\ x &\mapsto g^x. \end{aligned}$$

That $\exp_g(n)$ can be evaluated using $O(\log(n))$ generic group operations is obvious. What makes a group precious is the inverse map $g^x \mapsto x$, the *discrete logarithm*, being "difficult" to compute. Then $\exp_g$ is what most cryptographers call a *one-way function*.

De Cifris Koine – DE CIFRIS SEMINARS –https://doi.org/10.69091/koine/vol-2-R01

[1] "Who was he?", inquires Don Abbondio upon reading the name of Carneades in a hagiography of St Charles Borromeo.

[2] There would be much to say about how Manzoni's *faux savant* characters, from Don Abbondio to Don Ferrante, speak of our time. But this is an article about isogenies.

The other reason they are loved, and I claim it is the most important one, is that no knowledge of elliptic curves is required in order to make cryptography out of them. Cryptography is like humanity in Plato's cave: it only sees the tame generic group shadow of a wild real world elliptic curve. Do not get me wrong: this is great! We wouldn't have as powerful cryptographic tools, if creating them required a deep knowledge in number theory. We do not have such a luxury with isogenies.

What can you do with a generic group? A lot of things. I am sure the reader is familiar with the Diffie–Hellman key exchange [29], but I would like to highlight a different application. A *commitment scheme* is the cryptographic equivalent of a sealed envelope: in the first phase a party *commits to* a message m (*e.g.*, a monetary offering) by publishing the *commitment* $\mathcal{C}(m;r)$, where r represents an arbitrary auxiliary input (typically, some random bits); in the second phase, the party *opens* the commitment by revealing m and r; anyone can check that m is the message originally committed to by recomputing $\mathcal{C}(m;r)$. A cryptographic commitment must satisfy two properties: it must be *binding*, *i.e.*, after having committed to $\mathcal{C}(m;r)$ it must be difficult for the party to find

$$(m', r'),\ \ m \neq m' \quad \text{such that} \quad \mathcal{C}(m';r') = \mathcal{C}(m;r)\,.$$

It must also be *hiding*, *i.e.*, given only $\mathcal{C}(m;r)$ it must be difficult to deduct m.

Given a generic group G with some fixed generator g, it is easy to imagine a simple commitment scheme defined by $\mathcal{C}(m) = g^m$. This scheme is obviously binding if $0 < m < \#G$, and is hiding thanks to the one-wayness of the $\exp_g$ function. However, while the binding property is *perfect* (it's impossible for the party to cheat), the hiding property only holds against *computationally bounded* adversaries, rather than in an information-theoretic sense. This may be a problem if, for example, the messages m are likely to be taken from a small subset.

Pedersen [56] is credited with a very simple and elegant idea to obtain a perfectly hiding commitment scheme from generic groups. Let g and h be two random generators of G, he defined $\mathcal{C}(m;r) = g^m h^r$, where r is a random integer in $[1, \#G]$. It is easy to see that Pedersen's commitment is perfectly hiding, thanks to h^r being uniformly distributed in G. For the binding property, it is capital that the discrete logarithm relation between g and h is unknown to the committer; indeed, given $x = \log_g(h)$ the commitment simply becomes g^{m+xr}, and breaking binding simply amounts to solving the equation $m + xr = m' + xr'$ modulo $\#G$.

Pedersen commitments can do much more than just emulate digital envelopes, and in fact a great variety of cryptographic protocols is based on them and similar ideas. Most of the advanced cryptographic protocols used nowadays, such as the Signal protocol used by WhatsApp, or those used in privacy-preserving cryptocurrencies, use some advanced features of generic groups such as Pedersen commitments; and their generic group of choice is, inevitably, elliptic curves.

But the reader knows the story by now: our world is coming to an end, Shor's bane is free [66], soon hordes of quantum computers will roam the earth, mercilessly

hunting down discrete logarithms and composite integers, our mobile data plans will evaporate in just days to accommodate for post-quantum cryptography.

Isogeny graphs

I will assume some familiarity with elliptic curves and abstract algebra. At this point, we're obliged to chose a camp in a controversy as ancient as "Emacs vs vi": unlike cryptographers, algebraists like to write abelian groups additively. We will side with the algebraists and rewrite exponentiation as

$$[n]P \equiv \exp_P(n),$$

with the side-effect of loosing track of the original meaning of "discrete logarithm".

The *multiplication map* $[n]$ is an example of a morphism from an elliptic curve to itself. Isogenies are generalizations of these morphisms, when we view elliptic curves both as groups and as algebraic varieties.

Definition 1. *Let $\varphi : E \to E'$ be a map between two elliptic curves defined over an algebraically closed field, the following are equivalent:*

1. *φ is a surjective group morphism,*
2. *φ is a group morphism with finite kernel,*
3. *φ is a non-constant algebraic map of projective varieties sending the point at infinity of E onto the point at infinity of E'.*

In any of these cases, φ is called an isogeny*; or an* endomorphism *when $E = E'$.*

In cryptography, however, we typically deal with non-algebraically closed fields. In this case we need to take *rationality* into account. Let k be a field with algebraic closure $\bar{k}$. By E/k we mean a curve defined over k, *i.e.*, whose equation has coefficients in k. We can extend scalars to $\bar{k}$, and see E as a curve over $\bar{k}$; when it is necessary to distinguish between them, we will write $E(\bar{k})$ for the group of points in the algebraic closure, and $E(k)$ for the group of *k-rational* points. Then the Galois group of $\bar{k}/k$ acts on $E(\bar{k})$ by permuting its elements.

Definition 2. *Let E, E' be elliptic curves defined over k. Let $\varphi : E \to E'$ be an isogeny. We say that φ is* defined over k*, or* k-rational *if any of the following equivalent conditions holds.*

1. *$\sigma(\ker\varphi) = \ker\varphi$ for any $\sigma \in \mathrm{Gal}(\bar{k}/k)$,*
2. *$\sigma \circ \varphi = \varphi \circ \sigma$ for any $\sigma \in \mathrm{Gal}(\bar{k}/k)$,*
3. *φ is expressed by rational fractions with coefficients in k.*

Note that if φ is k-rational, the points in $\ker\varphi$ are not necessarily defined over k. To give a complete introduction to isogenies we would need to define separability vs inseparability, degree, and more. However to keep this presentation light we will skip these, and direct the curious reader to [67], [54], [23]. Here, unless stated otherwise, by *ℓ-isogeny* we mean a *separable* isogeny of *degree* $\ell = \#\ker\varphi$. The important property to keep in mind is that the degree is multiplicative:

$$\deg(\psi\circ\phi) = \deg(\psi)\deg(\phi).$$

Theorem 1 (Dual isogeny theorem). *Let $\varphi : E \to E'$ be an isogeny of degree m. There is a unique isogeny $\hat{\varphi} : E' \to E$ of degree m, called the* dual isogeny*, such that*

$$\hat{\varphi}\circ\varphi = [m]_E, \quad \varphi\circ\hat{\varphi} = [m]_{E'}.$$

Example. The map φ from the elliptic curve $y^2 = x^3 + x$ to $y^2 = x^3 - 4x$ defined by

$$\varphi(x,y) = \left(\frac{x^2+1}{x}, y\frac{x^2-1}{x^2}\right), \qquad \varphi(0,0) = \varphi(\mathcal{O}) = \mathcal{O}$$

is a separable isogeny between curves defined over $\mathbb{Q}$. It has degree 2, and its kernel is generated by the point $(0,0)$. Its dual is defined by

$$\hat{\varphi}(x,y) = \left(\frac{x^2-4}{4x}, y\frac{x^2+4}{8x^2}\right), \qquad \hat{\varphi}(0,0) = \varphi(\mathcal{O}) = \mathcal{O}.$$

Isogenies have been used in cryptography since the early days of Elliptic Curve Cryptography, most notably within the Schoof–Elkies–Atkin point counting algorithm [65]. But there is a general agreement that Isogeny Based Cryptography starts from the moment one stops focusing on a single elliptic curve with its isogenies, and *zooms out* to encompass *all* elliptic curves with isogenies between them.

An *isogeny graph* is a multi-graph whose vertices represent elliptic curves, and whose edges represent isogenies. By putting different kinds of restrictions on the curves and the isogenies, we obtain distinct isogeny graphs with interesting properties.

In general, it is easier to think of the vertices as isomorphism[3] classes of elliptic curves. Conveniently, the j-invariant classifies elliptic curves up to isomorphism (over the algebraic closure), thus we typically attach a single j-invariant to each vertex. Sometimes, a finer notion of isomorphism will have to be considered (*e.g.*, isomorphism over the base field k), and a different invariant corresponding to this isomorphism type will be used instead (*e.g.*, a Montgomery A-invariant as used in CSIDH).

[3] An isomorphism is an isogeny of degree 1, *i.e.*, a bijective isogeny.

For the edges, we will usually restrict to isogenies of a given degree, or possibly of degree taken in some small list. Hence, isogeny graphs will tend to be undirected (representing an isogeny and its dual by the same undirected edge), and regular (*e.g.*, for any prime ℓ different from the characteristic, any curve has exactly $\ell + 1$ isogenies of degree ℓ in the algebraic closure).

Figures 1 and 2 show two important examples of isogeny graphs. On the right, the graph of all supersingular curves defined over $\mathbb{F}_{89}$, up to $\mathbb{F}_{89}$-rational isomorphisms. These are the curves of j-invariant 0, 66, 52, 13, 7 or 6; each j-invariant being repeated twice, because each curve has a non-$\mathbb{F}_{89}$-isomorphic copy called the *quadratic twist*. The edges are the union of three distinct edge sets (represented by different colors), corresponding to the $\mathbb{F}_{89}$-rational isogenies of degree 3, 5 and 7, respectively.

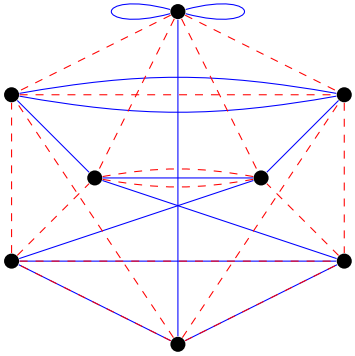


Fig. 1. The supersingular isogeny graphs of degree 2 (blue, continuous) and 3 (red, dashed) on $\mathbb{F}_{97^2}$.

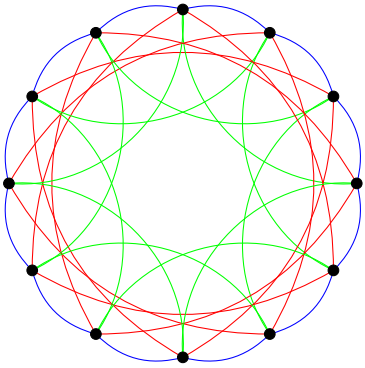


Fig. 2. The supersingular isogeny graphs of degree 3 (blue), 5 (red) and 7 (green), restricted to $\mathbb{F}_{89}$-rational isomorphism classes.
Also the connected component of $j = 77$ in the ordinary isogeny graph of $\mathbb{F}_{233}$ (same isogeny degrees).

This graph also occurs as a connected component of infinitely many ordinary graphs, for example the component containing the j-invariants 20, 28, 40, 77, 86, 87, 118, 136, 138, 142, 184 and 194 over $\mathbb{F}_{233}$ —in the ordinary case, the quadratic twists form a distinct, graph-isomorphic component—. The edges represent $\mathbb{F}_{233}$-rational isogenies of the same degrees as before.

This graph is in fact none else than the Cayley graph of the additive group $\mathbb{Z}/12\mathbb{Z}$, generated by 1, 3 and 4. The reason why it is such a common isogeny graph will become clear in the next section.

The graph on the left is different. Its vertices are all supersingular j-invariants in the algebraic closure of $\mathbb{F}_{97}$. A classical theorem shows that all supersingular invariants in characteristic p are defined in $\mathbb{F}_{p^2}$, and thus there is a finite number of supersingular isomorphism classes. The same theorem also shows that all supersingular isogenies are defined over $\mathbb{F}_{p^2}$. The figure presents two graphs (in different colors): a 3-regular graph whose edges are all isogenies of degree 2, and a 4-regular one whose edges are all isogenies of degree 3. The central symmetry visible to the naked eye is due to the Frobenius involution of $\mathbb{F}_{p^2}/\mathbb{F}_p$.

These graphs are essentially unique: they do not occur as isogeny graphs of any other elliptic curves on any other field. They are usually called *full supersingular isogeny graphs*, although the "full" and the "isogeny" are often dropped. Here is an interesting empirical study [2], and a database of the smallest ones [32].

Endomorphism rings

Everything about isogeny graphs can be understood via *endomorphism rings*. Endomorphisms of elliptic curves form a ring, under addition and composition.[4] Their structure is well understood: they are free $\mathbb{Z}$-modules of dimension 1, 2 or 4. There is more: if we exclude the subring $\mathbb{Z} \subset \mathrm{End}(E)$, any endomorphism is a quadratic integer, i.e., it is annihilated by a monic quadratic polynomial with integer coefficients. These constraints leave only a handful of possible choices.

Theorem 2. *Let E be an elliptic curve over a field of characteristic p, its endomorphism ring is isomorphic to one of the following:*

1. *the ring of integers, only if $p = 0$,*
2. *an order in a quadratic imaginary number field,*
3. *only if $p \neq 0$, a maximal order in the quaternion algebra ramified at p and infinity.*

In positive characteristic, the second case is called ordinary *and the third* supersingular.

If $\phi : E \to E'$ is an isogeny, $\hat{\phi} : E' \to E$ its dual, and $\omega : E \to E$ an endomorphism of E, then $\phi\omega\hat{\phi}$ is an endomorphism of E'. It stands to reason that the

[4] A common source of confusion is that an extra *null endomorphism* must be added to the set in order to make it a ring, although, by definition, a constant map does not qualify as an isogeny.

endomorphism rings of E and E' must be somehow related. Indeed, to any separable isogeny ϕ we can associate its *kernel ideal* $I_\phi \subset \mathrm{End}(E)$, defined by

$$I_\phi = \{\omega \in \mathrm{End}(E) \mid \omega(\ker \phi) = \{\mathcal{O}\}\},$$

and it turns out we can extend[5] this correspondence to a bijection between isogenies and ideals.

Then, for any ideal $I_\phi \subset \mathrm{End}(E)$ with associated isogeny $\phi : E \to E'$, we define the operation $\star$ by $I_\phi \star E \equiv E'$. We say that two ideals I, J are *equivalent* if $I \star E = J \star E$, or equivalently if $nI = J \cdot (\omega)$ for some integer n and some principal ideal (ω). We call *ideal class* a set of equivalent ideals. In general these classes do not have a simple algebraic structure, however, if we restrict them in an appropriate manner, $\star$ becomes a *group action* by an *ideal class group*. The simplest such case is the object of the *fundamental theorem of complex multiplication*.

Theorem 3 (Complex multiplication). *Let $\mathbb{F}_q$ be a finite field, let $\mathcal{O} \subset \mathbb{Q}(\sqrt{-D})$ be a quadratic imaginary order, denote by $\mathrm{Ell}_q(\mathcal{O})$ the set of elliptic curves over $\mathbb{F}_q$ with endomorphism ring isomorphic to $\mathcal{O}$ and assume it is non-empty. The operation $\star$ defines an action of the group of invertible fractional ideals of $\mathcal{O}$ on $\mathrm{Ell}_q(\mathcal{O})$, and the action factors through the subgroup of principal ideals. Said otherwise, the* class group $\mathrm{Cl}(\mathcal{O})$ *acts regularly on $\mathrm{Ell}_q(\mathcal{O})$.*

Similar statements hold when E is supersingular and $\mathcal{O} \subset \mathrm{End}(E)$ is a quadratic order. An easy case is when E is defined over a prime field $\mathbb{F}_p$: then the subring $\mathrm{End}_{\mathbb{F}_p}(E) \subset \mathrm{End}(E)$ of $\mathbb{F}_p$-rational endomorphisms is isomorphic to one of $\mathcal{O} = \mathbb{Z}[\sqrt{-p}]$ or $\mathcal{O} = \mathbb{Z}[(1 + \sqrt{-p})/2]$, and if we define $\mathrm{Ell}_p(\mathcal{O})$ as the set of all supersingular curves over $\mathbb{F}_p$ such that $\mathrm{End}_{\mathbb{F}_p}(E) \simeq \mathcal{O}$, the group $\mathrm{Cl}(\mathcal{O})$ acts regularly on $\mathrm{Ell}_p(\mathcal{O})$ like in the complex multiplication case [28].

These facts explain why the graph in Figure 2 is isomorphic to a Cayley graph of $\mathbb{Z}/12\mathbb{Z}$. To construct the examples, we chose $\mathcal{O} \simeq \mathbb{Z}[\sqrt{-89}]$, which has class group isomorphic to $\mathbb{Z}/12\mathbb{Z}$ and is generated by an ideal of norm 3 (more formally, an ideal class representing 3), corresponding to isogenies of degree 3 (the blue edges). Thus $\mathrm{Ell}_{89}(\mathcal{O})$ is the set of all $\mathbb{F}_{89}$-rational supersingular curves, but also, for any p such that -89 is a square modulo p, there exists a power q of p such that $\mathrm{Ell}_q(\mathcal{O})$ is non-empty. In any of these cases, $\mathrm{Cl}(\mathcal{O})$ acts faithfully and transitively on $\mathrm{Ell}_q(\mathcal{O})$, and the action of a basis of elements of $\mathrm{Cl}(\mathcal{O})$ can be visualized as a Cayley graph.

While Theorem 3 describes almost completely isogeny graphs of ordinary curves, the picture for supersingular graphs is still blurry. Mestre [53], then Pizer [59, 60], then Kohel [48] showed that full supersingular graphs are connected, regular, and satisfy the *Ramanujan property*, i.e., they are optimal expanders [41].

[5] The correspondence for inseparable isogenies is slightly more technical, and we are forced to omit the details.

CSIDH...

And we are back to isogeny based cryptography 101: key exchange.

Couveignes [21] was the first to propose a key exchange scheme based on the group action of complex multiplication, however his work stayed mostly unknown. His ideas were independently rediscovered ten years later by Rostovtsev and Stolbunov [63], who were the first to suggest isogenies may be good candidates for constructing quantum-resistant schemes.

Replicating the Diffie–Hellman key exchange with a *cryptographic group action* is almost immediate. Given a finite abelian group G acting regularly on a set X, given a *starting element* $x_0 \in X$, let secret keys be random elements $a, b \in G$, and define public keys as $x_a = a \star x_0$ and $x_b = b \star x_0$. Then, the shared secret is obtained as

$$a \star x_b = (ab) \star x_0 = b \star x_a.$$

This key exchange is secure if the analogue of the Diffie–Hellman assumption holds for the group action $(G, X, \star)$.

However the case of the complex multiplication group action $(\mathrm{Cl}(\mathcal{O}), \mathrm{Ell}_q(\mathcal{O}), \star)$ is more complicated for a number of reasons:

1. It is usually not possible to test equality in $\mathrm{Cl}(\mathcal{O})$, nor to sample uniformly from it;
2. Evaluating $a \star x$ cannot be done in polynomial time for a majority of inputs a, even though every element $a \in \mathrm{Cl}(\mathcal{O})$ does have a representation that supports fast evaluation of the group action.

These two limitations follow from two fundamental algorithmic obstacles:

1. The order, and thus also the group structure of $\mathrm{Cl}(\mathcal{O})$ is generally unknown. Indeed, the best classical algorithm to compute the group structure of $\mathrm{Cl}(\mathcal{O})$ is a type of index calculus, with subexponential complexity $L_{\#\mathcal{O}}(1/2)$. The current record is the computation for the class group of discriminant $4 \cdot 587 \cdot \prod_{i=1}^{73} \ell_i$, where ℓ_i are the first 73 odd primes [6], which took about 52 core years on an inhomogeneous cluster. Unfortunately discriminants used in isogeny-based cryptography may be larger. The good news is that computing the structure of $\mathrm{Cl}(\mathcal{O})$ is precisely as difficult as breaking RSA for a quantum computer, thus we only have to wait!
2. The cost of evaluating the action of an ideal $I_\phi \subset \mathrm{End}(E)$ is polynomial in the norm of the ideal, i.e., in the degree of the associated isogeny ϕ. This severely limits the kind of ideals for which it is feasible to evaluate the action $\star$.

Before giving the solution to this conundrum, let's take a step back and see how classical discrete logarithms are related to Cayley graphs. Given a group G of prime order p, exponentiation defines a regular action of $(\mathbb{Z}/p\mathbb{Z})^\times$ on $G \backslash \{1\}$ by

$$a \star g \equiv g^a.$$

From a subset $S \subset (\mathbb{Z}/p\mathbb{Z})^\times$, we may construct the Cayley graph $[(\mathbb{Z}/p\mathbb{Z})^\times, S]$. For example, the graph in Figure 2 can be equally seen as the Cayley graph of $(\mathbb{Z}/13\mathbb{Z})^\times$ generated by $S = \{2, 8, 3\}$. The same graph can be equally interpreted as the graph whose vertices are non-identity elements of G, and where two vertices g, h are connected whenever $h = g^a$ for some $a \in S$. This graph is sometimes called the *Schreier graph* $(\star, S)$.

Given two elements $g, h \in G$, finding a path between them in the Schreier graph is equivalent to computing their discrete logarithm. There is only one gotcha: the path must be *short*, e.g., of polynomial length in $\log(\#G)$, otherwise the solution is practically useless. Intuitively, the larger S, the smaller the diameter of the graph, and indeed it is well known that Cayley graphs tend to make good expanders.

If we take S large enough, then we even have an effective way to sample random elements in G nearly uniformly: it is sufficient to start from an arbitrary generator of G, and perform a random walk of polynomial length in $\log(\#G)$. This fact can be used to construct a key exchange similar to Diffie–Hellman: fix a starting generator g, sample random walks $(a_1, \dots, a_n), (b_1, \dots, b_n) \in S^*$, define as public keys

$$g^a = g^{\prod a_i}, \quad g^b = g^{\prod b_i},$$

then the shared secret g^{ab} is obtained by replaying the same random walks from g^b and g^a respectively.

Coming back to the complex multiplication group action, Jao *et al.* [44] proved, assuming the generalized Riemann hypothesis, that Cayley graphs $[\mathrm{Cl}(\mathcal{O}), S]$ form an expander family as soon as $\#S \in O\big(\log(\#\mathrm{Cl}(\mathcal{O}))^2\big)$. Following the previous sketch, we immediately obtain a key exchange scheme based on complex multiplication.

Setup. Choose a quadratic imaginary order $\mathcal{O}$ and an elliptic curve $E_0 \in \mathrm{Ell}(\mathcal{O})$. Fix a set $\{s_1, \dots, s_n\} \subset \mathrm{Cl}(\mathcal{O})$ of ideal representatives of small norm.

Public key generation. Sample a random integer vector $(e_1, \dots, e_n)$ and construct the ideal

$$I = \prod_{i=1}^{n} s_i^{e_i};$$

output the public key $I \star E_0$.

Shared secret computation. Given a public key E, and a secret ideal $I \subset \mathcal{O}$, output the shared secret $I \star E$.

This is precisely the key exchange scheme of Couveignes, Rostovtsev and Stolbunov, although we have left some details unspecified: how to choose $\mathcal{O}$, how to find E_0, how to compute the group action, ... For a long time, the only known way to instantiate the scheme produced a system too slow to be useful in practice, a fact reported as recently as 2018 [25]. A breakthrough came the same year, though, with the invention of CSIDH[6] [12], an instantiation based on the action of $\mathrm{Cl}(-p)$

[6] Pronounced like *"sea side"*.

on the set of supersingular curves defined over $\mathbb{F}_p$. Parameters in CSIDH are chosen as follows:

- p is a prime such that $p+1 = 4\prod \ell_i$, where ℓ_i is a set of small odd primes. For a target classical security of λ bits, $\log_2(p)$ needs to be approximately 4λ.
- The quadratic order is $\mathcal{O} = \mathbb{Z}[\sqrt{-p}]$. The set of ideal representatives of small norm is taken to be $s_i = (\ell_i, 1-\sqrt{-p})$, for each of the ℓ_i in the factorization of $p+1$.
- Thanks to the constraints on p, the elliptic curve of equation $y^2 = x^3 + x$ is supersingular, has $\mathbb{F}_p$-rational endomorphism ring isomorphic to $\mathcal{O}$, and is thus taken as E_0.
- The secret vectors (e_i) are sampled from an integer box $[-B, B]^n$, such that $(2B+1)^n \approx \sqrt{p}$.

These choices make for a surprisingly simple algorithm to evaluate the group action $\star$. Indeed, after identifying $\sqrt{-p}$ to the Frobenius endomorphism of any curve $E \in \mathrm{Ell}(\mathcal{O})$, the isogeny kernel associated to $s_i = (\ell_i, 1-\sqrt{-p})$ is simply the set

$$E[\ell_i] \cap E(\mathbb{F}_p)$$

of $\mathbb{F}_p$-rational points of order ℓ_i. Given this kernel, Vélu's formulas [70, 55, 62] efficiently compute the associated isogeny and the image curve $s_i \star E$ using $O(\ell_i)$ finite field operations.[7] Furthermore, it is not difficult to see that the inverse ideal class s_i^{-1} is represented by $(\ell_i, 1+\sqrt{-p})$, and the associated kernel is the set of points of order ℓ_i that have abscissa in $\mathbb{F}_p$ and ordinate in $\mathbb{F}_{p^2}$. Thus, the action of any ideal $s_i^{\pm e}$ can be evaluated by e applications of Vélu's formulas, and the action of the product ideal $\prod s_i^{e_i}$ is simply the composition of each individual action. The total cost of evaluating the CSIDH group action is thus $\sim B\sum \ell_i$ finite field operations, ignoring some not-so-negligible computations such as finding the generators of the various isogeny kernels.

It is worth pointing out that Jao *et al.*'s theorem does not apply to the CSIDH graph, because its degree of regularity is in $O(\log_2(p))$ rather than in $O(\log_2(p)^2)$; nevertheless, reasonable heuristics let us still argue that the graph has good expansion properties, and thus that the distribution of public keys is practically indistinguishable from uniform.

[7] In a recent development [4], the upper bound on the complexity of computing the isogeny action has been improved to $\tilde{O}(\sqrt{\ell_i})$.

. . . and SIDH

For all its elegance and simplicity, CSIDH has a serious drawback when it comes to quantum security, as we shall see next. Its evil twin SIDH[8] [42, 30] was designed to overcome this limitation.

The goal of SIDH is to be able to perform a key exchange based on random walks in the full supersingular graph. Since no group acts on the full graph, constructing commuting isogeny walks is not obvious; however the absence of a group action is also what makes attacking SIDH more difficult.

But let's start from the kind of isogeny walks we perform in SIDH. As we know, in CSIDH an isogeny walk is defined by a list $(e_1, \ldots, e_n)$ of integers. Each integer corresponds to a different isogeny degree ℓ_i, the magnitude $|e_i|$ indicates the number of steps to travel along the ℓ_i-isogeny cycle (each cycle is represented by a different color in Figure 2), and the sign of e_i means "go forward" or "go backward" (the meaning to the orientation was given by the Frobenius endomorphism). The order in which the different primes ℓ_i are processed is irrelevant, as we know that isogenies correspond to an abelian group action.

It is absolutely necessary that CSIDH uses a fairly large collection of primes ℓ_i. Indeed, if the vectors (e_i) are selected from a box $[-B, B]^n$, then the number of distinct end points for these isogeny walks is at most $(2B + 1)^n$, but the cost of computing one walk is proportional to Bn. Said otherwise, the only parameter in which the key space size grows exponentially (compared to the cost of executing the key exchange) is the number of distinct primes ℓ_i.

Moving to the full supersingular graph the outlook changes. There are approximately $p/12$ supersingular isomorphism classes in the algebraic closure of $\mathbb{F}_p$, and they are all defined over $\mathbb{F}_{p^2}$. Over $\mathbb{F}_{p^2}$, every supersingular curve has exactly $\ell + 1$ distinct isogenies for any prime ℓ, i.e., the ℓ-isogeny graph is $(\ell+1)$-regular.[9] Hence, unlike in the complex multiplication case, starting from any supersingular curve E there are exactly $(\ell + 1)\ell^n$ distinct non-backtracking[10] ℓ-isogeny walks of length $n+1$, instead of just 2. Furthermore, the Ramanujan property proved by A. K. Pizer [59, 60] indicates that the induced distribution on the vertices quickly approaches the uniform distribution as soon as $n \approx c_\ell \log(p)$ for some constant c_ℓ.

These facts were already exploited by Charles *et al.* [14] to construct a collision resistant hash function based on pseudo-random walks in supersingular 2-isogeny

[8] Prounounced by spelling out the acronym *"ess-eye-dee-aitch"*.

[9] A small exception must be granted to the curves of j-invariant 0 or 1728, which have out-degree $\ell + 1$, but lower in-degree.

[10] Most theorems for random walks in graphs are stated for ordinary walks, where one undirected edge can be immediately followed by the same edge in the opposite direction. However, in our context, following an isogeny step by its dual is not interesting: it produces a scalar multiplication $[\ell]$ which is easily factored out of the walk, and does not contribute to the security of the cryptosystem.

graphs, and their work was indeed an inspiration for SIDH. In principle, we would like to find a way for two parties to perform walks in the ℓ-isogeny graph in such a way that the walks commute. However there is an obvious tension here: if the proverbial Alice and Bob each perform an isogeny walk of length n, call them A and B, and if the order of A and B does not count, i.e., $A \circ B = B \circ A$, then why would the order of the steps *within* A or B count, in general? Indeed we know no way to construct commuting walks in a supersingular graph in a way that is compatible with the security of a key exchange scheme.

The trick used by SIDH is to have Alice and Bob do walks in two different supersingular graphs on the same vertex set (see Figure 1). Fix two primes, say 2 and 3. Alice performs a random walk A in the 2-isogeny graph, while Bob performs a random walk B in the 3-isogeny graph. By coordinating their efforts carefully, they can ensure that $A \circ B = B \circ A$, and still get a secure key exchange protocol.

The way this works is astonishingly simple. A 2-isogeny walk of length n is nothing else than a composition of isogenies of degree 2, i.e., an isogeny of degree 2^n. Call ϕ_A this isogeny, and call R_A a point of order 2^n generating $\ker \phi_A$. Similarly, let ϕ_B be a 3^m-isogeny and let R_B be a generator of its kernel.

Then $R_A + R_B$ is a point of order $2^n 3^m$, and to it is associated a unique isogeny ϕ_{AB} of the same degree. Then, there exist isogenies ϕ'_A and ϕ'_B such that the following diagram commutes

$$\begin{array}{ccc} E_0 & \xrightarrow{\phi_A} & E_A \\ {\scriptstyle \phi_B}\downarrow & \searrow^{\phi_{AB}} & \downarrow{\scriptstyle \phi'_B} \\ E_B & \xrightarrow[\phi'_A]{} & E_{AB} \end{array} \tag{1}$$

From the diagram, it is clear that $\ker \phi'_A = \langle \phi_B(R_A) \rangle$ and $\ker \phi'_B = \langle \phi_A(R_B) \rangle$.

But we seem to have reached a dead end: ϕ_B is a secret of Bob's, and R_A is a secret of Alice; how can Bob safely make Alice aware of the value $\phi_B(R_A)$? The trick is to define public bases $E[2^n] = \langle P_A, Q_A \rangle$ and $E[3^m] = \langle P_B, Q_B \rangle$, and to write R_A (resp., R_B) as a secret linear combination of P_A, Q_A (resp., P_B, Q_B). Then, Bob transmits to Alice the values $\phi_B(P_A), \phi_B(Q_A)$, from which Alice can compute R_A without giving away her secret integers.

There is one trick left to make SIDH work. For the torsion bases and the isogenies to have compact and efficient representations, it is necessary to choose the curves very carefully, similarly to what we did for CSIDH. Ideally, we would like the torsion groups $E[2^n]$ and $E[3^m]$ to be defined over the base field $\mathbb{F}_{p^2}$, so that P_A, Q_A, P_B, Q_B are represented by a pair of elements of $\mathbb{F}_{p^2}$ each.[11] Over $\mathbb{F}_{p^2}$, there

[11] One can do even better and represent P_A, Q_A by a triplet of elements of $\mathbb{F}_{p^2}$ [18].

are two isogeny classes of supersingular curves: one with curves of order $(p+1)^2$, and one of order $(p-1)^2$, and their ℓ-isogeny graphs are isomorphic. It is customary to pick the first and choose p so that $p+1 = 2^n 3^m$, thus fulfilling our requirements.[12]

To summarize, SIDH can be instantiated as follows; we only give the operations for Alice: for Bob, just switch the roles of A and B.

Setup Choose a prime p of the form $p+1 = 2^n 3^m$. The starting curve is

$$E_0 \ : \ y^2 = x^3 + x.$$

Select arbitrary bases $E[2^n] = \langle P_A, Q_A \rangle$ and $E[3^m] = \langle P_B, Q_B \rangle$.

Public key generation Choose a random integer r_a, set $R_A = P_A + [r_a]Q_a$. Compute the isogeny $\phi_A : E_0 \to E_A$ of kernel R_A. Send $E_A, \phi_A(P_B), \phi_A(Q_B)$ to Bob.

Shared secret computation Upon receiving $E_B, \phi_B(P_A), \phi_B(Q_A)$, compute

$$R'_A = \phi_B(P_A) + [r_a]\phi_B(Q_A).$$

Compute the isogeny $\phi'_A : E_B \to E_{AB}$, output $j(E_{AB})$.

This particular instantiation closely matches the parameters chosen for the NIST candidate SIKE [3],[13] where the pair (n, m) is one of $(216, 137)$, $(250, 159)$, $(305, 192)$, $(372, 239)$, depending on the security level.

Breaking isogenies

What does it take to break an isogeny based cryptosystem? At the base of the pyramid, lies the fundamental problem of isogeny based cryptography.

Definition 3 (Isogeny walk problem). *Let E_0, E_1 two elliptic curves drawn at random from some isogeny class over some finite field k, find a k-rational isogeny $\phi : E_0 \to E_1$ of smooth degree.*

The "smooth degree" condition means that ϕ can be represented as a walk in some k-rational isogeny graph, which is an effective representation as long as the length of the walk is subexponential. It is clear that a solution to this problem breaks CSIDH, as it produces an ideal in the same ideal class as the secret key. It is less evident that it also breaks SIDH, but Galbraith *et al.* showed that this is indeed the case, assuming credible heuristics [38]. Galbraith *et al.*, as well as Castryck *et al.* [13], also showed that, for supersingular curves, the isogeny walk problem is heuristically equivalent to the following one.

[12] Costello [17] has recently explored variants of SIDH where the two torsion groups are spread over the two classes of order $(p+1)^2$ and $(p-1)^2$.

[13] SIKE is the encryption scheme derived from SIDH, the difference is purely semantical, technical steps stay unchanged.

Definition 4 (Endomorphism ring problem). *Given a random supersingular curve $E/\mathbb{F}_{p^2}$, compute a basis of its endomorphism ring.*

These two problems are the mainstays of isogeny based cryptography: break them, and all the field evaporates. However, no cryptosystem is actually based on them: in every case, some stronger assumption is needed to prove their security. For CSIDH, for example, the curve E_0 is usually fixed, and its endomorphism ring known. This is not a major problem if the curve E_1 is uniformly random: indeed, an algorithm solving this specialized variant of the problem can be applied twice to solve the general instance. However, in CSIDH the curve E_1 is not provably uniformly distributed, but rather assumed to be computationally indistinguishable from random. Admittedly, this is a minor departure from the original problem, and there is a consensus that the security of CSIDH is not far from that of the isogeny walk problem.

The situation of SIDH is more delicate:

- The curve E_0 is also fixed and of known endomorphism ring;
- The curve E_1 is very far from being uniformly random, as it is at distance $\approx \log_\ell(p)/2$ from E_0, considerably shorter than the diameter of the graph;
- On top of E_1, the SIDH protocol also publishes the evaluation points $\phi(P_B)$ and $\phi(Q_B)$, from whose knowledge one can compute the action of ϕ on any point of $E_0[3^m]$ (change B to A and 3 to 2 for Bob's isogeny).

The *SIDH assumptions* essentially state that it is fine to give out this additional information, however they are widely believed to be considerably stronger than the isogeny walk assumption, as indicated by the existence of *torsion point attacks* on overstretched variants of SIDH [58, 51].

Finally, some may object that the prime p used in SIDH or CSIDH has a very special form, and this should be taken into account when evaluating the strength of the related assumptions. However using special primes for efficiency has been a common practice in elliptic curve cryptography for decades, and it is widely believed that such specialization has negligible impact on security.

It thus appears that from an assumption "quality" perspective CSIDH is better positioned than SIDH. Unfortunately the order is reversed when we look at actual attacks. Indeed the isogeny walk assumption is not a single one, but rather a family of assumptions: one for each isogeny class considered. The isogeny class of SIDH comprises all supersingular curves over $\mathbb{F}_{p^2}$, and for that class no algorithm better than exponential is known to solve the isogeny walk problem. For the isogeny classes considered in CSIDH or in the earlier Couveignes–Rostovtsev–Stolbunov protocols, instead, a powerful quantum algorithm due to Kuperberg solves the problem in subexponential time [61, 49, 50, 16].

Kuperberg's is a generic algorithm for the *abelian hidden shift problem*, also known as the *dihedral hidden subgroup problem*: given a regular abelian group action $(G, X, \star)$ and a pair $x_0, x_1 \in X$, given quantum access to an oracle evaluating

$g \star x_0$ for arbitrary g, it finds the unique $\bar{g}$ such that $x_1 = \bar{g} \star x_0$. Its asymptotic complexity is roughly $\exp\left(\sqrt{\log(\#G)}\right)$, and thus CSIDH parameters must scale quadratically with the security level. However the exact quantum security of concrete CSIDH parameters is the subject of a heated debate, and a consensus has yet to be reached [7, 43, 5, 9, 57, 15].

On the classical front, things are simpler. The best classical attack on CSIDH has been known for 20 years: it is a simple meet-in-the-middle algorithm on the graph, running two pseudo-random walks in parallel until they meet [34, 37, 35, 28]. The CSIDH graph contains $O(p^{1/2})$ vertices, and thus the meet-in-the-middle attack finds a solution in $O(p^{1/4})$ steps on average, using a negligible amount of memory if a Pollard-rho style technique is used for collision detection. This justifies the 128-bits of classical security claim for the 511 bits prime CSIDH-512.

The same collision finding algorithm works equally well on the full supersingular graph; since the graph has $\approx p/12$ vertices, the algorithm runs in $O(\sqrt{p})$ time. In practice, Delfs and Galbraith [28] recommend working in two steps:

1. Find paths from $E_0 \to E_0'$ and $E_1 \to E_1'$ to curves E_0', E_1' defined over $\mathbb{F}_p$;
2. Use collision finding over the CSIDH graph to connect the paths.

While the asymptotic complexity is the same, this algorithm produces shorter walks and is easier to parallelize; its quantum version using Grover search runs in $O(p^{1/4})$ operations [8].

However neither algorithm is appropriate for SIDH. Indeed, as we mentioned, the secret isogeny in SIDH has unusually small degree $\approx 2^n \approx 3^m \approx \sqrt{p}$. Let's assume for concreteness that the degree is 2^n, then we may compute two sets: the set T_0 of all curves at distance $\lceil n/2 \rceil$ from E_0, and T_1 of those at distance $\lfloor n/2 \rfloor$ from E_1. We expect T_0 and T_1 to intersect in a single point, which is sometimes called a *claw* of T_0 and T_1. Using a $O(1)$ access time structure such as a hash table to store, say, T_0, this algorithm requires $O(p^{1/4})$ time and storage. Considerably better than the generic one.

It is however unrealistic to assume constant time access to such a huge amount of memory. Van Oorschot and Wiener's parallel collision search [69] provides a much more realistic solution to the claw finding problem, performing well in practice on parallel architectures with a limited amount of memory; with a constant amount of memory, it runs in asymptotic time $O(p^{3/8})$. The application to SIDH was analyzed in detail by Adj *et al.* [1], then by Costello *et al.* [19], and their conclusions were used to set parameters for the NIST candidate SIKE.

We note that no known attack is capable of exploiting the knowledge of the action of the secret isogeny on the torsion bases. So called *torsion point attacks* [58, 51] seem so far to only give an advantage against "overstretched" versions of SIDH where the degrees of the isogenies are exponentially larger than $\sqrt{p}$. It is an open question to determine whether the torsion point information in SIDH can be exploited in an attack.

Finally, quantum attacks. A generic claw finding algorithm by Tani [68] is claimed to break SIDH using $O(p^{1/6})$ time and memory. However a more in-depth analysis of the claims reveals that Tani's algorithm has no advantage over a simpler Grover search, and has thus a cost of $O(p^{1/4})$ at best, providing no speed-up over classical algorithms [45]. Quantum accelerations of van Oorschot and Wiener's collision search have recently been analyzed and shown not to invalidate the security claims of SIKE either [46].

CSIDH and SIDH are not the only existing isogeny based schemes. A larger variety of assumptions exists to support post-quantum signatures, identification protocols, oblivious transfer, and many more. Nevertheless, the best available attacks always come down to claw finding or Kuperberg's algorithm, depending on the target.

What now?

Let us come full circle and have a look back at Pedersen's commitments. There, we needed g and h, two random generators of a group G, and we formed the commitment $g^m h^r$ for message m and randomness r. Trying to port Pedersen's idea to isogenies, we may be tempted to interpret g and h as two distinct starting points in an isogeny graph, m and r as isogeny walks, g^m and h^r as their endpoints, or as $m \star g$ and $r \star h$ for those who prefer ideal action notation. However we are faced with two difficulties:

- What meaning to give to the product $g^m \cdot h^r$? In a group, this is a natural operation with homomorphic properties. But elliptic curve invariants of $m \star h$ and $r \star h$ do not support a natural homomorphic operation, and thus the hiding properties of Pedersen's commitment are lost.
- Recall that the discrete logarithm $\log_g(h)$ must be unknown to the committer for the commitment to be binding. How does one ensure that? In principle there could be a trusted authority who is in charge of generating g and h honestly, so that $\log_g(h)$ is unknown to anyone.
  In practice, trusted authorities are a burden, but there is a much simpler option available for many discrete logarithm groups G. A surjective function $H : \mathbb{Z} \to G$ is called a *hash into* G if given $(x, y, H(x), H(y))$ it is hard to compute $\log_{H(x)}(H(y))$. A non-example of hash is the map $x \mapsto g^x$ for some fixed generator g. An example of hash into the multiplicative group $\mathbb{F}_p^\times$ of a finite field is the map $x \mapsto (x \bmod (p-1)) + 1$. A hash into G can be used to generate Pedersen's base elements by setting $g = H(r_1)$ and $h = H(r_2)$ from some verifiable (pseudo)-random integers r_1, r_2 (e.g., some parts of the digits of π).
  In the realm of isogenies there is no efficient *hash into interesting isogeny classes*: we do not know how to generate random supersingular curves over $\mathbb{F}_p$, or over

$\mathbb{F}_{p^2}$, other than by starting from a well known supersingular elliptic curve (e.g., $y^2 = x^3 + x$) and performing a long enough random walk in some isogeny graph. This generation process is clearly the isogeny graph equivalent of the non-hash $x \mapsto g^x$. In fact, defining an efficient *hash into the supersingular set* is one of the major open questions in isogeny based cryptography [36], and the most "obvious" ideas have already been ruled out [13, 52].

There is no evidence that hashing in the supersingular set should be hard, and solving this problem would pave the way to many applications, such as making the SIDH assumptions weaker by using a verifiably random starting curve E_0, removing trusted setups from some protocols [27, 10], constructing efficient oblivious pseudo-random functions from CSIDH [47], and certainly many more.

The lack of a homomorphic operation on SIDH or CSIDH public keys, though, is an even greater problem. Not only it breaks the idea behind Pedersen commitments, but it is also the main obstacle to translating to the isogeny setting efficient discrete logarithm signature schemes such as Schnorr's [64] or ECDSA, and many more basic protocols known from discrete logarithms.

Which naturally brings us to the topic of signatures: as the reader may know, no isogeny based signatures were submitted to the NIST competition. Indeed, isogeny-based signatures tend to be extremely large and inefficient. The reason is that they are all obtained by applying the Fiat-Shamir transform [31] to hundreds of parallel executions of an interactive identification protocol, thus an SIDH or CSIDH based signature typically costs hundreds of times more than the corresponding key exchange scheme.

There is not much to SIDH signatures: they consist in proving knowledge of a secret isogeny by committing to the curves of a commutative square like in Eq. (1), and then revealing some but not all of the involved isogenies [30]. In practice they make for signatures in the hundreds of kilobytes, taking seconds to generate and verify [71].

CSIDH signatures offer more variety. They are somehow similar to discrete logarithm signatures: to prove knowledge of a secret ideal S such that $E_p = S \star E_0$, they commit to a random curve $E_r = R \star E_0$, then reveal RS^{-b}, in response to a binary challenge $b \in \{0, 1\}$. Compare this to Schnorr signatures where knowledge of the secret exponent in g^s is proven by committing to g^r and then revealing $r - cs$ for some challenge $c \in \mathbb{Z}/p\mathbb{Z}$. With such a large challenge space, the Schnorr protocol needs to be executed only once in order to produce an unforgeable signature. In contrast CSIDH can support a larger space only at the cost of an exponential increase in public key size: this produces decently short signatures, at the cost of several minutes for signing [24]. Signing times can be considerably reduced, though, if the structure of the class group is pre-computed, an extremely expensive task that we already discussed previously. This is the idea behind CSI-FiSh [6], the only practically usable isogeny-based signature until recently.

A third family of isogeny signatures is based on different assumptions than CSIDH or SIDH. At the hearth of these signatures, there is an interactive protocol to prove knowledge of the endomorphism ring of a supersingular curve; we already saw that this is heuristically equivalent to knowing an isogeny walk between a special starting curve such as $y^2 = x^3 + x$, and a random curve E_p. The idea is similar to CSIDH based signatures: first commit to a random curve $E_r = R \star E_p$, then respond to a challenge by revealing some ideal related to the secret. The first such protocol [39, 40] only accepted binary challenges and was notoriously difficult to implement, it has thus always been viewed as a purely theoretical effort. In a recent breakthrough SQISign, a similar signature scheme with exponentially large challenge space, has been introduced [26].

SQISign is not easy to implement, nor to analyze, however it boasts the shortest signature and public key combined size among all post-quantum candidates, by a fair margin. Signing time is not exactly fast, in the order of seconds, but verification is comparable in speed to SIDH or CSIDH.

Conclusions

To summarize, despite the similarities between CSIDH/SIDH and classic Diffie–Hellman, several challenges materialize when trying to rebuild on them most cryptographic protocols that we used to take for granted. Highly advanced techniques are needed even for the relatively basic task of signing, and for most other protocols we do not even have an isogeny based solution yet. Fortunately, there is a vast space of cryptographic possibilities as of yet unexplored.

At present, research on isogeny based cryptography mainly focuses on three areas: efficient implementations, both in software and hardware; cryptanalysis, both mathematical and physical; and achieving new primitives. I am happy to remark that there is more work in each of these areas than I could possibly cite in this short survey.

Turning to more prospective research, isogeny graphs of higher dimensional abelian varieties are still an insufficiently researched area. While some preliminary results indicate that they might not be the best candidates for basic schemes such as key exchange [33, 11, 20], there is still hope that the additional structure may be used to construct advanced functionalities. Another promising source of advanced protocols comes from the interplay between isogenies and pairings. Although it clearly cannot lead to post-quantum schemes, it has been recently used to realize some unique *time-release* primitives [27, 10].

My feeling is that we are only scratching the surface of isogeny-based cryptography, and that much more is to come. I hope this short and incomplete summary will motivate many of you to look more in depth into these topics!

References

1. Gora Adj, Daniel Cervantes-Vázquez, Jesús-Javier Chi-Domínguez, Alfred Menezes, and Francisco Rodríguez-Henríquez. On the cost of computing isogenies between supersingular elliptic curves. In Carlos Cid and Michael J. Jacobson Jr:, editors, *SAC 2018*, volume 11349 of *LNCS*, pages 322–343. Springer, Heidelberg, August 2019.
2. Sarah Arpin, Catalina Camacho-Navarro, Kristin Lauter, Joelle Lim, Kristina Nelson, Travis Scholl, and Jana Sotáková. Adventures in supersingularland. Cryptology ePrint Archive, Report 2019/1056, 2019. `https://eprint.iacr.org/2019/1056`.
3. R. Azarderakhsh, M. Campagna, C. Costello, L. De Feo, B. Hess, A. Hutchinson, A. Jalali, D. Jao, K. Karabina, B. Koziel, B. LaMacchia, P. Longa, M. Naehrig, G. Pereira, J. Renes, V. Soukharev, and D. Urbanikand. Supersingular isogeny key encapsulation. Technical report, University of Waterloo and evolutionQ, Inc., 2022.
4. Daniel J. Bernstein, Luca De Feo, Antonin Leroux, and Benjamin Smith. Faster computation of isogenies of large prime degree. Cryptology ePrint Archive, Report 2020/341, 2020. `https://eprint.iacr.org/2020/341`.
5. Daniel J. Bernstein, Tanja Lange, Chloe Martindale, and Lorenz Panny. Quantum circuits for the CSIDH: Optimizing quantum evaluation of isogenies. In Yuval Ishai and Vincent Rijmen, editors, *EUROCRYPT 2019, Part II*, volume 11477 of *LNCS*, pages 409–441. Springer, Heidelberg, May 2019.
6. Ward Beullens, Thorsten Kleinjung, and Frederik Vercauteren. CSI-FiSh: Efficient isogeny based signatures through class group computations. In Steven D. Galbraith and Shiho Moriai, editors, *ASIACRYPT 2019, Part I*, volume 11921 of *LNCS*, pages 227–247. Springer, Heidelberg, December 2019.
7. Jean-François Biasse, Annamaria Iezzi, and Michael J. Jacobson Jr. A note on the security of CSIDH. In Debrup Chakraborty and Tetsu Iwata, editors, *INDOCRYPT 2018*, volume 11356 of *LNCS*, pages 153–168. Springer, Heidelberg, December 2018.
8. Jean-François Biasse, David Jao, and Anirudh Sankar. A quantum algorithm for computing isogenies between supersingular elliptic curves. In Willi Meier and Debdeep Mukhopadhyay, editors, *INDOCRYPT 2014*, volume 8885 of *LNCS*, pages 428–442. Springer, Heidelberg, December 2014.
9. Xavier Bonnetain and André Schrottenloher. Quantum security analysis of CSIDH. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part II*, volume 12106 of *LNCS*, pages 493–522. Springer, Heidelberg, May 2020.
10. Jeffrey Burdges and Luca De Feo. Delay encryption. Cryptology ePrint Archive, Report 2020/638, 2020. `https://eprint.iacr.org/2020/638`.
11. W. Castryck, T. Decru, and B. Smith. Hash functions from superspecial genus-2 curves using Richelot isogenies. *Journal of Mathematical Cryptology*, 14(1):268–292, 2020.
12. Wouter Castryck, Tanja Lange, Chloe Martindale, Lorenz Panny, and Joost Renes. CSIDH: An efficient post-quantum commutative group action. In Thomas Peyrin and Steven Galbraith, editors, *ASIACRYPT 2018, Part III*, volume 11274 of *LNCS*, pages 395–427. Springer, Heidelberg, December 2018.
13. Wouter Castryck, Lorenz Panny, and Frederik Vercauteren. Rational isogenies from irrational endomorphisms. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part II*, volume 12106 of *LNCS*, pages 523–548. Springer, Heidelberg, May 2020.

14. Denis Xavier Charles, Kristin E. Lauter, and Eyal Z. Goren. Cryptographic hash functions from expander graphs. *Journal of Cryptology*, 22(1):93–113, January 2009.
15. Jorge Chávez-Saab, Jesús-Javier Chi-Domínguez, Samuel Jaques, and Francisco Rodríguez-Henríquez. The SQALE of CSIDH: Sublinear Vélu quantum-resistant isogeny action with low exponents. Cryptology ePrint Archive, Report 2020/1520, 2020. `https://eprint.iacr.org/2020/1520`.
16. A. Childs, D. Jao, and V. Soukharev. Constructing elliptic curve isogenies in quantum subexponential time. *Journal of Mathematical Cryptology*, 8(1):1–29, 2014.
17. Craig Costello. B-SIDH: supersingular isogeny Diffie-Hellman using twisted torsion. Cryptology ePrint Archive, Report 2019/1145, 2019. `https://eprint.iacr.org/2019/1145`.
18. Craig Costello, Patrick Longa, and Michael Naehrig. Efficient algorithms for supersingular isogeny Diffie-Hellman. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016, Part I*, volume 9814 of *LNCS*, pages 572–601. Springer, Heidelberg, August 2016.
19. Craig Costello, Patrick Longa, Michael Naehrig, Joost Renes, and Fernando Virdia. Improved classical cryptanalysis of SIKE in practice. In Aggelos Kiayias, Markulf Kohlweiss, Petros Wallden, and Vassilis Zikas, editors, *PKC 2020, Part II*, volume 12111 of *LNCS*, pages 505–534. Springer, Heidelberg, May 2020.
20. Craig Costello and Benjamin Smith. The supersingular isogeny problem in genus 2 and beyond. In Jintai Ding and Jean-Pierre Tillich, editors, *Post-Quantum Cryptography - 11th International Conference, PQCrypto 2020*, pages 151–168. Springer, Heidelberg, 2020.
21. Jean-Marc Couveignes. Hard homogeneous spaces. Cryptology ePrint Archive, Report 2006/291, 2006. `https://eprint.iacr.org/2006/291`.
22. L. De Feo. Key exchange in supersingular space-time, 2016. Divulgative (unpublished) article.
23. L. De Feo. Mathematics of isogeny based cryptography. arXiv preprint, 2017. `http://arxiv.org/abs/1711.04062`.
24. Luca De Feo and Steven D. Galbraith. SeaSign: Compact isogeny signatures from class group actions. In Yuval Ishai and Vincent Rijmen, editors, *EUROCRYPT 2019, Part III*, volume 11478 of *LNCS*, pages 759–789. Springer, Heidelberg, May 2019.
25. Luca De Feo, Jean Kieffer, and Benjamin Smith. Towards practical key exchange from ordinary isogeny graphs. In Thomas Peyrin and Steven Galbraith, editors, *ASIACRYPT 2018, Part III*, volume 11274 of *LNCS*, pages 365–394. Springer, Heidelberg, December 2018.
26. Luca De Feo, David Kohel, Antonin Leroux, Christophe Petit, and Benjamin Wesolowski. SQISign: Compact post-quantum signatures from quaternions and isogenies. In Shiho Moriai and Huaxiong Wang, editors, *ASIACRYPT 2020, Part I*, volume 12491 of *LNCS*, pages 64–93. Springer, Heidelberg, December 2020.
27. Luca De Feo, Simon Masson, Christophe Petit, and Antonio Sanso. Verifiable delay functions from supersingular isogenies and pairings. In Steven D. Galbraith and Shiho Moriai, editors, *ASIACRYPT 2019, Part I*, volume 11921 of *LNCS*, pages 248–277. Springer, Heidelberg, December 2019.
28. Christina Delfs and Steven D. Galbraith. Computing isogenies between supersingular elliptic curves over $\mathbb{F}_p$. *DCC*, 78(2):425–440, 2016.

29. Whitfield Diffie and Martin E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.
30. L. De Feo, D. Jao, and J. Plût. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *Journal of Mathematical Cryptology*, 8(3):209–247, 2014.
31. Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In Andrew M. Odlyzko, editor, *CRYPTO'86*, volume 263 of *LNCS*, pages 186–194. Springer, Heidelberg, August 1987.
32. E. Florit and G. Finol. Isogeny database. Online resource, 2020. https://isogenies.enricflorit.com/.
33. E. Victor Flynn and Yan Bo Ti. Genus two isogeny cryptography. In Jintai Ding and Rainer Steinwandt, editors, *Post-Quantum Cryptography - 10th International Conference, PQCrypto 2019*, pages 286–306. Springer, Heidelberg, 2019.
34. S. D. Galbraith. Constructing isogenies between elliptic curves over finite fields. *LMS Journal of Computation and Mathematics*, 2:118–138, 1999.
35. S. D. Galbraith and A. Stolbunov. Improved algorithm for the isogeny problem for ordinary elliptic curves. *Applicable Algebra in Engineering, Communication and Computing*, 24(2):107–131, 2013.
36. S. D. Galbraith and F. Vercauteren. Computational problems in supersingular elliptic curve isogenies. *Quantum Information Processing*, 17(10):265, 2018.
37. Steven D. Galbraith, Florian Hess, and Nigel P. Smart. Extending the GHS Weil descent attack. In Lars R. Knudsen, editor, *EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 29–44. Springer, Heidelberg, April / May 2002.
38. Steven D. Galbraith, Christophe Petit, Barak Shani, and Yan Bo Ti. On the security of supersingular isogeny cryptosystems. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *ASIACRYPT 2016, Part I*, volume 10031 of *LNCS*, pages 63–91. Springer, Heidelberg, December 2016.
39. Steven D. Galbraith, Christophe Petit, and Javier Silva. Identification protocols and signature schemes based on supersingular isogeny problems. In Tsuyoshi Takagi and Thomas Peyrin, editors, *ASIACRYPT 2017, Part I*, volume 10624 of *LNCS*, pages 3–33. Springer, Heidelberg, December 2017.
40. Steven D. Galbraith, Christophe Petit, and Javier Silva. Identification protocols and signature schemes based on supersingular isogeny problems. *Journal of Cryptology*, 33(1):130–175, January 2020.
41. S. Hoory, N. Linial, and A. Wigderson. Expander graphs and their applications. *Bulletin of the American Mathematical Society*, 43(4):439–561, 2006.
42. D. Jao and L. De Feo. Towards Quantum-Resistant cryptosystems from supersingular elliptic curve isogenies. In *Post-Quantum Cryptography*, volume 7071 of *LNCS*, pages 19–34. Springer, 2011.
43. D. Jao, J. LeGrow, C. Leonardi, and L. Ruiz-Lopez. A subexponential-time, polynomial quantum space algorithm for inverting the CM group action. *Journal of Mathematical Cryptology*, 14(1):129–138, 2020.
44. D. Jao, S. D. Miller, and R. Venkatesan. Expander graphs based on GRH with an application to elliptic curve cryptography. *Journal of Number Theory*, 129(6):1491–1504, 2009.

45. Samuel Jaques and John M. Schanck. Quantum cryptanalysis in the RAM model: Claw-finding attacks on SIKE. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part I*, volume 11692 of *LNCS*, pages 32–61. Springer, Heidelberg, August 2019.
46. Samuel Jaques and André Schrottenloher. Low-gate quantum golden collision finding. Cryptology ePrint Archive, Report 2020/424, 2020. `https://eprint.iacr.org/2020/424`.
47. Stanislaw Jarecki, Aggelos Kiayias, Hugo Krawczyk, and Jiayu Xu. Highly-efficient and composable password-protected secret sharing (or: How to protect your bitcoin wallet online). Cryptology ePrint Archive, Report 2016/144, 2016. `https://eprint.iacr.org/2016/144`.
48. D. Kohel. *Endomorphism rings of elliptic curves over finite fields*. PhD thesis, University of California at Berkley, 1996.
49. G. Kuperberg. A subexponential-time quantum algorithm for the dihedral hidden subgroup problem. *SIAM Journal of Computation*, 35(1):170–188, 2005.
50. G. Kuperberg. Another Subexponential-time Quantum Algorithm for the Dihedral Hidden Subgroup Problem. In *8th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2013)*, volume 22 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 20–34. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2013.
51. Péter Kutas, Chloe Martindale, Lorenz Panny, Christophe Petit, and Katherine E. Stange. Weak instances of SIDH variants under improved torsion-point attacks. Cryptology ePrint Archive, Report 2020/633, 2020. `https://eprint.iacr.org/2020/633`.
52. J. Love and D. Boneh. Supersingular curves with small non-integer endomorphisms. arXiv preprint, 2019. `https://arxiv.org/abs/1910.03180`.
53. J. Mestre. La méthode des graphes. Exemples et applications. In *Proceedings of the International conference on class numbers and fundamental units of algebraic number fields*, pages 217–242. Nagoya University, 1986.
54. J. S. Milne. *Elliptic Curves*. World Scientific, 2nd edition, 2020.
55. D. Moody and D. Shumow. Analogues of Vélu's formulas for isogenies on alternate models of elliptic curves. *Mathematics of Computation*, 85(300):1929–1951, 2016.
56. Torben P. Pedersen. Non-interactive and information-theoretic secure verifiable secret sharing. In Joan Feigenbaum, editor, *CRYPTO'91*, volume 576 of *LNCS*, pages 129–140. Springer, Heidelberg, August 1992.
57. Chris Peikert. He gives C-sieves on the CSIDH. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part II*, volume 12106 of *LNCS*, pages 463–492. Springer, Heidelberg, May 2020.
58. Christophe Petit. Faster algorithms for isogeny problems using torsion point images. In Tsuyoshi Takagi and Thomas Peyrin, editors, *ASIACRYPT 2017, Part II*, volume 10625 of *LNCS*, pages 330–353. Springer, Heidelberg, December 2017.
59. A. K. Pizer. Ramanujan graphs and Hecke operators. *Bulletin of the American Mathematical Society*, 23(1):127–137, 1990.
60. A. K. Pizer. Ramanujan graphs. In *Computational perspectives on number theory*, volume 7 of *AMS/IP Stud. Adv. Math.* American Mathematical Society, Providence, RI, 1998.

61. O. Regev. A subexponential time algorithm for the dihedral hidden subgroup problem with polynomial space. arXiv preprint, 2004. http://arxiv.org/abs/quant-ph/0406151.
62. Joost Renes. Computing isogenies between Montgomery curves using the action of (0, 0). In Tanja Lange and Rainer Steinwandt, editors, *Post-Quantum Cryptography - 9th International Conference, PQCrypto 2018*, pages 229–247. Springer, Heidelberg, 2018.
63. Alexander Rostovtsev and Anton Stolbunov. Public-Key Cryptosystem Based On Isogenies. Cryptology ePrint Archive, Report 2006/145, 2006. https://eprint.iacr.org/2006/145.
64. Claus-Peter Schnorr. Efficient identification and signatures for smart cards. In Gilles Brassard, editor, *CRYPTO'89*, volume 435 of *LNCS*, pages 239–252. Springer, Heidelberg, August 1990.
65. R. Schoof. Counting points on elliptic curves over finite fields. *Journal de Théorie des Nombres de Bordeaux*, 7(1):219–254, 1995.
66. Peter W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *35th FOCS*, pages 124–134. IEEE Computer Society Press, November 1994.
67. J. H. Silverman. *The arithmetic of elliptic curves*, volume 106 of *Graduate Texts in Mathematics*. Springer, New York, 1992.
68. S. Tani. Claw finding algorithms using quantum walk. *Theoretical Computer Science*, 410(50):5285–5297, 2009.
69. Paul C. van Oorschot and Michael J. Wiener. Parallel collision search with cryptanalytic applications. *Journal of Cryptology*, 12(1):1–28, January 1999.
70. J. Vélu. Isogénies entre courbes elliptiques. *Comptes Rendus de l'Académie des Sciences de Paris*, 273:238–241, 1971.
71. Youngho Yoo, Reza Azarderakhsh, Amir Jalali, David Jao, and Vladimir Soukharev. A post-quantum digital signature scheme based on supersingular isogenies. In Aggelos Kiayias, editor, *FC 2017*, volume 10322 of *LNCS*, pages 163–181. Springer, Heidelberg, April 2017.

Continued Fractions, Quadratic Fields, and Factoring: Some Computational Aspects

Michele Elia

Department of Telecommunications Engineering, Politecnico of Torino, Italy
michele.elia@polito.it

Introduction

In a letter to Pierre de Carcavi, August 14^{th} 1659, Pierre de Fermat reported several propositions; in particular, he stated the following theorem: *Every prime p of the form $4k+1$ is uniquely expressible as the sum of two squares, i.e.*

$$p = X^2 + Y^2 \Leftrightarrow p \equiv 1 \bmod 4,$$

whose first known proof was given by Euler using Fermat's *infinite descent* method. Many other proofs have been given, some constructive, others non-constructive; in particular, among the latter, Zagier's one-sentence proof deserves to be mentioned for its conciseness [17]. Among the numerous constructive proofs, two different proofs by Gauss stand out. The first is direct, and gives $x = \frac{(2k)!}{2(k!)^2} \bmod p$ and $y = \frac{((2k)!)^2}{2(k!)^2} \bmod p$; the partially incomplete proof was completed, a century later, by Jacobsthal. The second proof is based on quadratic forms of discriminant -4, and considers two equivalent principal quadratic forms with discriminant -4: $pX^2 + 2b_1XY + \frac{b_1^2+1}{p}Y^2$ and $x^2 + y^2$, where b_1 is a root of $z^2 + 1$ modulo p. The first form represents p trivially with $X = 1$ and $Y = 0$, thus Gauss' reduction [6] produces the unique reduced form in the class [11], and meanwhile yields x and y. Jacobsthal's constructive solution (1906) is based on counting the number of points on the elliptic curve $y^2 = n(n^2-a)$ in $\mathbb{Z}_p$. He considers the sum of Legendre symbols

$$S(a) = \sum_{n=1}^{p-1} \left(\frac{n(n^2-a)}{p} \right) \Rightarrow x = \frac{1}{2} S(q_R) \quad , \quad y = \frac{1}{2} S(q_N)$$

where $q_R, q_N \in \mathbb{Z}_p$ are any quadratic residue and non-residue, respectively, [8]. Legendre's proof is reported on pages 59-60 of [10]. It is constructive, since it yields X and Y from the complete remainder of the continued fraction expansion of $\sqrt{p}$. It is well explained in his own words

De Cifris Koine – DE CIFRIS SEMINARS –https://doi.org/10.69091/koine/vol-2-R02

> *... Donc tous le fois que l'équation* $x^2 - Ay^2 = -1$ *est résoluble (ce qui ha lieu entre autre cas lorsque A est un nombre premier* $4n + 1$*) le nombre A peut toujours être decomposé en deux quarrés; et cette décomposition est donnée immédiatement par lo quotient-complet* $\frac{\sqrt{A}+I}{D}$ *qui répond au second des quotients moyens compris dans la première période du développement de* $\sqrt{A}$*; le nombres I et D étant ainsi connu, on aura* $A = D^2 + I^2$*. Cette conclusion renferme un des plus beaux théorèmes de la science des nombres, savoir,* que tout nombre premier $4n + 1$ est la somme de deux quarrés; *elle donne en même temps le moyen de faire cette décomposition d'une manière directe et sans aucun* ***tâtonnement****.*

Thus, Legendre's proof gives the representation of any composite N such that the period of the continued fraction for $\sqrt{N}$ is odd, or equivalently, $x^2 - Ny^2 = -1$ is solvable in integers [5, 10, 15].
As a counterpart to Legendre's finding, when the period of the continued fraction expansion of $\sqrt{N}$ is even, we directly obtain, under mild conditions, a factor of a composite N. In particular, this is certainly the case when both prime factors of $N = pq$ are congruent 3 modulo 4 [5]. Legendre's solution of Fermat's theorem tacitly introduces a connection between continued fractions and the ramified primes of quadratic number fields, obviously without using this notion more than a century before Dedekind's invention.

Preliminaries

A regular continued fraction is an expression of the form

$$a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{a_3 + \cdots}}} \quad , \tag{1}$$

where a_0, a_1, $a_2, \ldots, a_i, \ldots$ is a sequence, possibly infinite, of positive integers. A convergent of a continued fraction is the sequence of fractions $\frac{A_m}{B_m}$, each of which is obtained by truncating the continued fraction at the $(m+1)$-th term. The fraction $\frac{A_m}{B_m}$ is called the m-th convergent [4, 7]. A continued fraction is said to be definitively periodic, with period τ, if, starting from a finite position n_o, a fixed pattern a'_1, $a'_2, \ldots, a'_\tau$ repeats indefinitely. Lagrange showed that any definitively periodic continued fraction, of period length τ, represents a positive number of the form $a + b\sqrt{N}$, $a, b \in \mathbb{Q}$, i.e. an element of $\mathbb{Q}(\sqrt{N})$, and conversely any such positive number is represented by a definitively periodic continued fraction [4, 15]. The period of the continued fraction expansion of $\sqrt{N}$ begins immediately after the first term a_0, and is written as $\sqrt{N} = \left[a_0, \overline{a_1, a_2, \ldots, a_2, a_1, 2a_0}\right]$, where the over-lined part is the period, which includes a palindromic part formed by the $\tau - 1$ terms

$a_1, a_2, \ldots, a_2, a_1$. In Carr's book [2, p.70-71] we find a good collection of properties of the continued fraction expansion of $\sqrt{N}$, which are summarized in the following, along with some properties taken from [4, 15].

1. Let c_n and r_n be the elements of two sequences of positive integers defined by the relation
$$\frac{\sqrt{N}+c_n}{r_n} = a_{n+1} + \frac{r_{n+1}}{\sqrt{N}+c_{n+1}}$$
with $c_0 = \lfloor\sqrt{N}\rfloor$, and $r_0 = N - a_0^2$; the elements of the sequence $a_1, a_2, \ldots, a_n \ldots$ are thus obtained as the integer parts of the left-side fraction, which is known as the complete quotient.
2. Let $a_0 = \lfloor\sqrt{N}\rfloor$ be initially computed, and set $c_0 = a_0$, $r_0 = N - a_0^2$, then sequences $\{c_n\}_{n\geqslant 0}$ and $\{r_n\}_{n\geqslant 0}$ are produced by the recursions
$$a_{m+1} = \left\lfloor \frac{a_0 + c_m}{r_m} \right\rfloor \ , \ c_{m+1} = a_{m+1} r_m - c_m \ , \ r_{m+1} = \frac{N - c_{m+1}^2}{r_m}. \qquad (2)$$
These recursions allow us to compute the sequence $\{a_m\}_{m\geqslant 1}$ using only rational arithmetical operations, and the iterations may be stopped when $a_m = 2a_0$, having completed a period.
3. If the period length τ is odd, set $\ell = \frac{\tau-1}{2}$; Legendre discovered and proved that the complete quotient $\frac{\sqrt{N}+c_\ell}{r_\ell}$ gives a representation of $N = c_\ell^2 + r_\ell^2$ as the sum of two squares.
4. Numerator A_n and denominator B_n of the n-th convergent to $\sqrt{N}$ can be recursively computed as $A_n = a_n A_{n-1} + A_{n-2}$ and $B_n = a_n B_{n-1} + B_{n-2}$, $n \geqslant 1$, respectively, with initial conditions $A_{-1} = 1$, $B_{-1} = 0$, $A_0 = a_0$, and $B_0 = 1$. The numerator A_m and the denominator B_m of any convergent are shown to be relatively prime by the relation $A_m B_{m-1} - A_{m-1} B_m = (-1)^{m-1}$ [4, p.85].
5. Using the sequences $\{A_m\}_{m\geqslant 0}$ and $\{B_m\}_{m\geqslant 0}$, two sequences
$$\boldsymbol{\Delta} = \{\Delta_m = A_m^2 - N B_m^2\}_{m\geqslant 0}, \quad \boldsymbol{\Omega} = \{\Omega_m = A_m A_{m-1} - N B_m B_{m-1}\}_{m\geqslant 1}$$
are introduced. It can easily be checked that $\Omega_m^2 - \Delta_m \Delta_{m-1} = N$, $\forall m \geqslant 1$. The elements of $\boldsymbol{\Delta}$ and $\boldsymbol{\Omega}$ satisfy a system of linear recurrences
$$\begin{cases} \Delta_{m+1} = a_{m+1}^2 \Delta_m + 2a_{m+1}\Omega_m + \Delta_{m-1} \\ \Omega_{m+1} = \Omega_m + a_{m+1}\Delta_m \end{cases} \qquad m \geqslant 1 \qquad (3)$$
with initial conditions
$$\Delta_0 = a_0^2 - N, \quad \Delta_1 = (1 + a_0 a_1)^2 - N a_1^2, \quad \Omega_1 = (1 + a_0 a_1) a_0 - N a_1.$$
By (3), it is immediate to see that $c_{m+1} = |\Omega_m|$ and $r_{m+1} = |\Delta_m|$.
6. The period of $\boldsymbol{\Delta}$ and $\boldsymbol{\Omega}$ is τ or 2τ, depending on whether τ is even or odd.

7. The sequence of ratios $\frac{A_n}{B_n}$ assumes the limit value $\sqrt{N}$ as n goes to infinity, due to the inequality $\left|\frac{A_n}{B_n} - \sqrt{N}\right| \leqslant \frac{1}{B_n B_{n+1}}$, since A_n and B_n go to infinity along with n. Since $\frac{A_n}{B_n} < \sqrt{N}$, if n is even, and $\frac{A_n}{B_n} > \sqrt{N}$, if n is odd [7], any convergent of even index is smaller than any convergent of odd index. This property implies that the terms of the sequence $\mathbf{\Delta}$ have alternating signs, with $\Delta_1 > 0$.
8. The value $c_0 = a_0$ is the greatest value that c_n may assume. No a_n or r_n can be greater than $2a_0$.
   If $r_n = 1$ then $a_{n+1} = a_0$. For all n greater than 0, we have

$$a_0 - c_n < r_n \leqslant 2a_0.$$

   The first complete quotient that is repeated is $\frac{\sqrt{N}+c_0}{r_0}$, and a_1, r_0, and c_0 commence each cycle of repeated terms.
9. Through the first period, we have the equalities $a_{\tau-j} = a_j$, $r_{\tau-j-2} = r_j$, and $c_{\tau-j-1} = c_j$.
10. The period τ has the tight upper bound $0.72\sqrt{N}\ln N$, $N > 7$, as was shown by Kraitchik [16, p.95]. However, the period length has irregular behavior as a function of N, because it may assume any value from 1, when $N = M^2 + 1$, to values close to the order $O(\sqrt{N}\ln N)$ [15].
11. Define the sequence of quadratic forms $\mathbf{f}_m(x, y) = \Delta_m x^2 + 2\Omega_m xy + \Delta_{m-1} y^2$, $m \geqslant 1$, which has the same period as $\mathbf{\Delta}$. Every $\mathbf{f}_m(x, y)$ is a reduced form of discriminant $4N$. Within the first block, all quadratic forms $\mathbf{f}_m(x, y)$, $1 \leqslant m \leqslant \tau$ are distinct, and constitute the principal class $\mathbf{\Gamma}(\mathbf{f})$ of reduced forms, with the ordering of the elements inherited from $\mathbf{\Delta}$. The definition of reduced form used here is slightly different from the classic one: set $\kappa = \min\{|\Delta_m|, |\Delta_{m-1}|\}$; it is easily checked that Ω_m is the sole integer such that

$$\sqrt{N} - |\Omega_m| < \kappa < \sqrt{N} + |\Omega_m|,$$

   with the sign of Ω_m chosen opposite to the sign of Δ_m. Since the sign of Δ_{m-1} is the same as that of Ω_m, which is opposite to that of Δ_m, in $\mathbf{\Gamma}(f)$ the two triples of signs (signatures) $(-,+,+)$ and $(+,-,-)$ alternate.

The following theorems are taken, without proof, from [5].

Theorem 1. *Starting with $m = 1$, the sequences $\mathbf{\Delta} = \{\Delta_m\}_{m\geqslant 0}$ and $\mathbf{\Omega} = \{\Omega_m\}_{m\geqslant 0}$ are periodic with the same period τ or 2τ depending on whether τ is even or odd. The elements of the blocks $\{\Delta_m\}_{m=0}^{\tau}$ and $\{\Omega_m\}_{m=1}^{\tau}$ satisfy the symmetry relations $\Delta_m = (-1)^{\tau}\Delta_{\tau-m-2}$, $\forall\ m \leqslant \tau - 3$ and $\Omega_{\tau-m-1} = (-1)^{\tau+1}\Omega_m$, $\forall\ m \leqslant \tau - 2$, respectively.*

If τ is odd, the ordered set $\{\Delta_m\}_{m=1}^{\tau}$ has a central term of index $\ell = \frac{\tau-1}{2}$, with $\Delta_\ell = -\Delta_{\ell-1}$ since $\tau - \ell - 2 = \ell - 1$, and the equation $\Omega_\ell^2 - \Delta_\ell\Delta_{\ell-1} = N$ gives

a solution of the Diophantine equation $x^2 + y^2 = N$ with $x = \Delta_\ell$ and $y = \Omega_\ell$, the situation first recognized by Legendre.
If τ is even, the ordered set $\{\Delta_m\}_{m=1}^{\tau}$ has no central term; in this case, with $\ell = \frac{\tau-2}{2}$ we have $\Omega_{\ell+1} = -\Omega_\ell$ and $\Delta_{\ell+1} = \Delta_\ell$, hence $\mathbf{f}_{\ell+1}(x, y) = \mathbf{f}_\ell(y, -x)$.

Theorem 2. *Let the period τ of the continued fraction expansion of $\sqrt{N}$ be even; we have $\Omega_{\tau-1} = -a_0$, $\Delta_\tau = \Delta_{\tau-2}$, and $\Omega_\tau = -\Omega_{\tau-1}$. Defining the integer $\gamma \in O_{\mathbb{Q}(\sqrt{N})}$ by the product*

$$\gamma = \prod_{m=1}^{\tau} \left(\sqrt{N} + (-1)^m \Omega_m\right) \quad ,$$

let σ denote the Galois automorphism of $\mathbb{Q}(\sqrt{N})$ (i.e. $\sigma(\sqrt{N}) = -\sqrt{N}$), then

$$\frac{\gamma}{\sigma(\gamma)} = A_{\tau-1} + B_{\tau-1}\sqrt{N}$$

is a positive fundamental unit (or the cube of the fundamental unit) of $\mathbb{Q}(\sqrt{N})$.

Based on this theorem, we say that the unit $c_{\tau-1} = A_{\tau-1} + B_{\tau-1}\sqrt{N}$ in $\mathbb{Q}(\sqrt{N})$ splits N, if $N_1 = \gcd\{A_{\tau-1} - 1, N\}$ is neither 1 nor N. Then we have the proper factorization $N = N_1 N_2$. Further, using the following involutory matrix, [5], whose square is $(-1)^\tau I_2$

$$M_{\tau-1} = \begin{bmatrix} -A_{\tau-1} & NB_{\tau-1} \\ -B_{\tau-1} & A_{\tau-1} \end{bmatrix} \quad ,$$

it is shown that

$$A_{\tau-m-2} = (-1)^{m-1}(A_{\tau-1}A_m - NB_{\tau-1}B_m) \quad 1 \leqslant m \leqslant \tau - 2 \quad . \tag{4}$$

As an immediate consequence of this equation, if the unit $c_{\tau-1}$ splits N, then any pair $(A_m, A_{\tau-m-2})$ splits N, since taking $A_{\tau-m-2}$ modulo N we have

$$A_{\tau-m-2} = (-1)^{m-1} A_m A_{\tau-1} \bmod N,$$

thus $A_{\tau-m-2}$ is certainly different from A_m, because $A_{\tau-1} \neq \pm 1 \bmod N$.

Theorem 3. *If the period τ of the continued fraction expansion of $\sqrt{N}$ is even, the element $c_{\tau-1}$ in $\mathbb{Q}(\sqrt{N})$ splits $4N$, and a factor of $4N$ is located at positions $\frac{\tau-2}{2} + j\tau$, $j = 0, 1, \ldots$, in the sequence $\mathbf{\Delta} = \{c_m \sigma(c_m)\}_{m \geqslant 1}$.*

Factorization

Gauss recognized that the factoring problem was to be important, although very difficult,

> *... Problema, numeros primos a compositis dignoscendi, hosque in factores suos primos resolvendi, ad gravissima ac utilissima totius arithmeticae pertinere, et geometrarum tum veterum tum recentiorum industriam ac sagacitatem occupavisse, tam notum est, ut de hac re copiose loqui superfluum foret. ...* C. F. Gauss [*Disquisitiones Arithmeticae* Art. 329]

In spite of much effort, various different approaches, and the increased importance stemming from the large number of cryptographic applications, no satisfactorily factoring method has yet been found. However, approaches to factoring based on continued fractions have lead to some of the most efficient factoring algorithms. In the following, a new variant of Shanks' infrastructural method [14] is described which exploits the property of the block $\mathbf{\Delta}_1 = \{\Delta_m\}_{m=1}^{\tau}$, which is made more precise in the following theorem taken without proof from [5].

Theorem 4. *Let N be a positive square-free integer. If the norm of the positive fundamental unit $u \in \mathbb{Q}(\sqrt{N})$ is 1, and some factor of N is a square of a principal integral ideal in $\mathbb{Q}(\sqrt{N})$, then u is split for N. A proper factor of N is found in position $\frac{\tau-2}{2}$ of $\mathbf{\Delta}_1$.*

It should be noted that $\mathbf{\Delta}_1$ offers several different ways for factoring a composite number N:

1. If τ is even and 2 is not a quadratic residue modulo N, then in position $\frac{\tau-2}{2}$ of the sequence $\mathbf{\Delta}_1$ we find a factor of N.
2. If τ is odd, then by Legendre's results we find a representation $N = X^2 + Y^2$, which implies that $s_1 = \frac{X}{Y} \bmod N$ is a square root of -1. If we are able to find another square root s_2 different from $-\frac{X}{Y} \bmod N$ (we have four different square roots of a quadratic residue modulo $N = pq$), then the difference $s_1 - s_2$ contains a proper factor of N.
3. If some square d_o^2 is found in the sequence $\mathbf{\Delta}_1$, it implies $A_m^2 - NB_m^2 = d_o^2$, thus there is a chance that some proper factor of N divides $(A_m - d_o)$ or $(A_m + d_o)$. The number of squares in $\mathbf{\Delta}_1$ is $O(\sqrt{\tau})$, and about $(\frac{1}{2})$ of these squares factor N. This method was introduced by NTC:Shanks72.
4. If equal terms $\Delta_m = \Delta_n, \quad m \neq n$ occur in $\mathbf{\Delta}_1$, with $m, n < \frac{\tau}{2}$, then

$$A_m^2 - A_n^2 = 0 \bmod N$$

allows us to find two factors of N by computing $\gcd\{A_m - A_n, N\}$ and $\gcd\{A_m + A_n, N\}$. This is an implementation of an old idea of Fermat.

Computational issues

By Theorem 4 we know that a factor of N is $\Delta_{\frac{\tau-2}{2}}$, which can be directly computed from the continued fraction of $\sqrt{N}$ in $\frac{\tau-2}{2}$ steps. Unfortunately, this number is usually prohibitively large. However, if τ is known, using the baby-step/giant-step artifice, the number of steps can be reduced to the order $O(\log_2 \tau)$. To this end, we can move through the principal class $\mathbf{\Gamma}(\mathbf{f})$, of ordered quadratic forms $\mathbf{f}_m(x, y)$, by introducing a notion of distance between pairs of quadratic forms compliant with Gauss' quadratic form composition. The distance between two adjacent quadratic forms $\mathbf{f}_{m+1}(x, y), \mathbf{f}_m(x, y) \in \mathbf{\Gamma}(\mathbf{f})$ is defined as

$$d(\mathbf{f}_{m+1}, \mathbf{f}_m) = \frac{1}{2} \ln \left(\frac{\sqrt{N} + (-1)^m \Omega_m}{\sqrt{N} - (-1)^m \Omega_m} \right) \quad , \tag{5}$$

and the distance between two quadratic forms $\mathbf{f}_m(x, y)$ and $\mathbf{f}_n(x, y)$, with $m > n$, is defined as the sum $d(\mathbf{f}_m, \mathbf{f}_n) = \sum_{j=n}^{m-1} d(\mathbf{f}_{j+1}, \mathbf{f}_j)$. The distance of $\mathbf{f}_m(x, y)$ from the beginning of $\mathbf{\Gamma}(\mathbf{f})$ is defined referring to a properly-chosen quadratic form

$$\mathbf{f}_0 = \Delta_0 x^2 - 2\sqrt{N - \Delta_0} xy + y^2$$

hypotetically located before $\mathbf{f}_1$. Thus we have

$$d(\mathbf{f}_m, \mathbf{f}_0) = \sum_{j=0}^{m-1} d(\mathbf{f}_{j+1}, \mathbf{f}_j)$$

if $m \leqslant \tau$. The notion is also extended to index $k\tau \leqslant m < (k+1)\tau$ by setting $d(\mathbf{f}_m, \mathbf{f}_0) = d(\mathbf{f}_{m \bmod \tau}, \mathbf{f}_0) + kR_{\mathbb{F}}$. The distance $d(\mathbf{f}_\tau, \mathbf{f}_0)$ is exactly equal to $R^* = \ln c_{\tau-1}$, which is the regulator $R_{\mathbb{F}}$, or three times $R_{\mathbb{F}}$, and the distance $d(\mathbf{f}_{\frac{\tau}{2}}, \mathbf{f}_0)$ is exactly equal to $\frac{R^*}{2}$, see [5] for a straightforward proof. Now, a celebrated formula of Dirichlet's gives the product

$$h_{\mathbb{F}} R_{\mathbb{F}} = \frac{\sqrt{D}}{2} L(1, \chi) = - \sum_{n=1}^{\lfloor \frac{D-1}{2} \rfloor} \left(\frac{D}{n} \right) \ln \left(\sin \frac{n\pi}{D} \right) \tag{6}$$

where $h_{\mathbb{F}}$ is the class field number, $L(1, \chi)$ is a Dedekind L-function, $D = N$ if $N \equiv 1 \bmod 4$ or $D = 4N$ otherwise, and character χ is the Jacobi symbol in this case. If we know $h_{\mathbb{F}}$ exactly, we know R^* exactly and we can proceed to factorization, with complexity $O((\log_2 N)^4)$ [9], conditioned on the computation of $L(1, \chi)$. The Dirichlet $L(1, \chi_N)$ function can be efficiently evaluated using the following expression for the product $h_{\mathbb{F}} R_{\mathbb{F}}$ as a function of N

$$h_{\mathbb{F}} R_{\mathbb{F}} = \frac{1}{2} \sum_{x \geqslant 1} \left(\frac{D}{x} \right) \left(\frac{\sqrt{D}}{x} \operatorname{erfc} \left(x \sqrt{\frac{\pi}{D}} \right) + E_1 \left(\frac{\pi x^2}{D} \right) \right) \quad . \tag{7}$$

where both the complementary error function $\mathrm{erfc}(x)$ and the exponential integral function $E_1(x)$ can be quickly evaluated. Once we know R^*, with the NTC:Shanks72' infrastructural method [14] or some of its improvements [1, 3, 12], we can find $\mathbf{f}_{\frac{\tau-2}{2}}(x, y)$, thus a factor of N. The goal is to obtain $\mathbf{f}_{\frac{\tau-2}{2}}(x, y)$ with as few steps as possible. To this end we can perform 1) giant-steps within $\mathbf{\Gamma}(\mathbf{f})$ which are realized by the Gauss composition law of quadratic forms, followed by a reduction of this form to $\mathbf{\Gamma}(\mathbf{f})$, and 2) baby-steps moving from one quadratic form to the next in $\mathbf{\Gamma}(\mathbf{f})$. Two operators ρ^+ and ρ^- are further defined [3, p.259] to allow small (baby) steps, precisely

- ρ^+ transforms $\mathbf{f}_m(x, y)$ into $\mathbf{f}_{m+1}(x, y)$ in $\mathbf{\Gamma}(\mathbf{f})$, and is defined as

$$\rho^+([a, 2b, c]) = [\frac{b_1^2 - N}{a}, 2b_1, a],$$

  where b_1 is $2b_1 = [2b \bmod (2a)] + 2ka$ with k chosen in such a way that $-|a| < b_1 < |a|$.
- ρ^- transforms $\mathbf{f}_m(x, y)$ into $\mathbf{f}_{m-1}(x, y)$ in $\mathbf{\Gamma}(\mathbf{f})$ and is defined as

$$\rho^-([a, 2b, c]) = [c, 2b_1, \frac{b_1^2 - N}{c}],$$

  where b_1 is $2b_1 = [-2b \bmod (2c)] + 2kc$ with k chosen in such a way that $-|c| < b_1 < |c|$.

The composed form $\mathbf{f}_m \bullet \mathbf{f}_n$ has the distance $d(\mathbf{f}_m \bullet \mathbf{f}_n, \mathbf{f}_0) \approx d(\mathbf{f}_m, \mathbf{f}_0) + d(\mathbf{f}_n, \mathbf{f}_0)$.

1. By the law $\bullet$, $\mathbf{\Gamma}(\mathbf{f})$ resembles a cyclic group, with $\mathbf{f}_{\tau-1}$ playing the role of identity.
2. Since in $\mathbf{\Gamma}(f)$ the two triples of signs (signatures) $(-, +, +)$ and $(+, -, -)$ alternate, the composed form $\mathbf{f}_m(x, y) \bullet \mathbf{f}_n(x, y)$ must have one of these signatures.
3. The composition of a quadratic form with itself is called *doubling* and denoted $2_\bullet\mathbf{f}_n$, thus s iterated doublings are written as $2^s_\bullet\mathbf{f}_n(x, y)$. The distance is nearly maintained by the composition $\bullet$ (giant-steps). The error affecting this distance estimation is of order $O(\ln N)$ as shown by Schoof in [12], and is rigorously maintained by the one-step moves $\rho^\pm$ (baby-steps).

An outline of the procedure is the following, assuming that R^* is preliminarily computed:

1. Let ℓ be a small integer. Compute an initial quadratic form $\mathbf{f}_\ell = [\Delta_\ell, 2\Omega_\ell, +\Delta_{\ell-1}]$ and its distance $d_\ell = d(\mathbf{f}_\ell, \mathbf{f}_0)$ from the continued fraction expansion of $\sqrt{N}$ stopped at term $\ell + 1$.
2. Compute $j_t = \lceil \log_2 \frac{R^*}{d_\ell} \rceil$.

3. Starting with $[\mathbf{f}_\ell, d_\ell]$, iteratively compute and store in a vector $\mathcal{F}_{j_t}$ the sequence $[2^j_\bullet \mathbf{f}_\ell, 2^j d_\ell]$ up to j_t. The middle term (i.e. $\mathbf{f}_{\frac{\tau-2}{2}}$) of $\mathbf{\Gamma}(\mathbf{f})$ is located between the terms $2^{j_t-1}_\bullet \mathbf{f}_\ell$ and $2^{j_t}_\bullet \mathbf{f}_\ell$.
4. The middle term of $\mathbf{\Gamma}(\mathbf{f})$ can be quickly reached using the elements of $\mathcal{F}_{j_t}$, starting by computing $\mathbf{f}_r = (2^{j_t-1}_\bullet \mathbf{f}_\ell) \bullet (2^{j_t-2}_\bullet \mathbf{f}_\ell)$ and checking whether $2^{j_t-1} d_\ell + 2^{j_t-2} d_\ell$ is greater or smaller than $\frac{R^*}{2}$; in the first case set $\mathbf{f}_s = \mathbf{f}_r$, otherwise set $\mathbf{f}_s = 2^{j_t-1}_\bullet \mathbf{f}_\ell$. Iterate this composition by computing $\mathbf{f}_r = \mathbf{f}_s \bullet (2^i_\bullet \mathbf{f}_\ell)$ and setting $\mathbf{f}_s = \mathbf{f}_r$ for decreasing i up to 0, and let the final term be $[\mathbf{f}_s, d_s]$.
5. Iterate the operation $\rho^{\pm}$ a convenient number $O(\ln N)$ of times, until a factor of $4N$ is found.

Conclusions

An iterative algorithm has been described which produces a factor of a composite square-free N with $O((\ln(N))^4)$ iterations at most, if hR is exactly known, h being the class number, and R the regulator of $\mathbb{Q}(\sqrt{N})$. The bound $O((\ln(N))^4)$ is computed by multiplying the number of giant-steps, which is $O(\ln(N))$, by the number of steps at each reduction, completing a giant-step, which is upper bounded by $O((\ln(N))^3)$ as shown in [9, 13]. It is remarked that, in this bound computation, the cost of the arithmetics in $\mathbb{Z}$, i.e. multiplications and additions of big integers, is not counted [9]. Furthermore, it is not difficult to modify the algorithm to use a rough approximation of hR; the computations become cumbersome, but asymptotically the algorithm is polynomial, because a sufficient approximation of hR is easily obtained by computing the series in equation (7) truncated at a number of terms $O(\ln(N))$, since the series converges exponentially [3, Proposition 5.6.11, p.262-263]. It remains to ascertain whether this asymptotically-good factoring algorithm is also practically better than any sub-optimal probabilistic factoring algorithm.

References

1. J. Buchmann and H. Williams. On the infrastructure of the principal ideal class of an algebraic number field of unit rank one. *Mathematics of Computation*, 50(182):569–579, 1988.
2. G. S. Carr. *Formulas and Theorems in Mathematics*. Chelsea, New York, 1970.
3. H. Cohen. *A Course in Computational Algebraic Number Theory*. Springer, New York, 1993.
4. H. Davenport. *The Higher Arithmetic: An Introduction to the Theory of Numbers*. Dover, New York, 1983.
5. M. Elia. Continued fractions and factoring. *Rendiconti Sem. Mat., Univ. Pol. Torino*, 78(1):83–101, 2020.
6. C. F. Gauss. *Disquisitiones Arithmeticae*. Springer, New York, 1986. Originally published in 1801.
7. G. H. Hardy and E. M. Wright. *An Introduction to the Theory of Numbers*. Oxford University Press, Oxford, 1971.
8. E. Jacobsthal. Über die Darstellung der Primzahlen der Form 4n+1 als Summe zweier Quadrate. *Journal für die reine und angewandte Mathematik*, 2:238–245, 1907.
9. J. C. Lagarias. Worst-case complexity bounds for algorithms in the theory of integral quadratic forms. *J. Algorithms*, 1(2):142–186, 1980.
10. A.-M. Legendre. *Essai sur la Théorie des Nombres*. Cambridge University Press, 2009. Originally published by Chez Courcier in 1808.
11. G. B. Mathews. *Theory of Numbers*. Chelsea, New York, 1892.
12. R. Schoof. Quadratic fields and factorization. In *Computational methods in number theory*, volume 154-155 of *Mathematical Centre Tracts*, pages 235–286. Mathematisch Centrum, Amsterdam, 1982.
13. A. Schönhage. Fast reduction and composition of binary quadratic forms. In *International symposium on Symbolic and algebraic computation 1961*, pages 128–133, New York, 1961.
14. D. Shanks. The infrastructure of a real quadratic field and its applications. In *1972 Number Theory Conference*, pages 217–224, Boulder, 1972.
15. W. Sierpinski. *Elementary Theory of Numbers*. North Holland, New York, 1988.
16. J. Steuding. *Diophantine Analysis*. Chapman & Hall, New York, 2003.
17. D. Zagier. A one-sentence proof that every prime $p \equiv 1 (\mathrm{mod}\ 4)$ is a sum of two squares. *American Mathematical Monthly*, 97(2):144, 1990.

Ringraziamenti

Vogliamo ringraziare tutti i volontari e collaboratori di De Cifris, fondamentali per la pubblicazione di questo Volume.
In particolare, ringraziamo il Tesoriere dell'associazione, Elisa Cermignani, nonché i volontari Elena Broggini, Leonardo Errati, Naima Noukti.