International Journal of Advanced Research in Engineering Technology & Science ISSN: 2349-2819
Email: editor@ijarets.org July- 2015 Volume 2, Issue-7 www.ijarets.org

COMPUTATIONAL NUMBER THEORY AND CRYPTOGRAPHY

Anupam Tandon

Assistant Professor, Mathematics, SIRDA Engineering College

Abstract

Computational Number Theory is an essential component of contemporary cryptography, contributing to the establishment of a mathematical basis for the security of data and the transmission of encrypted communications. This topic investigates the intricacy of number-theoretic issues, which are essential for cryptographic algorithms. Some examples of these difficulties include prime factorisation, discrete logarithms, and elliptic curve arithmetic. Cryptography, on the other hand, makes use of these issues in order to construct safe encryption, digital signatures, and systems for important information sharing. RSA, Diffie-Hellman, and ECC are examples of classical cryptographic methods that rely on number-theoretic hardness assumptions. However, recent developments in quantum computing have presented researchers with new hurdles, which has led to the development of post-quantum cryptography algorithms. In this study, the interaction between computational number theory and cryptography is brought to light. Key methodologies, algorithmic developments, and security implications in the ever-changing digital ecosystem are discussed.

Keywords: *Computational, Cryptography, Theory*

Introduction

Computational Number Theory is a subfield of mathematics that focusses on the study of algorithms that are associated with numbers, namely those that pertain to prime numbers, modular arithmetic, and integer factorisation. This set of ideas serves as the foundation for contemporary cryptography, which is responsible for ensuring the safety of communication within digital systems. In order to construct encryption techniques, digital signatures, and authentication procedures that secure sensitive information from being accessed by adversaries, cryptography is dependent on the difficulty of particular number-theoretic problems. RSA, Diffie-Hellman, and Elliptic Curve Cryptography (ECC) are examples of algorithms that provide solid methods for secure communication. The introduction of public-key cryptography was a game-changer in terms of data security. These techniques are dependent on issues such as integer factorisation and discrete logarithms, which are computationally impossible to answer within an acceptable amount of time using traditional computers. However, as a result of the advent of quantum computing, standard cryptographic techniques are now exposed to the possibility of being compromised. As a result, it is necessary to build post-quantum cryptography solutions that are founded on lattice problems and hash functions. This article investigates the connection between computational number theory and cryptography, focussing on important algorithms, the issues that are faced in terms of security, and the potential future routes that the discipline will take. For the purpose of building cryptographic systems that are able to withstand changing computing capabilities and cyber threats, it is essential to have a solid understanding of these concepts.

Related Work

The purpose of this survey study is to provide an introduction to the use of number theory principles in cryptography. Some of the topics that are discussed in this book include prime numbers, modular arithmetic, the Euclidean method, the Chinese Remainder Theorem, Euler's Theorem, the RSA technique, and the discrete logarithm problem. The purpose of this overview article is to give a comprehensive introduction to public key cryptography, with a particular focus on the mathematical foundations of this discipline. Identity-based cryptography, RSA, Diffie-Hellman, ElGamal, elliptic curve encryption, digital signatures, and key exchange protocols are some of the topics that are covered in this book. For the purpose of this in-depth article, the topic of discussion is lattice-based cryptography, which provides defence against attacks from both classical and quantum computers. The principles of lattices, lattice-based key exchange protocols, lattice-based encryption algorithms (such as Learning With Errors), and lattice-based signature systems are all covered in this book. The purpose of this in-depth research project is to investigate post-quantum cryptography, which aims to develop cryptographic algorithms that are resistant to assaults from quantum computers. Isogeny-based, lattice-based, code-based, and hash-based cryptography systems are some of the families of post-quantum encryption systems that are included in this category. The purpose of this review research is to provide a snapshot of the most recent advancements in cryptanalysis techniques. The topics that are covered in this course include, amongst others, side-channel attacks, fault attacks, collision attacks, linear cryptanalysis, algebraic cryptanalysis, and differential cryptanalysis on cryptographic hash functions.

Fibonacci Sequence in Cryptography

The Fibonacci sequence, which consists of the numbers 0, 1, 1, 2, 3, 5, 8, and 13, has been researched to see whether or not it may be utilised in the field of cryptography. It is the sum of the two numbers that came before it that determines each number in the series. A cryptosystem that is constructed on the Fibonacci sequence might provide an exciting alternative in some circumstances, despite the fact that it might not be used as frequently as other cryptographic approaches. Here are some specifics on a fundamental use of a cryptosystem that is based on the Fibonacci sequence:

Generating a Key:

1. In the beginning of the cryptosystem, two starting parameters are selected to act as the secret key. These parameters are typically two consecutive Fibonacci numbers. For example, if n is a high number that appears to be appropriate, we have the option of selecting F(n) and F(n+1) as the initial parameters.
2. Utilising the Fibonacci Sequence as a Calculation Tool To construct a Fibonacci sequence, begin with the beginning parameters and continue doing so until the series reaches the necessary desired length. It is possible to derive the subsequent number by repeatedly adding both of the numbers that comprise the sequence's last position. It is anticipated that the sequence that was created would be utilised for both encryption and decryption.

Encryption:

1. Message Representation: Transform the plaintext message into numerical form by employing a method that is appropriate, such as mapping to numerical values or ASCII representation.
2. Process of Encryption: Create a numerical representation of the message by encrypting it with the Fibonacci sequence. Iterate through the Fibonacci sequence for each numerical value in the message, and for each Fibonacci number that you come across, add it to the value that was initially given.
3. Ciphertext generation: The altered numerical values that are the outcome of the process are what make up the ciphertext.

$$C = P * F(i) \qquad (1)$$

Decryption:

1. Retrieve the numerical ciphertext values after processing the ciphertext.
2. In order to decipher the numerical ciphertext, the Fibonacci sequence is an essential tool. Carry on with the process of iterating through the Fibonacci sequence, removing each Fibonacci number from the numerical value as you proceed.
3. In order to get the message, obtain the numerical numbers that were produced and reconstruct them into the plaintext form in which they were initially stored.

The addition of the two words that come before it results in the creation of the well-known Fibonacci sequence, which is a series that starts with 0 and 1. An updated version of the Fibonacci sequence is presented in the following paragraphs:

{0, 1, 1, 2, 3, 5, 8, 13, ...}

The result of the two terms that came before it is the meaning of each phrase in this series. For instance, the result of adding 1 and 1 is 2, the result of adding 1 and 2 is 3, the result of adding 2 and 3 is 5, and so on. The product of the two words that came before it is the product of each term in the series, and the sequence never comes to a conclusion. The Fibonacci sequence is defined as explained in the following paragraphs:

For all non-negative integers n, F(0) = 0, F(1) = 1, F(n+2) = F(n) + F(n+1), and so forth etc. This definition states that the series starts with F(0) = 0 and F(1) = 1 at the beginning. The following term, F(n+2), is created by adding together the two words that come before it, which are denoted by the letters F(n) and F(n+1). This recursive formula results in the Fibonacci sequence, in which each term is equal to the sum of the two terms that came before it.

Property: F(n+2)= F(0)+ F(1)+ F(2)+ … + F(n)+ F(n+1) (1)

As the basis for Binet's explicit formula for the Fibonacci sequence, which was devised in 1843, the Golden ratio, which is represented by the Greek symbol phi, served as the foundation. An example of an expression for the formula is shown below:

$$F(n)=((\varphi^n- (1-\varphi)^n))/\sqrt{5} \qquad (2)$$

Proposition:

Let α and β be the roots of the equation x2 = x + 1, where and are determined as follows:

$$\alpha = (1 + \sqrt{5}) / 2 \qquad (3)$$
$$\beta = (1 - \sqrt{5}) / 2 \qquad (4)$$

The Fibonacci sequence of numbers may thus be stated using and in the following manner:

$$F(n) = (\alpha^n - \beta^n) / \sqrt{5} \qquad (5)$$

Within the context of this example, the nth Fibonacci number is denoted by the symbol F(n), and the roots of the equation x2 = x + 1 are represented by the letters and. It is possible to finish the formula by first increasing and to the power of n, then subtracting the results, and then dividing by the square root of 5. Through the utilisation of this approach, it is possible to instantaneously compute the value of any Fibonacci number without the requirement of doing computations that are iterative or recursive.

Leema: In point of fact, the Fibonacci sequence is not a series that is extremely growing. Specifically, any subsequence that is consecutive and iterative {Fm, Fm+1, ..., Fm+r} where r > 2 and m > 0 is not a super increasing sequence.

Proof: It is possible to demonstrate that the Fibonacci sequence is not a super growing sequence by employing a counterexample. This is something that may be performed. Let us consider the initial few terms of the series, which are as follows: 0, 1, 1, 2, 3, 5,... The fact that there is less than the sum of the phrases that came before it, which is equal to zero plus one plus one plus two, is manifestly obvious from the observation. Put another way, F4 is equal to F0 plus F1 plus F2 plus F3.

Examining the subsequence "Fm, Fm+1,..., Fm+r" for m > 0 and r > Next up is the number two. We are able to do a speedy check in order to demonstrate that the term sum of the subsequence does not beyond the subsequent Fibonacci number. In particular, we have it that

$$Fm + Fm+1 + ...+ Fm+r < Fm+2 \qquad (6)$$

It can be further simplified as

$$Fm+r < Fm+2 - (Fm + Fm+1) \qquad (7)$$

Since the Fibonacci sequence is recursive, Fm+2 - (Fm + Fm+1) It is possible to express it as Fm+1. On account of this, we have:

$$Fm+r < Fm+1 \qquad (8)$$

The purpose of this counterexample is to demonstrate that the total of each term in the subsequence "Fm, Fm+1,..., Fm+r" does not exceed the following Fibonacci number, Fm+1, for any m that is more than zero and r that is greater than two. Furthermore, as a consequence of this, the sequence of Fibonacci numbers does not satisfy the super growing property. According to the criteria for a super growing series, the Fibonacci sequence does not meet the requirements. This is despite the fact that it possesses distinctive

properties and characteristics of its own. It is possible to draw parallels between the Fibonacci sequence and the Lucas sequence, which is denoted by the symbol Ln0. However, the Lucas series starts with the numbers 2 and 1. Following the first two numbers, it repeats itself in accordance with the same laws that govern the Fibonacci range of numbers. By definition, L0 is equal to two, while L1 is equal to one. Furthermore, for n greater than zero, Ln+2 is equal to Ln plus Ln+1.

The Lucas sequence can therefore be illustrated as follows:

{2, 1, 3, 4, 7, 11, ...}

It is important to keep in mind that the Lucas series, much like the Fibonacci sequence, is not a sequence that follows a super increasing pattern. In order to be considered a super growing sequence, each term must be bigger than the sum of all the terms that came before it, as was described earlier. It is important to note that the Lucas sequence does not fulfil this condition.

$$Ln = F(n-1) + F(n+1) \qquad (10)$$

Ln is the symbol that is used to indicate the nth term of the Lucas sequence, while F(n) is the symbol that is used to represent the nth term of the Fibonacci sequence. Both the Lucas series and the Fibonacci sequence may be directly connected to one another via the use of this approach, which allows for the computation of the Lucas numbers based on the Fibonacci numbers that correspond to them.

Cryptosystem Methodology

The Knapsack Cryptosystem and the concept of superincreasing sequences served as the foundation for the development of two other cryptosystems of our own:

Using a pre-shared key cryptosystem

- In order for Alice and Bob to be able to communicate or interact with one another inside this cryptosystem, they must both possess a pre-shared key that is in the form of a sequence that is growing in complexity.
- The common key for encryption and decoding is a superincreasing sequence.
- This key can be used by Alice and Bob to securely encrypt and decrypt messages sent back and forth.

Cryptosystem for exchanging keys:

- Alice and Bob are able to generate a shared key using this cryptography while they are still in contact with one another.
- They are not need to have a pre-shared key at first.
- In a protocol for key exchange, Alice and Bob each exchange information with one another in order to build a single superincreasing sequence that will serve as their shared key.

During their communication, this shared key might be used for later encryption and decryption procedures.

Secret Key provide:

The superincreasing sequence vector r = (n, 2n,..., rn) in the provided cryptosystem scenario represents the shared key that Alice and Bob share. The values g Z+ and p, a huge prime number satisfying p > 2rn, are included in the public parameters. gcd(a, p - 1) = 1 for Alice's secret key a Z+, and gcd(k, p - 1) = 1 for Bob's secret key k Z+.

When it comes to this particular cryptosystem, the techniques for encryption and decryption are as follows:

1. Alice calculates A = ga mod p and sends Bob the result, A.

2. Bob wants to encrypt a plaintext message with the following format: x Zq (where q is a modulus that represents the plaintext space in an acceptable manner). Two components of the cypher text are produced by him:

C1 = p mod gk
C2 = Ak(x r) mod p,

represents the process of doing a dot product between two vectors, where stands for the action.

3. Bob gives the ciphertext parts (C1, C2) to Alice, who then begins the decryption process.

Using her secret key a, she calculates C' = (C1)-a C2 mod p. Alice is then able to get the original plaintext message x by utilising the super rising sequence r and the value that she has got from it in order to solve the Knapsack Problem.

Proof:

$$(ci)^{(-a)(c2)\bmod}p = \left(g^{ka}\right)^{a^{-1}} * \left(g^{akx\cdot r}\right)\bmod p = g^{kx\cdot r}\bmod p = x\cdot f \bmod p \qquad (11)$$

The equation xf mod p may be transformed into a Knapsack problem by employing the vector (x • r) and the super growing sequence r. This is possible due to the fact that p is greater than 2rn. Given this, we are in a position to make use of Proposition 2.2, which may be utilised in order to determine x.

Example:

We might reconstruct the process in the following manner, taking into account the information that was supplied and the steps that were stated in the scenario: The value of A is sent to Bob after Alice has computed it using the formula A = ga mod p = 997 mod 1223 = 856. Bob has the plaintext version of the message x = (0, 0, 1, 1, 0, 0, 1, 0) in his possession. The results of his computation of (C1, C2) are as follows:

C1 = g^k mod p = 99^3 mod 1223 = 460
C2 = A^(x* f) mod p = 856^(x* f) mod 1223 = 45

The number (460, 45) is given to Alice by Bob. C' = (C1)(-a) * 45 mod 1223 = 233 * C2 mod p: C' = 460(-7) This mathematical expression was derived by Alice. * Alice then solves the Knapsack Problem for (r, 233) in order to extract the plaintext message x = (0, 0, 1, 1, 0, 0, 1, 0). This allows her to recover the message successfully. Because of this, Alice is able to successfully decrypt the ciphertext and acquire

the plaintext that was originally intended. A common super growing sequence is already in existence between Alice and Bob, which is the foundation upon which the aforementioned cryptosystem is built. Due to the fact that this is a rather significant assumption, we have developed an additional cryptosystem that is analogous to this one. This cryptosystem generates a common superincreasing sequence that is based on the Fibonacci subsequence.

Conclusion

current cryptography is supported by computational number theory, which provides the mathematical framework for secure communication, data integrity, and authentication. This theory serves as the backbone of current cryptography. Cryptographic algorithms that are frequently used include RSA, Diffie-Hellman, and Elliptic Curve Cryptography (ECC). These methods are based on fundamental difficulties such as integer factorisation, discrete logarithms, and elliptic curve arithmetic. Transactions conducted online, digital signatures, and conversations that are kept secret have all benefited greatly from the use of these approaches. Traditional cryptographic systems, on the other hand, are facing substantial hurdles as a result of the fast developments in computational powers, particularly with the introduction of quantum computing. Considering that quantum algorithms such as Shor's algorithm pose a risk of breaking public-key encryption systems that are extensively utilised, it is imperative that post-quantum cryptographic solutions be developed as quickly as possible. A variety of alternative methods, including hash-based encryption, lattice-based cryptography, and others, are now being investigated in order to guarantee security in the post-quantum period. Cryptography will continue to rely heavily on computational number theory, which will continue to play an important role as the digital world continues to grow. Ongoing research and innovation in this area will be the driving force behind the creation of cryptographic systems that are more secure, efficient, and durable. These systems will protect sensitive data from emerging cyber threats. To ensure effective protection for digital communications in a world that is becoming increasingly linked, the future of cryptography will continue to advance by solving both classical and quantum security difficulties by addressing both types of security challenges.

References

[1] Hoffstein, J., Pipher, J., Silverman, J.H.. “An Introduction to M athem atical Cryptography,” Springer. 2010.
[2] Koblitz, Neal. “A Course in Number Theory and Cryptography,” Springer-Verlag, 1987.
[3] Luma, A., Raufi, B. “Relationship between Fibonacci and Lucas Sequences and Their Application in Symmetric Cryptosystem s,” Latest Trends on Circuits, Systems and Signals, 2010.
[4] Matousesk, Radomil. “Knapsack Cipher and Cryptanalyst Using Heuristic M ethods,” Institute of Autom ation and Com puter Science, Brno University of Technology,_
[5] Menezes, A., Vanstone, S., “Elliptic Curve Cryptosystems and Their Im plem entation,” Journal of Cryptology, 1993.
[6] Paterson, Kenneth G. “Cryptography from Pairings: A Snapshot of Current Research,” Information Security Group, University of London. November, 2002.
[7] Raphael, A. Joseph, Sundaram, Dr. V., “Secured Communication through Fibonacci Numbers and Unicode Symbols,” International Journal of Scientific and Engineering Research, Vol. 3, Iss.4, April, 2012

[8] S. Chandra, "A comparative survey of symmetric and asymmetric key cryptography", file:///C:/Users/deepanshu/Desktop/workingpaper/working paper/Analysis and Comparison of Substitution and Transposition Cipher.pdf, pp. 83-93, 2014.

[9] K. Renuka and G.N. Harshini, "Analysis and Comparison of Substitution andTransposition Cipher", vol. 6, no. 2, pp. 549-555, 2019.

[10] P. Security, "Recent Parables in Cryptography", vol. 2, no. file:///C:/Users/deepanshu/Desktop/working paper/working paper/Understanding Cryptography by Christof Paar.pdf, pp. 82-86, 2014.

[11] M. Mushtaq Faheem, S. Jamel, A. Hassan Disina, Z. A. Pindar, N. Shafinaz Ahmad Shakir and M. Mat Deris, "A survey on the cryptographic encryption algorithms", Int. J. Adv. Comput. Sci. Appl, vol. 8, no. 11, pp. 333-344, 2017.

[12] B. Purnama and A. H. H. Rohayani, "A New Modified Caesar Cipher Cryptography Method with LegibleCiphertext from a Message to Be Encrypted", Procedia Comput. Sci, vol. 59, no. Iccsci, pp. 195-204, 2015.

[13] A. Jain and A. K. Pandey, "Modeling And Optimizing Of Different Quality Characteristics In Electrical Discharge Drilling Of Titanium Alloy (Grade-5) Sheet", Material Today Proceedings, vol. 18, pp. 182-191, 2019

[14] Vikram Jagannathan, Aparna Mahadevan, Hariharan R., Srinivasan E., "Simultaneous color image compression and encryption using number theory", Proceedings of ICIS 05, 2005, pp. 1.

[15] Mehmet Utku Celika, Gaurav Sharma, A. Murat Tekalp, "Gray-level- embedded lossless image compression", Signal Processing: Image Communication 18, 2003, pp. 443-454.

[16] Said and W. A. Pearlman, "A New, Fast and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees," IEEE Trans. on Circuits and Systems for Video Technology, Vol. 6, No. 3, June 2000, pp. 243.