


Article

Some Properties of the Computation of the Modular Inverse with Applications in Cryptography

Michele Bufalo ¹, Daniele Bufalo ² and Giuseppe Orlando ^{3,4,*} 

¹ Department of Methods and Models for Economics, Territory and Finance, Università degli Studi di Roma "La Sapienza", Via del Castro Laurenziano 9, 00185 Rome, Italy

² Department of Informatics, Università degli Studi di Bari Aldo Moro, Via Orabona 4, 70125 Bari, Italy

³ Department of Mathematics, Università degli Studi di Bari Aldo Moro, Via Orabona 4, 70125 Bari, Italy

⁴ Department of Economics, HSE University, Soyuzna Pechatnikov Street 16, 190121 St. Petersburg, Russia

* Correspondence: giuseppe.orlando@uniba.it; Tel.: +39-080-5049218

Abstract: In the field of cryptography, many algorithms rely on the computation of modular multiplicative inverses to ensure the security of their systems. In this study, we build upon our previous research by introducing a novel sequence, $(z_j)_{j \geq 0}$, that can calculate the modular inverse of a given pair of integers (a, n) , i.e., $a^{-1} \pmod{n}$. The computational complexity of this approach is $\mathcal{O}(a)$, which is more efficient than the traditional Euler's phi function method, $\mathcal{O}(n \ln n)$. Furthermore, we investigate the properties of the sequence $(z_j)_{j \geq 0}$ and demonstrate that all solutions of the problem belong to a specific set, \mathcal{I} , that only contains the minimum values of $(z_j)_{j \geq 0}$. This results in a reduction of the computational complexity of our method, especially when $a \sim n$ and it also opens new opportunities for discovering closed-form solutions for the modular inverse.

Keywords: extended-Euclid algorithm; RSA algorithm; modular multiplicative inverse; public-key cryptography

MSC: 11T71; 11Y16; 11Y05



Citation: Bufalo, M.; Bufalo, D.; Orlando, G. Some Properties of the Computation of the Modular Inverse with Applications in Cryptography. *Computation* **2023**, *11*, 70. <https://doi.org/10.3390/computation11040070>

Academic Editors: Demos T. Tsahalidis and Xinwei Cao

Received: 10 February 2023

Revised: 21 March 2023

Accepted: 21 March 2023

Published: 27 March 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The modulo operation is a mathematical function that calculates the remainder of the division between two numbers, the dividend and the modulus. It is expressed as $a \bmod n$, where a and n are two positive numbers. This operation uses the Euclidean division method to find the remainder of dividing the dividend a by the divisor n .

The modular multiplicative inverse of an integer a is a number x such that ax is congruent to 1 modulo n , or in mathematical terms, $ax \equiv 1 \pmod{n}$. This means that the product of a and x gives a result that is equivalent to 1 when taken modulo n .

Modulo n forms an equivalence relation. The set of all integers equivalent to a modulo n , denoted by \bar{a}_n , is the set $\{\dots, a - 2n, a - n, a, a + n, a + 2n, \dots\}$. This set is known as the congruence class or residue class of the integer a modulo n .

If an integer a has a modular multiplicative inverse modulo n , there are an infinite number of solutions that are equivalent to a with respect to the modulus n . Additionally, for any integer that is congruent to a modulo n , any element from the congruence class of x can serve as a modular multiplicative inverse. This can be represented as the multiplication of congruence classes modulo n , denoted by the symbol \cdot_n , where the modular multiplicative inverse of the congruence class \bar{a} is the congruence class \bar{x} such that $\bar{a} \cdot_n \bar{x} = \bar{1}$.

The multiplication of congruence classes modulo n , represented by the symbol \cdot_n , is analogous to the concept of a multiplicative inverse in the set of real numbers. However, in this case, the numbers are replaced by congruence classes. This operation is used to

solve linear congruences, such as Equation (1), where the goal is to find a solution for x that satisfies the equation and is equivalent to b modulo n :

$$ax \equiv b \pmod{n}. \quad (1)$$

In the field of public-key cryptography, solving Equation (1) is crucial in the RSA algorithm [1], which employs two large prime numbers that are modular multiplicative inverses with respect to a specific modulus to perform secure encryption and decryption operations. Many cryptographic algorithms, such as RSA, ElGamal, and NTRU, heavily rely on the use of modular multiplicative inverses in their calculations. Examples of this can be found in references such as Crandall [2], Rivest [3], Verkhovsky [4,5], ElGamal [6], Rabin [7], and Hoffstein [8]. Additionally, in recent times, Boolean functions have gained attention due to their useful properties in cryptography, specifically regarding “nonlinearity, propagation criterion, resiliency, and balance” [9].

In our previous study [10,11], we examined a particular sequence $(z_j)_{j \geq 0}$ and its ability to determine the modular inverse for a given pair of integers (a, n) , or $a^{-1} \pmod{n}$. We found that the complexity of this search was $\mathcal{O}(a)$, which is less than the classic Euler’s phi function method at $\mathcal{O}(n \ln n)$. Additionally, we delved deeper into the properties of this sequence and discovered that all possible solutions of the problem belong to a set called \mathcal{I} , which only contains the minima of $(z_j)_{j \geq 0}$. This realization reduces the complexity of the algorithm, particularly when $a \sim n$, and opens the possibility of finding a closed formula for the modular inverse.

In this paper, we present a particular sequence $(z_j)_{j \geq 0}$ able to determine the modular inverse for a given pair of integers (a, n) , i.e., $a^{-1} \pmod{n}$. The complexity required for this search is $\mathcal{O}(a)$, which is less than $\mathcal{O}(n \ln n)$ of the classic Euler’s phi function method. Moreover, we investigate more properties of such a sequence $(z_j)_{j \geq 0}$, concluding that all the possible solutions of the problem belong to a proper set, named \mathcal{I} , which contains only the minima of $(z_j)_{j \geq 0}$. This result reduces the complexity of our algorithm, especially when $a \sim n$, and opens the way to the calculation of a possible closed formula for the modular inverse. Last but not least, we compare the complexity of our method with that of the post-quantum encryption (PQC) algorithm.

This research is structured as follows. Section 2 briefly reports the literature with a particular mention of post-quantum cryptography. In Section 3, the different methods for computing the modulus are discussed. Section 4 explores different expressions of β_j that can help in comprehending the behavior of the sequence $(z_j)_{j \geq 0}$ and in identifying the optimal approach for determining the critical index i . Section 5 discusses the results with particular attention to the sequence $(z_j)_{j \geq 0}$ and its properties. It then provides a comparison between the complexity of our algorithm and that of the PQC method. Finally, Section 6 summarizes the research and hints at future developments.

2. Literature

When it was first introduced, RSA was considered to be a highly effective algorithm due to the lack of key exchange in the encryption and decryption process. However, the security of RSA relies heavily on the difficulty of factoring large numbers, a problem that is known to be NP-complete [12]. As technology progressed and computer speed increased, RSA keys began to be broken more frequently. To counteract this, developers have increased the length of the encryption key to ensure the continued security and privacy of systems protected by RSA. There have been other alternative solutions suggested to improve security in RSA cryptography. Some of these include the use of multiple public and private keys (Mezher et al. [13]), an enhanced version of RSA (ESRPKC) that incorporates the Chinese remainder theorem (Kumar et al. [14]), the use of random numbers and their modular multiplicative inverse (Islam et al. [15]), and an optimization algorithm (Cuckoo Search Optimization or CSA) to maintain data integrity in the cloud (Raja et al. [16]). A comprehensive overview of these methods can be found in the study by Mumtaz et al. [17]. In

addition, the literature has proposed many solutions for specific needs such as lightweight algorithms suitable for use on resource-constrained nodes in sensitive applications (Bayat-Sarmadi et al. [18]).

Fault attacks are a type of attack on cryptographic algorithms that take advantage of malicious or unintentional errors introduced during their computation. The concept of Differential Cryptanalysis [19], combined with the pioneering work of Boneh, DeMillo, and Lipton [20,21], has given rise to the field of Differential Fault Attacks (DFA). DFA has revealed that many ciphers can be compromised if the errors can be manipulated in a specific manner. The DFA attack has shown that several ciphers can be compromised if the faults can be suitably controlled, and it is not limited to old ciphers but can be a powerful attack vector even for modern ciphers such as the Advanced Encryption Standard (AES). For a review, see Ali et al. [22]. Finally, on fault-detection methods capable of detecting random faults in the cipher implementation and, at the same time, against intelligent fault attacks, see Dofe et al. [23].

With the advent of post-quantum cryptography, post-quantum cryptography (PQC) will replace ECC/RSA so that every security application from smartphones to blockchains will be affected. However, there are still some issues to solve. For example, the SIKE protocol is a post-quantum candidate for cryptography that is considered to be the best alternative to curve-based cryptography. Nevertheless, its long latency is a drawback, since the serial large-degree isogeny computation, which is dominated by modular multiplications, can make it less competitive compared to other popular post-quantum candidates. A possible solution has been recently suggested by Tian et al. [24] who described an optimized SIKE algorithm, with a focus on achieving high speed and low latency. Furthermore, the rise of quantum computing has driven researchers to develop new security systems that can withstand future attacks. These post-quantum cryptographic approaches include hash-based, code-based, lattice-based, multivariate-quadratic-equations, and secret-key cryptography. They are all potential candidates because they are thought to be resistant to both classical and quantum computers, and applying Shor's algorithm [25], the quantum-computer discrete-logarithm algorithm that can break classical schemes, is believed to be infeasible. Mozaffari-Kermani [26] proposed a method for constructing reliable and error-detection hash trees for stateless hash-based signatures. Such signatures are considered one of the leading post-quantum cryptographic schemes, as they offer security proofs that are based on plausible properties of the underlying hash function.

CRYSTALS-Kyber is a significant public-key encryption and key encapsulation mechanism, as it has been chosen by NIST for standardization and recommended for national security systems by the NSA. Therefore its implementations need to be evaluated for their resistance to side-channel attacks. Dubrova et al. [27] introduced a neural network recursive learning for training to attack ω -order masked implementations of CRYSTALS-Kyber in ARM Cortex-M4 CPU for message recovery. Last but not least, CRYSTALS-Dilithium has been selected by NIST as the new primary standard for quantum-safe digital signatures, and it has a constant-time implementation with consideration for side-channel resilience. For a profiling side-channel attack on the signature scheme CRYSTALS-Dilithium see Berzati et al. [28].

3. Preliminary Results

As is well known, there exists a unique (closed) formula in the literature for the computation of the modular inverse, which is related to Euler's phi function

$$\Phi(n) = n \prod_{p_j | n} \left(1 - \frac{1}{p_j}\right),$$

i.e.,

$$a^{-1} = a^{\Phi(n)-1}. \quad (2)$$

Equation (2) derives directly from Fermat’s little theorem, and its computation has complexity $\mathcal{O}(n \ln n)$. This is because Euler’s phi function is related to the prime factorization with complexity $\mathcal{O}(n \ln n)$. For a further comparison with other classic methods (and their complexities) about the modular inverse see ([10], Section 2) and Table 1.

Table 1. Complexity comparison between the naive method (i.e., recursive multiplications), Euler’s phi function, extended Euclidean algorithm, and the suggested approach described by the pseudocode Algorithm 1.

Method	Naive	Euler’s phi Func.	Ext. Euclidean Algo.	Sugg. Algo.
Complexity	$\mathcal{O}(n)$	$\mathcal{O}(n \ln n)$	$\mathcal{O}(\ln n)$	$\mathcal{O}(a)$

Algorithm 1 Pseudocode of the algorithm for solving (10).

1. Initialize $j = 0, z_0 = 0$;
2. **while** $z_j \neq 1$
3. set $z_j = a(jm - \frac{a-1}{a} + 1) - jn$ and $j = j + 1$;
4. **end**
5. set $i = j - 1$ and $a^{-1} = a(im - \frac{a-1}{a} + 1)$.

We will begin by reviewing the important findings from a previous study by Bufalo et al. [10,11], as well as other relevant information that will be useful for our analysis. For the sake of notation, given $x \in \mathbb{Q}$, we denote by $\lfloor x \rfloor$ the floor integer part of x , and by $\{x\}$ the fractional one, i.e., $\{x\} = x - \lfloor x \rfloor$. We will also be using a sequence called $(z_j)_{j \geq 0}$ in our calculations for finding the modular inverse.

Definition 1. Let a, n be two integers with $0 < a < n$ and $\text{GCD}(a, n) = 1$. Define the sequence $(z_j)_{j \geq 0}$ recursively by the equation

$$\begin{cases} z_j = z_{j-1} + a\beta_j - n & (j \geq 1), \\ z_0 = 0 \end{cases} \tag{3}$$

with

$$\begin{cases} \beta_1 = M \\ \beta_j = \lfloor \frac{n - z_{j-1}}{a} \rfloor + 1 & j \geq 2, \end{cases} \tag{4}$$

where M is the ceiling part of $m := n/a$.

The explicit representation of $(z_j)_{j \geq 0}$ can be found in the next proposition.

Proposition 1. The solution of the difference Equation (3) is given by

$$z_j = a \sum_{h=1}^j \beta_h - jn. \tag{5}$$

Proof. For the proof, see ([10], Proposition 1). □

Special care is deserved to the meaning and the mathematical form assumed by β_j ’s, which allow giving other equivalent expressions of $(z_j)_{j \geq 0}$.

Proposition 2. The sequence $(\beta_j)_{j \geq 1}$ introduced in (4) may be written as

$$\beta_j = \lfloor jm \rfloor - \lfloor (j - 1)m \rfloor. \tag{6}$$

As a consequence, for any $j \geq 1$, we obtain

(i)
$$\sum_{h=1}^j \beta_h = \lfloor jm \rfloor + 1; \tag{7}$$

(ii)
$$z_j = a(\lfloor jm \rfloor + 1) - jn. \tag{8}$$

Proof. For the proof, see ([10], Proposition 2 and Corollary 1). □

The above results imply the next fundamental theorem.

Theorem 1. Let a, n be two integers with $0 < a < n$ and $\text{GCD}(a, n) = 1$. If $(z_j)_{j \geq 0}$ is the sequence of Equation (3), define the "critical" index $i \geq 1$ such that $z_i = 1$. Then the inverse of a modulo n is given by

$$a^{-1} = \lfloor im \rfloor + 1. \tag{9}$$

Proof. See ([10], Theorem 1). □

To illustrate the significance of Theorem 1, we will mention a related result.

Proposition 3. The sequence $(z_j)_{j \geq 1}$ is periodic of period a .

Proof. See ([10], Proposition 3). □

It is immediately clear that the unique limitation of Theorem 1 is the determination of such critical index i , which can be found by solving the following equation

$$a(\lfloor im \rfloor + 1) - in = 1. \tag{10}$$

Although Equation (10) is nonlinear, one observes that $\lfloor im \rfloor = im - \{im\}$, where $\{im\} = \frac{a-1}{a}$ (see [10], Proposition 4). The knowledge of $\{im\}$ jointly with the periodicity information provided by Proposition 3 suggests solving the modular problem (10) by the code detailed in Algorithm 1.

Observe that the complexity of the above algorithm is $\mathcal{O}(a)$. Hence, this procedure is more convenient when $a \ll n$ (e.g., $a < \ln n$). Notice that, even in the worst case $a \sim n$ (i.e., complexity $\mathcal{O}(n)$), the algorithm of Table 1 and the resolving formula (9) is still better compared to the Euler’s phi formula (2), which has complexity $\mathcal{O}(n \ln n)$. Additionally, Section 5 delves further into the advantages of the algorithm when $a \sim n$.

At this point, we present some new properties of $(z_j)_{j \geq 0}$ which will be helpful in the next sections. In particular, we denote by \mathcal{A} the set $[0, a] \cap \mathbb{N}$.

Proposition 4. Let $(z_j)_{j \geq 0}$ be the sequence defined in Equation (3), then

- (i) for any $j \in \mathcal{A}$, it holds $z_j \in [0, n] \cap \mathbb{N}$.
- (ii) for any two different integers j_1, j_2 in \mathcal{A} , one has

$$z_{j_1} \neq z_{j_2}.$$

Proof. First of all, observe that Equation (8) may be rewritten as

$$z_j = a(1 - \{jm\}). \tag{11}$$

- (i) By Definition 1 it is clear that $(z_j)_{j \geq 0} \in \mathbb{N}$ and its (absolute) minimum is given by 0. Moreover, Equation (11) implies that $(z_j)_{j \geq 0}$ is positive since $(1 - \{jm\}) > 0$.
- (ii) Without loss of generality, set $j_2 = j_1 + k$ ($k < a$). Observe that $z_{j_1} = z_{j_2}$ if and only if $\{j_1 m\} = \{j_2 m\}$, which is equivalent to say that $j_2 m = j_1 m + 1$ (if $j_1 < j_2$), or, equivalently $(j_2 - j_1)n = a$, and this is true only if $(j_2 - j_1) = \frac{1}{m} \in \mathbb{Q}$, which is absurd.

□

4. New Results about β_j

In this section, we will study various equivalent formulations of β_j to gain insight into the properties of the sequence $(z_j)_{j \geq 0}$ and to determine the best way to calculate the critical index i .

Proposition 5. *The coefficient β_2 defined in Equation (4) is given by*

$$\beta_2 = \begin{cases} M - 1 & \text{if } \{m\} < 0.5 \\ M & \text{if } \{m\} > 0.5. \end{cases} \tag{12}$$

Proof. It is clear that

$$2\{m\} \begin{cases} < 1 & \text{if } \{m\} < 0.5 \\ > 1 & \text{if } \{m\} > 0.5; \end{cases}$$

so, one has

$$\lfloor 2m \rfloor = 2\lfloor m \rfloor + \lfloor 2\{m\} \rfloor = \begin{cases} 2\lfloor m \rfloor & \text{if } \{m\} < 0.5 \\ 2\lfloor m \rfloor + 1 & \text{if } \{m\} > 0.5, \end{cases}$$

which gives the assertion, being $M = \lfloor m \rfloor + 1$. □

Now, let us introduce the new quantity $(\mathcal{M}_j)_{j \geq 1}$, as follows.

Definition 2. *Let a, n be two positive integers a, n . For any $j \in \mathbb{N}^*$ define*

$$\mathcal{D}_{j,k} := \{a \in \mathbb{N}^* \mid q \mid (jn - k), k \in \mathbb{N}^*\},$$

and

$$\mathcal{M}_j := \sum_{k=1}^{n-1} \mathbb{1}_{\mathcal{D}_{j,k}}(a) \tag{13}$$

which denotes the amount of multipliers of a in $[(j - 1)n + 1, jn]$.

Lemma 1. *Given two positive integers p, q , it holds*

$$\left\lfloor \frac{p}{q} \right\rfloor - \left\lfloor \frac{p-1}{q} \right\rfloor = \begin{cases} 1 & \text{if } q \mid p \\ 0 & \text{otherwise.} \end{cases}$$

Proposition 6. *For any $j \in \mathbb{N}^*$, $j \geq 2$, the coefficients β_j defined in Equation (4), may be rewritten as*

$$\beta_j = \mathcal{M}_j,$$

where \mathcal{M}_j is defined by Equation (13).

Proof. By virtue of Proposition 2, we have

$$\beta_j = \lfloor jm \rfloor - \lfloor (j - 1)m \rfloor \quad (j \geq 2).$$

In particular, it is easy to see that

$$\lfloor jm \rfloor - \lfloor (j - 1)m \rfloor = \sum_{k=0}^{n-1} \left\lfloor \frac{jn - k}{a} \right\rfloor - \left\lfloor \frac{jn - k - 1}{a} \right\rfloor$$

where

$$\left\lfloor \frac{jn - k}{a} \right\rfloor - \left\lfloor \frac{jn - k - 1}{a} \right\rfloor = \begin{cases} 1 & \text{if } a \mid (jn - k) \\ 0 & \text{otherwise,} \end{cases}$$

due to Lemma 1, for any k . Therefore, with refer to Definition 2, we may write

$$\lfloor jm \rfloor - \lfloor (j-1)m \rfloor = \sum_{k=1}^{n-1} \mathbb{1}_{\mathcal{D}_{j,k}}(a),$$

that is $\beta_j = \mathcal{M}_j$. \square

Proposition 7. Consider two positive integers a, n , with $\text{GCD}(a, n) = 1$ and let $m = n/a$. Fixed $j \in [2, a] \cap \mathbb{N}$, let \mathcal{M}_j be the quantity defined by Definition 2.

(i) If $\{m\} < 0.5$ and $\frac{h}{\{m\}} \neq a$ ($h \in \mathbb{N}^*$), then

$$\mathcal{M}_j = \begin{cases} \lfloor m \rfloor & \text{if } j \in (\lfloor \frac{h-1}{\{m\}} \rfloor + 1, \lfloor \frac{h}{\{m\}} \rfloor + 1) \cap \mathbb{N} \\ \lfloor m \rfloor + 1 & \text{if } j = \lfloor \frac{h}{\{m\}} \rfloor + 1 \end{cases} \quad (h \in \mathbb{N}^*). \tag{14}$$

In particular, if there exists $\bar{h} \in \mathbb{N}^*$ such that $\frac{\bar{h}}{\{m\}} = a$, then

$$\mathcal{M}_j = \begin{cases} \lfloor m \rfloor & \text{if } j \in (\lfloor \frac{\bar{h}-1}{\{m\}} \rfloor + 1, a) \cap \mathbb{N} \\ \lfloor m \rfloor + 1 & \text{if } j = a. \end{cases}$$

(ii) If $\{m\} > 0.5$ and $\frac{h}{1-\{m\}} \neq a$ ($h \in \mathbb{N}^*$), then

$$\mathcal{M}_j = \begin{cases} \lfloor m \rfloor + 1 & \text{if } j \in (\lfloor \frac{h-1}{1-\{m\}} \rfloor + 1, \lfloor \frac{h}{1-\{m\}} \rfloor + 1) \cap \mathbb{N} \\ \lfloor m \rfloor & \text{if } j = \lfloor \frac{h}{1-\{m\}} \rfloor + 1 \end{cases} \quad (h \in \mathbb{N}^*). \tag{15}$$

In particular, if there exists $\bar{h} \in \mathbb{N}^*$ such that $\frac{\bar{h}}{1-\{m\}} = a$, then

$$\mathcal{M}_j = \begin{cases} \lfloor m \rfloor + 1 & \text{if } j \in (\lfloor \frac{\bar{h}-1}{1-\{m\}} \rfloor + 1, a) \cap \mathbb{N} \\ \lfloor m \rfloor & \text{if } j = a. \end{cases}$$

Proof. It is clear that $\mathcal{M}_j \in \{\lfloor m \rfloor, \lfloor m \rfloor + 1\}$, for any $j \in \mathbb{N}^*$. We prove Formulas (14) and (15) by induction on h .

(i) If $\{m\} < 0.5$, Propositions 5 and 6 yield that $\mathcal{M}_2 = \lfloor m \rfloor$. Let us compute the smallest integer j ($j > 2$) such that $\mathcal{M}_j = \lfloor m \rfloor + 1$. This is equivalent to solving the following equation:

$$\lfloor jm \rfloor = j\lfloor m \rfloor + 1, \tag{16}$$

which may be rewritten as

$$\lfloor j\{m\} \rfloor = 1,$$

being $\lfloor jm \rfloor = j\lfloor m \rfloor + \lfloor j\{m\} \rfloor$. The latter equation holds if

$$1 \leq j\{m\} < 2,$$

therefore, the smallest integer j solving Equation (16) is given by $\lfloor \frac{1}{\{m\}} \rfloor + 1$. This prove Formula (14) for $h = 1$.

Now, assume that Formula (14) holds for $h - 1$. Since $\mathcal{M}_{\lfloor \frac{h-1}{\{m\}} \rfloor + 2} = \lfloor m \rfloor$, we want to compute the smallest integer j ($j > \lfloor \frac{h-1}{\{m\}} \rfloor$) such that $\mathcal{M}_j = \lfloor m \rfloor + 1$, that is equivalent to solve

$$\lfloor jm \rfloor = j\lfloor m \rfloor + h, \tag{17}$$

or, equivalently,

$$\lfloor j\{m\} \rfloor = h.$$

Hence, the smallest integer j solving Equation (17) is given by $\lfloor \frac{h}{\{m\}} \rfloor + 1$.

In particular, if there exists $\bar{h} \in \mathbb{N}^*$ such that $\frac{\bar{h}}{\{m\}} = a$, then the smallest integer j ($j > \lfloor \frac{\bar{h}-1}{\{m\}} \rfloor$) solving

$$\lfloor j\{m\} \rfloor = \bar{h},$$

or, equivalently,

$$\bar{h} \leq j\{m\} < \bar{h} + 1,$$

is given by a (being $\bar{h}/\{m\} \in \mathbb{N}^*$).

- (ii) The assertion comes arguing similarly to the case (i). More specifically, here, Equation (17) is replaced by

$$\lfloor jm \rfloor = j\lfloor m + 1 \rfloor - h, \tag{18}$$

which is equivalent to

$$\lfloor j(1 - \{m\}) \rfloor = h,$$

and the smallest integer j solving Equation (18) is given by $\lfloor \frac{h}{1-\{m\}} \rfloor + 1$.

□

5. Empirical Findings and Discussion

Fixed $h \in \mathcal{A}$, an interesting development inspired by Proposition 7 concerns the rule of the indices $\lfloor \frac{h}{\{m\}} \rfloor$ when $\{m\} < 0.5$, or $\lfloor \frac{h}{1-\{m\}} \rfloor$ when $\{m\} > 0.5$. In particular, as highlighted from Theorem 2, one has that

- The sequence $(z_j)_{j \geq 0}$ admits local minimum at $\lfloor \frac{h}{\{m\}} \rfloor$ if $\{m\} < 0.5$, or $\lfloor \frac{h}{1-\{m\}} \rfloor$ if $\{m\} > 0.5$.
- The critical index i coincides with one of such indices.

This result is confirmed by computational experiments, as one can see from Figure 1. For a better understanding of the figure, as explained in ([10], Example 4), the blue line represents the series z_j for $a = 91, n = 131$, starting from $j = 1$ and with a fixed (large) integer $j = 200$. The red line represents the periodic part of z_j , which arises from the critical index $i = 25$ and $(i + a) = 116$ (highlighted by the red circles). In this example, $m = 0.4396 < 0.5$, so all the minima are of type $\lfloor \frac{h}{m} \rfloor$, which are 2, 4, 6, 9, 11, 13, 15, 18, 20, 22, 25, and obviously, 25 is one of them. We will refer to the previous set as \mathcal{I} . It is worth noting that the 11th index of \mathcal{I} corresponds to the critical index $i = 25$, such that $z_{25} = 1$. Figure 2 shows the set \mathcal{I} for $h \leq 30$.

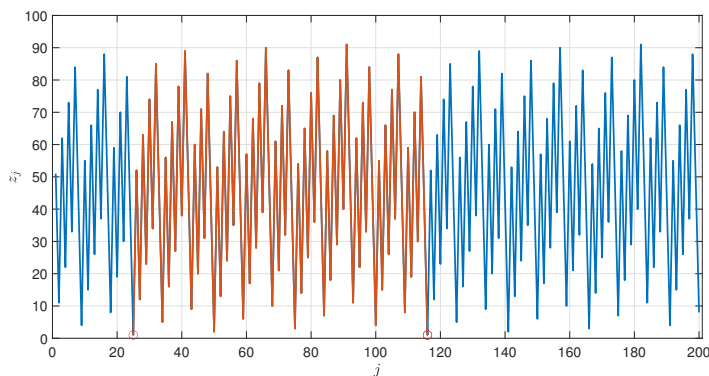


Figure 1. The sequence $(z_j)_{1 \leq j \leq 200}$ for the case where $a = 91$ and $n = 131$. The red line highlights the entire sequence between two consecutive unitary values of z_i (represented by red circles).

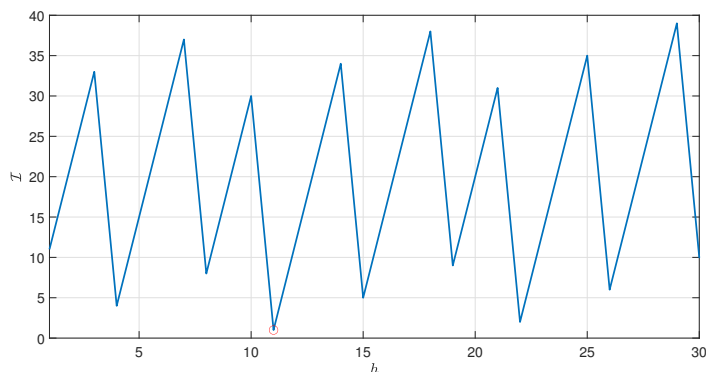


Figure 2. Set \mathcal{T} representing the minima of $(z_j)_{j \geq 0}$ when $a = 91$ and $n = 131$.

Theorem 2. The critical index i belongs to the set

$$\mathcal{I} = \begin{cases} \left\{ \left\lfloor \frac{h}{\{m\}} \right\rfloor \mid h \in \mathcal{A} \right\} & \text{if } \{m\} < 0.5, \\ \left\{ \left\lfloor \frac{h}{1-\{m\}} \right\rfloor \mid h \in \mathcal{A} \right\} & \text{if } \{m\} > 0.5, \end{cases} \tag{19}$$

Proof. It is clear that the critical index i has to be a local minimum of the sequence $(z_j)_{j \geq 0}$ that assumes only integer values and has 0 as an absolute minimum (see Proposition 4). Hence, it remains to prove that \mathcal{I} is the set of the local minimum of $(z_j)_{j \geq 0}$. Let us consider just the case $\{m\} < 0.5$ for simplicity (the other one is analogous). It is easy to see, from Proposition 7, that

$$z_{\lfloor \frac{h}{\{m\}} \rfloor + 1} - z_{\lfloor \frac{h}{\{m\}} \rfloor} = a\beta_{\lfloor \frac{h}{\{m\}} \rfloor + 1} - n = a(\lfloor m \rfloor + 1) - n > 0,$$

and

$$z_{\lfloor \frac{h}{\{m\}} \rfloor} - z_{\lfloor \frac{h}{\{m\}} \rfloor - 1} = a\beta_{\lfloor \frac{h}{\{m\}} \rfloor} - n = a(\lfloor m \rfloor) - n < 0,$$

for any $h \in \mathcal{A}$, and this concludes the proof. \square

In light of the above results, Algorithm 2 can be rewritten as follows.

Algorithm 2 Pseudocode of the optimized algorithm solving (10).

1. Initialize $h, j = 0; z_0 = 0; m = n/a;$
 2. **while** $z_j \neq 1$
 3. **if** $\{m\} < 0.5$
 4. set $j = \lfloor \frac{h}{\{m\}} \rfloor;$
 5. **else**
 6. set $j = \lfloor \frac{h}{1-\{m\}} \rfloor;$
 7. **end**
 8. set $z_j = a(jm - \frac{a-1}{a} + 1) - jn$ and $h = h + 1;$
 9. **end**
 10. set $i = j - 1$ and $a^{-1} = a(im - \frac{a-1}{a} + 1).$
-

Moreover, what is observed from an empirical point of view is that:

- The sequences of the minimum of $(z_j)_{j \geq 0}$ oscillates till $j = i$ (see Figure 2);
- The relative minimum defined in (19) belongs to the bundle of parallel straight lines (see Figure 3)

$$\mathcal{F} = \left\{ y = \frac{z_{\lfloor \frac{h+1}{\{m\}} \rfloor} - z_{\lfloor \frac{h}{\{m\}} \rfloor}}{\lfloor \frac{h+1}{\{m\}} \rfloor - \lfloor \frac{h}{\{m\}} \rfloor} x + \left\lfloor \frac{h}{\{m\}} \right\rfloor + k \mid k \in [0, n] \cap \mathbb{N} \right\} \quad (h \in \mathcal{A})$$

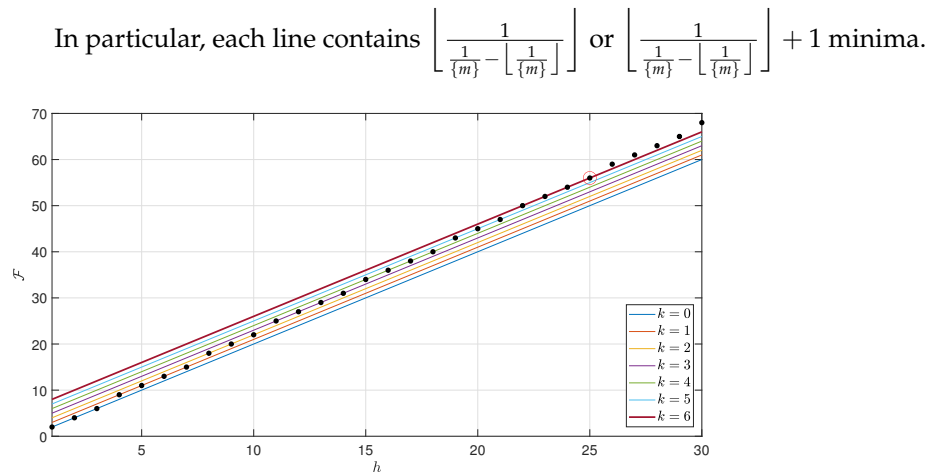


Figure 3. Set \mathcal{F} representing the bundle of parallel straight lines passing related to the minima of $(z_j)_{j \geq 0}$ (when $a = 91$ and $n = 131$), for different values of k .

Remark 1. We end with a note about the complexity of our algorithm. In the spirit of Theorem 2, if we restrict the searching of the critical index i to the set \mathcal{I} , also the complexity of the algorithm is reduced by a factor $\left\lfloor \frac{1}{\{m\}} \right\rfloor$, i.e., from $\mathcal{O}(a)$ to $\mathcal{O}\left(\left\lfloor \frac{a}{\left\lfloor \frac{1}{\{m\}} \right\rfloor} \right\rfloor\right)$. As explained in Section 3, the complexity $\mathcal{O}(a)$ is very good with respect to the literature, especially when $a \ll n$. However, even when $a \sim n$, the complexity $\mathcal{O}\left(\left\lfloor \frac{a}{\left\lfloor \frac{1}{\{m\}} \right\rfloor} \right\rfloor\right)$ sounds well. Indeed, in the extreme case $a = n - 1$, one has $m = 1 + \frac{1}{n}$ and $\left\lfloor \frac{1}{\{m\}} \right\rfloor$ approaches to 1 for large n . In other words, when a tends to n , one has that the complexity reduction tends to 100%, so that the resulting complexity is a constant, i.e., $\mathcal{O}(1)$.

In the non-trivial case where $a = n - 2$, a complexity reduction of approximately 50% is observed. For example, if $a = 327$ and $n = 329$, then $\left\lfloor \frac{1}{\{m\}} \right\rfloor = 163$ and $\frac{163}{327} = 0.4985$. In particular, in such case, the critical index i is just $\left\lfloor \frac{1}{\{m\}} \right\rfloor$.

Therefore, we can conclude that our algorithm runs very well when $a \ll n$ or $a \sim n$, while the worst case is the middle one, i.e., $a \sim \frac{n}{2}$.

Post-Quantum Cryptography (PQC)

As is well known, the National Institute of Standards and Technology (NIST) has launched a program and competition to standardize one or more post-quantum cryptography (PQC) algorithms to fight against quantum attacks. In recent work, Huang [29] has conducted an early mathematical analysis of lattice-based and polynomial-based PQC. Such analysis can help businesses and organizations leverage NIST-selected PQC algorithms to safeguard their digital services from quantum attacks. In particular, the brute force failure probability for polynomial or multivariate PQC is calculated using Yitang Zhang’s Landau-Siegel zero bound according to [30].

In Figure 4 we compare the complexity of our optimized algorithm (see Algorithm 2) with those of coming from the post-quantum cryptography (PQC) architecture which was estimated by Huang [29] in Matlab. More specifically, Figure 4 represents the logarithm of complexity needed to encrypt/decrypt a message with a public/private key (modulus n). As shown, our proposed algorithm, in the best case, is better than the PCQ up to $n = 3000$.

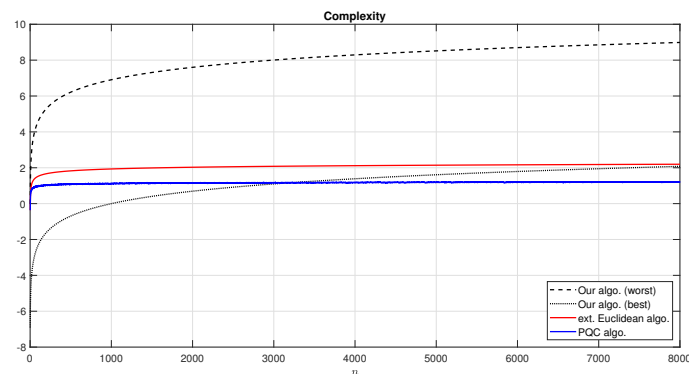


Figure 4. Logarithm of complexity $\mathcal{O}(\cdot)$ comparison between our algorithm in both a possible worst and better case (i.e., $a = 0.001n$ and $a = 0.999n$, respectively—black dotted lines), the extended Euclidean algorithm (red line) and the post-quantum cryptography (PQC) algorithm (blue line).

6. Conclusions

This research builds upon previous work [10,11] where we introduced the concept of the modulo operation and discussed the standard methods for determining the inverse modulo n .

In the above-mentioned research, we determined that to find a closed-form solution for the equation in Equation (10), it was necessary to study the properties of the sequence $(z_j)_{j \geq 0}$. In this article, we expand on our previous aforementioned research by introducing a new sequence, $(z_j)_{j \geq 0}$, which can efficiently calculate the modular inverse of a given pair of integers (a, n) , i.e., $a^{-1} \pmod n$, particularly in the non-trivial case $a = n - 2$. This new method has a computational complexity of $\mathcal{O}(a)$, which is more efficient than the traditional Euler's phi function method, which has a computational complexity of $\mathcal{O}(n \ln n)$. Additionally, we examine the properties of the sequence $(z_j)_{j \geq 0}$ and demonstrate that all solutions to the problem belong to a specific set, \mathcal{I} , that only contains the minimum values of $(z_j)_{j \geq 0}$. This leads to a reduction in the computational complexity of our method, especially when $a \sim n$, and also opens up new possibilities for finding closed-form solutions for the modular inverse.

Future studies will focus on the characteristics of the minimum sequences to understand the emergence of the critical index i , and to find a closed formula for the modular inverse.

Author Contributions: Conceptualization, M.B.; methodology, M.B. and D.B.; software, D.B.; validation, G.O., M.B. and D.B.; formal analysis, M.B. and D.B.; investigation, G.O., M.B. and D.B.; resources, M.B. and D.B.; data curation, M.B. and D.B.; writing—original draft preparation, M.B.; writing—review and editing, G.O. and M.B.; visualization, G.O. and M.B.; supervision, G.O. and M.B.; project administration, G.O. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are available on request from the corresponding author.

Acknowledgments: G.O. and M.B. are members of GNAMPA and INdAM research groups.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Rivest, R.L.; Shamir, A.; Adleman, L.M. Cryptographic Communications System and Method. US Patent 4,405,829, 20 September 1983.
2. Crandall, R.; Pomerance, C.B. *Prime Numbers: A Computational Perspective*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2006; Volume 182.
3. Rivest, R.L.; Shamir, A.; Adleman, L. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM* **1978**, *21*, 120–126. [[CrossRef](#)]
4. Verkhovsky, B. Overpass-Crossing Scheme for Digital Signature. In Proceedings of the International Conference on System Research, Informatics and Cybernetics, Baden-Baden, Germany, 22–25 July 2001; Volume 30.

5. Verkhovsky, B. Enhanced Euclid Algorithm for Modular Multiplicative Inverse and Its Application in Cryptographic Protocols. *IJCNS* **2010**, *3*, 901–906. [[CrossRef](#)]
6. ElGamal, T. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Trans. Inf. Theory* **1985**, *31*, 469–472. [[CrossRef](#)]
7. Rabin, M.O. *Digitalized Signatures and Public-Key Functions as Intractable as Factorization*; Technical Report; Massachusetts Institute of Technology Cambridge Lab for Computer Science: Cambridge, MA, USA, 1979.
8. Hoffstein, J.; Pipher, J.; Silverman, J.H.; Silverman, J.H. *An Introduction to Mathematical Cryptography*; Springer: Berlin/Heidelberg, Germany, 2008; Volume 1.
9. Sosa-Gómez, G.; Paez-Osuna, O.; Rojas, O.; Madarro-Capó, E.J. A New Family of Boolean Functions with Good Cryptographic Properties. *Axioms* **2021**, *10*, 42. [[CrossRef](#)]
10. Bufalo, M.; Bufalo, D.; Orlando, G. A Note on the Computation of the Modular Inverse for Cryptography. *Axioms* **2021**, *10*, 116. [[CrossRef](#)]
11. Bufalo, D.; Bufalo, M.; Orlando, G.; Tetta, R. A new algorithm to find prime numbers with less memory requirements. *J. Discret. Math. Sci. Cryptogr.* **2023**, *in press*. [[CrossRef](#)]
12. Somani, U.; Lakhani, K.; Mundra, M. Implementing digital signature with RSA encryption algorithm to enhance the Data Security of cloud in Cloud Computing. In Proceedings of the 2010 First International Conference on Parallel, Distributed and Grid Computing (PDGC 2010), Solan, India, 28–30 October 2010; pp. 211–216.
13. Mezher, A.E. Enhanced RSA cryptosystem based on multiplicity of public and private keys. *Int. J. Electr. Comput. Eng.* **2018**, *8*, 3949. [[CrossRef](#)]
14. Kumar, V.; Kumar, R.; Pandey, S. An enhanced and secured RSA public key cryptosystem algorithm using Chinese remainder theorem. In Proceedings of the International Conference on Next Generation Computing Technologies, Dehradun, India, 30–31 October 2017; pp. 543–554.
15. Islam, M.A.; Islam, M.A.; Islam, N.; Shabnam, B. A modified and secured RSA public key cryptosystem based on “n” prime numbers. *J. Comput. Commun.* **2018**, *6*, 78. [[CrossRef](#)]
16. Raja shree, S.; Chilambu Chelvan, A.; Rajesh, M. An efficient RSA cryptosystem by applying cuckoo search optimization algorithm. *Concurr. Comput. Pract. Exp.* **2019**, *31*, e4845. [[CrossRef](#)]
17. Mumtaz, M.; Ping, L. Forty years of attacks on the RSA cryptosystem: A brief survey. *J. Discret. Math. Sci. Cryptogr.* **2019**, *22*, 9–29. [[CrossRef](#)]
18. Bayat-Sarmadi, S.; Kermani, M.M.; Azarderakhsh, R.; Lee, C.Y. Dual-Basis Superserial Multipliers for Secure Applications and Lightweight Cryptographic Architectures. *IEEE Trans. Circuits Syst. II Express Briefs* **2013**, *61*, 125–129. [[CrossRef](#)]
19. Biham, E.; Shamir, A. Differential fault analysis of secret key cryptosystems. In *Advances in Cryptology—CRYPTO ’97*; Springer: Berlin, Germany, 2006; pp. 513–525. [[CrossRef](#)]
20. Boneh, D.; DeMillo, R.A.; Lipton, R.J. On the Importance of Eliminating Errors in Cryptographic Computations. *J. Cryptol.* **2001**, *14*, 101–119. [[CrossRef](#)]
21. Boneh, D.; DeMillo, R.A.; Lipton, R.J. On the Importance of Checking Cryptographic Protocols for Faults. In *Advances in Cryptology—EUROCRYPT ’97*; Springer: Berlin, Germany, 2001; pp. 37–51. [[CrossRef](#)]
22. Ali, S.; Guo, X.; Karri, R.; Mukhopadhyay, D. Fault Attacks on AES and Their Countermeasures. In *Secure System Design and Trustable Computing*; Springer: Cham, Switzerland, 2016; pp. 163–208. [[CrossRef](#)]
23. Dofe, J.; Frey, J.; Pahlevanzadeh, H.; Yu, Q. Strengthening SIMON Implementation Against Intelligent Fault Attacks. *IEEE Embed. Syst. Lett.* **2015**, *7*, 113–116. [[CrossRef](#)]
24. Tian, J.; Wu, B.; Wang, Z. High-Speed FPGA Implementation of SIKE Based on an Ultra-Low-Latency Modular Multiplier. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2021**, *68*, 3719–3731. [[CrossRef](#)]
25. LaPierre, R. Shor Algorithm. In *Introduction to Quantum Computing*; Springer: Cham, Switzerland, 2021; pp. 177–192. [[CrossRef](#)]
26. Mozaffari-Kermani, M.; Azarderakhsh, R. Reliable hash trees for post-quantum stateless cryptographic hash-based signatures. In Proceedings of the 2015 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFTS), Amherst, MA, USA, 12–14 October 2015; pp. 103–108. [[CrossRef](#)]
27. Dubrova, E.; Ngo, K.; Gärtner, J. Breaking a Fifth-Order Masked Implementation of CRYSTALS-Kyber by Copy-Paste. *Cryptology ePrint Archive*. 2022. Available online: <https://eprint.iacr.org/2022/1713> (accessed on 20 March 2023).
28. Berzati, A.; Viera, A.C.; Chartouni, M.; Madec, S.; Vergnaud, D.; Vigilant, D. A Practical Template Attack on CRYSTALS-Dilithium. *Cryptology ePrint Archive*. 2023. Available online: <https://eprint.iacr.org/2023/050> (accessed on 20 March 2023).
29. Steed, H. Integer-Complexity-Bound-of-Post-Quantum-Cryptography. 2023. Available online: <https://github.com/steedhuang/Integer-Complexity-Bound-of-Post-Quantum-Cryptography> (accessed on 20 January 2023).
30. Zhang, Y. Discrete mean estimates and the Landau-Siegel zero. *arXiv* **2022**, arXiv:2211.02515.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.