axioms

MDPI

Article

A Note on the Computation of the Modular Inverse for Cryptography

Michele Bufalo [1], Daniele Bufalo [2] and Giuseppe Orlando [3,*]

[1] Department of Methods and Models for Economics, Università degli Studi di Roma "La Sapienza", Territory and Finance, Via del Castro Laurenziano 9, 00185 Roma, Italy; michele.bufalo@uniroma1.it
[2] Department of Informatics, Università degli Studi di Bari Aldo Moro, Via Orabona 4, 70125 Bari, Italy; danielebufalo@libero.it
[3] Department of Economics and Finance, Università degli Studi di Bari Aldo Moro, Via C. Rosalba 53, 70124 Bari, Italy
* Correspondence: giuseppe.orlando@uniba.it; Tel.: +39-080-5049218

Abstract: In literature, there are a number of cryptographic algorithms (RSA, ElGamal, NTRU, etc.) that require multiple computations of modulo multiplicative inverses. In this paper, we describe the modulo operation and we recollect the main approaches to computing the modulus. Then, given a and n positive integers, we present the sequence $(z_j)_{j\geq 0}$, where $z_j = z_{j-1} + a\beta_j - n$, $a < n$ and $\text{GCD}(a,n) = 1$. Regarding the above sequence, we show that it is bounded and admits a simple explicit, periodic solution. The main result is that the inverse of a modulo n is given by $a^{-1} = \lfloor im \rfloor + 1$ with $m = n/a$. The computational cost of such an index i is $\mathcal{O}(a)$, which is less than $\mathcal{O}(n \ln n)$ of the Euler's phi function. Furthermore, we suggest an algorithm for the computation of a^{-1} using plain multiplications instead of modular multiplications. The latter, still, has complexity $\mathcal{O}(a)$ versus complexity $\mathcal{O}(n)$ (naive algorithm) or complexity $\mathcal{O}(\ln n)$ (extended Euclidean algorithm). Therefore, the above procedure is more convenient when $a << n$ (e.g., $a < \ln n$).

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

MSC: 11T71; 11Y16; 11Y05

Citation: Bufalo, M.; Bufalo, D.; Orlando, G. A Note on the Computation of the Modular Inverse for Cryptography. *Axioms* **2021**, *10*, 116. https://doi.org/10.3390/axioms10020116

Academic Editors: Hsien-Chung Wu and Javier Fernandez

Received: 26 April 2021
Accepted: 5 June 2021
Published: 9 June 2021

1. Introduction

The modulo operation returns the remainder of a division, after one number is divided by another number called "modulus". In other terms, given two positive numbers a and n, $a \mod n$ is the remainder of the Euclidean division of the dividend a by the divisor n.

A modular multiplicative inverse of an integer a is an integer x such that the product ax is congruent to 1 with respect to the modulus n, and it is denoted as

$$ax \equiv 1 \pmod{n}.$$

Modulo n is an equivalence relation. The equivalence class of the integer a, denoted by $\overline{a}_n$, is the set $\{\ldots, a-2n, a-n, a, a+n, a+2n, \ldots\}$. This set, consisting of all the integers congruent to a modulo n, is called congruence class or residue class of the integer a modulo n.

If a has an inverse modulo n, then there are an infinite number of solutions that belong to the congruence class with respect to the said modulus. In addition, any integer that is congruent to a will have any element of x's congruence class as a modular multiplicative inverse. In other terms, denoted with the symbol $\cdot_n$, the multiplication of equivalence classes modulo n, the modulo multiplicative inverse of the congruence class $\overline{a}$ is the congruence class $\overline{x}$ such that:

$$\overline{a} \cdot_n \overline{x} = \overline{1}.$$

This multiplication is the analogue of the multiplicative inverse in the set of real numbers where numbers are replaced by congruence classes. Therefore, a fundamental use of this operation is to solve (whenever possible) linear congruences of the form

$$ax \equiv b \pmod{n}. \tag{1}$$

The solution of Equation (1) has practical applications in the field of public-key cryptography and, in particular, in the Rivest–Shamir–Adleman (RSA) algorithm [1] where encryption and decryption are performed by using a pair of large prime numbers that are multiplicative inverses with respect to a selected modulus.

When invented, RSA was considered one of the most effective algorithms because there was no key exchange in the encryption and decryption processes. In the RSA algorithm, the strength depends on the factorization problem that is NP complete [2] and the key length was the only way to protect systems. However, the RSA key is broken from time to time due to the development of both software and computer speed. To counter that, developers have increased key length from one time to another to maintain a high security and privacy to systems that are protected by the RSA. Other countermeasures vary from using multiple public and private keys [3] to enhance and secure the RSA public key cryptosystem (ESRPKC) algorithm using the Chinese remainder theorem [4], from the use of a pair of random numbers and their modular multiplicative inverse [5] to the Cuckoo Search Optimization (CSA) algorithm for securing data integrity in the cloud [6]. For a survey, see Mumtaz et al. [7].

As mentioned, cryptographic algorithms rely on multiple computations of modulo multiplicative inverses. Examples are the RSA cryptographic algorithm by [8,9], RSA with digital signature [10], ElGamal cryptocol [11]; encryption and decryption schemes based on extraction of square roots [12], NTRU cryptosystem [13], modular multiplicative inverse (MMI) for cryptanalysis of public-key cryptographic protocols [14], etc. Recently, Boolean functions have gained attraction because of some interesting properties from a cryptographic point of view such as "nonlinearity, propagation criterion, resiliency, and balance" [15]. However, following similar research on RSA cryptographic algorithms, we focused on the problem of encrypting/decoding information based on the use of the vector-modular methods. For example, Yakymenko et al. [16] suggest a modular exponential to "replace the complex operation of modular multiplication with the addition operation, which increases the speed of the RSA cryptosystem". In our case, instead, we investigate the properties of the sequence $(z_j)_{j\geq 0}$ in Definition 1, which we show to be useful for computing the inverse modulo. In particular, for the above sequence, we show that it is bounded and admits a simple explicit, periodic solution. Next, we illustrate that the inverse of a modulo n is given by $a^{-1} = \lfloor im \rfloor + 1$ with $m = n/a$. The advantage is that the computational cost of such an index i is $\mathcal{O}(a)$ versus $\mathcal{O}(n \ln n)$ of the Euler's phi function. Finally, we suggest an algorithm for calculating a^{-1} using plain multiplications instead of modular multiplications. The latter, again, has complexity $\mathcal{O}(a)$ versus complexity $\mathcal{O}(\ln n)$ of the extended Euclidean algorithm. Therefore, the above procedure is more convenient when $a << n$ (e.g., $a < \ln n$). Those results are new in literature.

This work is divided as follows: Section 2 describes the main approaches to the computation of modulus. Section 3 illustrates the sequence $(z_j)_{j\geq 0}$ along with some of its properties. Section 4 presents the conclusions.

2. Main Approaches to the Computation of Modulus

In the following, we describe the most common methods to compute the inverse modulo n.

2.1. Naive Method (Recursive Multiplications)

This is the simplest way to compute the inverse of a positive integer a, modulo n, with $a < n$ and greatest common divisor $\mathrm{GCD}(a, n) = 1$. We have to multiply a by all the

elements of $\mathbb{N}_n^* = \{1, 2, ..., n-1\}$ and the first of them which gives a product equal to 1 (modulo n) will be the inverse of a. The complexity in this case is $\mathcal{O}(n)$.

Example 1. *To find the inverse of $a = 6$ modulo $n = 7$, we have to multiply a by every element of $\mathbb{N}_7^* = \{1, 2, ..., 6\}$, i.e.,*

$$1 \cdot 6 = 6 \pmod 7, \quad 2 \cdot 6 = 12 \equiv 5 \pmod 7, \quad 3 \cdot 6 = 18 \equiv 4 \pmod 7,$$

$$4 \cdot 6 = 24 \equiv 3 \pmod 7, \quad 5 \cdot 6 = 30 \equiv 2 \pmod 7, \quad 6 \cdot 6 = 36 \equiv 1 \pmod 7.$$

Therefore, $a^{-1} = 6$ modulo 7.

2.2. Euler's Phi Function

The following approach was introduced in modern terms by Gauss with reference to Euler (even though the method has been reported before [17]). Given a positive integer n, the Euler's phi function $\Phi(n)$ (or Euler's totient function) counts the number of primes, up to n, which are relatively prime to n. It can be expressed as

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

with p_j's being the primes dividing n. Given a positive integer a, with $a < n$ and $\mathrm{GCD}(a, n) = 1$, one has

$$a^{\Phi(n)} \equiv 1 \pmod n$$

due to the well-known Fermat's little theorem. The above relation provides an explicit formula for the inverse of a modulo n that is

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

However, the calculation of $\Phi(n)$ is equivalent to doing the prime factorization of n, hence the complexity of Formula (2) is $\mathcal{O}(n \ln n)$. Thus, despite (2) giving a closed formula, it is less convenient than a recursive algorithm (like those of Sections 2.1 and 2.3).

Example 2. *To compute the inverse of* 23 *modulo* 36 *through Formula* (2)*, one has*

$$\Phi(36) = 36\left(1 - \frac{1}{2}\right)\left(1 - \frac{1}{3}\right) = 12,$$

and $23^{12-1} \equiv 11 \pmod{36}$*, i.e.,* $23^{-1} \equiv 11 \pmod{36}$.

2.3. Extended Euclidean Algorithm

One of the ancient methods to compute the GCD between two integers a, b, with $a > b$, is given by the Euclidean algorithm. It is based on the following property: if both a and b divide a same integer c, then also their difference $a - b$ divides c. The algorithm states that $\mathrm{GCD}(a, b) = b$ if the difference $d = a - b$ is equal to b; otherwise, a, b are replaced by $\max\{a - b, b\}$ and $\min\{a - b, b\}$, respectively, and the previous procedure is repeated by computing the new difference d. Table 1 describes the pseudocode of the algorithm.

Table 1. Pseudocode of the Euclidean algorithm (repeated differences).

1. Initialize $i = 0$, $a_i = a,\ b_i = b$ and $Flag = 0$;
2. **while** $Flag = 0$
3. set $d_i = a_i - b_i$;
4. **if** $a_i - b_i = b_i$
5. set $Flag = 1$;
6. **else** set $i = i + 1$, $a_i = \max\{a_{i-1} - b_{i-1}, b_{i-1}\}$, and $b_i = \min\{a_{i-1} - b_{i-1}, b_{i-1}\}$;
7. **end**
8. **end**
9. set $\mathrm{GCD}(a, b) = d_i$.

An interesting extension of such method works with repeated divisions instead of the repeated differences. By computing the following quotients q_i and remainders r_i,

$$a = b \cdot q_0 + r_0,$$

$$b = r_0 \cdot q_1 + r_1,$$

$$r_0 = r_1 \cdot q_2 + r_2,$$

$$\vdots$$

$$r_{i-1} = r_i \cdot q_{i+1} + r_{i+1},$$

$$\vdots$$

it is possible to say that $\mathrm{GCD}(a, b)$ is the last non-zero remainder r_i. The complexity of this method is $\mathcal{O}(\ln\ n)$. The pseudocode of this procedure is reported in Table 2.

Table 2. Pseudocode of the Euclidean algorithm (repeated divisions).

1. Initialize $i = 0$, $a_i = a,\ b_i = b$, and let q_i, r_i be the quotient and the remainder of a_i / b_i, respectively ;
2. **if** $r_0 = 0$
3. let $\mathrm{GCD}(a, b) = b$;
4. **else**
5. **while** $r_i \neq 0$
6. set $i = i + 1$, $a_i = b_{i-1}$,$b_i = r_{i-1}$, and let q_i, r_i be the quotient and the remainder of a_i / b_i, respectively;
7. **end**
8. set $\mathrm{GCD}(a, b) = r_{i-1}$.
9. **end**

The above method allows us to compute the inverse modulo n through the so-called Bézouts's identity which states that there exist two integer s, t such that

$$\mathrm{GCD}(a, b) = s \cdot a + t \cdot b.$$

The numbers s, t can be computed from the quotients q_i ($i \geq 0$), by reversing the order of the equations in the Euclidean algorithm (with repeated divisions). Beginning with the last non-zero remainder r_i, we can write

$$\mathrm{GCD}(a, b) = r_i = r_{i-2} - q_i \cdot r_{i-1}.$$

The quantity r_{i-1}, r_{i-2} may be likewise expressed in terms of their quotients and preceding remainders, i.e.,

$$r_{i-1} = r_{i-3} - q_{i-1} \cdot r_{i-2},$$

$$r_{i-2} = r_{i-4} - q_{i-2} \cdot r_{i-3}.$$

Substituting these formulas into the first equation yields $\mathrm{GCD}(a,b)$ as a linear sum of r_{i-3}, r_{i-4}. The process of substituting remainders by formulas involving their predecessors can be continued until a and b are reached, as follows:

$$\vdots$$

$$r_2 = r_0 - q_2 \cdot r_1,$$
$$r_1 = b - q_1 \cdot r_0,$$
$$r_0 = a - q_0 \cdot b.$$

After all the remainders r_i ($i \geq 0$) have been replaced, the final equation expresses $\mathrm{GCD}(a,b)$ as the linear combination $s \cdot a + t \cdot b$.

In the special case that $\mathrm{GCD}(a,b) = 1$, then t is the multiplicative inverse of b, modulo a, or, equivalently, s is the multiplicative inverse of a, modulo b.

The pseudocode of this method is shown in Table 3.

Table 3. Pseudocode of the inverse modulo n (through the extended Euclidean algorithm).

1. Compute q_j, r_j, $-2 \leq j \leq i$ (where $r_{-1} = a$, $r_{-2} = n$ and $r_i = 1$ is the last remainder) by the extended Euclidean algorithm (see Table 2) between a and n;
2. **for** $j = i : -1 : -2$
3. write r_j as linear combination of r_{j-1} and r_{j-2};
4. **end**
5. set a^{-1} equal to the coefficient multiplied by a in the final recursive relation.

Example 3. *Consider* $a = 27$ *and* $n = 392$. *Obviously,* $GCD(27,392) = 1$. *The extended Euclidean algorithm gives*

$$392 = 27 \cdot 14 + 14,$$
$$27 = 14 \cdot 1 + 13,$$
$$14 = 13 \cdot 1 + 1.$$

By rewriting the next steps backward, we obtain

$$1 = 14 - 13 \cdot 1 = 14 - (27 - 14 \cdot 1) = 2 \cdot 14 - 2 = 2(392 - 27 \cdot 14) - 27 = 2 \cdot 392 + 27(-29),$$

where $-29 \equiv 363 \pmod{392}$. *Hence, we can conclude that* $27^{-1} \equiv 363 \pmod{392}$.

3. The Sequence z_j: Definition and Properties

In this section, given a and n positive integers, we define the sequence $(z_j)_{j \geq 0}$, where $z_j = z_{j-1} + a\beta_j - n$, $a < n$ and $\mathrm{GCD}(a,n) = 1$. For the said sequence, we illustrate some properties and results useful to the computation of the inverse modulo.

3.1. Definitions and Main Results

Definition 1. *Given two positive integers* a, n *with* $a < n$ *and* $GCD(a,n) = 1$, *define the sequence* $(z_j)_{j \geq 0}$ *as follows:*

$$z_j = z_{j-1} + a\beta_j - n \qquad (j \geq 1), \tag{3}$$

starting from $z_0 = 0$, *with*

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

with M *being the ceiling part of* $m := n/a$.

Observe that β_j's represent the (ceiling) difference between n and z_j relative to a.

Next, the Proposition gives an explicit expression for the sequence $(z_j)_{j\geq 0}$.

Proposition 1. *The explicit form of the sequence* $(z_j)_{j\geq 0}$ *defined in* (3) *is given by*

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

Proof. The proof is immediate, indeed starting from definition (3), one has

$$z_1 = a\beta_1 - n,$$

$$z_2 = z_1 + a\beta_2 - n = a(\beta_1+\beta_2) - 2n,$$

$$\vdots$$

$$z_j = z_{j-1} + a\beta_j - n = a\sum_{h=1}^{j}\beta_h - jn.$$

□

The following Proposition gives an explicit, and more convenient, expression for the sequence $(\beta_j)_{j\geq 1}$.

Proposition 2. *Let* $(\beta_j)_{j\geq 1}$ *be the sequence defined in* (4). *For any* $j \geq 1$, *it holds that*

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

with $m = n/a$. *Moreover,*

$$\sum_{h=1}^{j}\beta_h = \lfloor jm\rfloor + 1. \tag{7}$$

Proof. First of all, observe that (6) implies that the partial sum is (7), since

$$\sum_{h=1}^{j}\beta_h = \sum_{h=1}^{j}\left(\lfloor hm\rfloor - \lfloor (h-1)m\rfloor\right) =$$

$$\lfloor jm\rfloor - \lfloor (j-1)m\rfloor + \lfloor (j-1)m\rfloor - \lfloor (j-2)m\rfloor + \ldots + \lfloor 3m\rfloor - \lfloor 2m\rfloor + \lfloor 2m\rfloor - \lfloor m\rfloor + M =$$

$$\lfloor jm\rfloor - \lfloor m\rfloor + M = \lfloor jm\rfloor + 1,$$

being $\lfloor m\rfloor = M - 1$.

Formula (6) can be proved by induction on j. Indeed, if $j = 2$, by relation (3), one has

$$\beta_2 = \left\lfloor \frac{n - z_1}{a}\right\rfloor + 1 = \left\lfloor \frac{n - aM + n}{a}\right\rfloor + 1 = \lfloor 2m\rfloor - M + 1 = \lfloor 2m\rfloor - \lfloor m\rfloor.$$

Now, if (6) holds true up to the index $(j-1)$, then, by relations (5) and (7), it is easy to see that

$$\beta_j = \left\lfloor \frac{n - z_{j-1}}{a}\right\rfloor + 1 = \left\lfloor \frac{n - a\sum_{h=1}^{j-1}\beta_h + (j-1)n}{a}\right\rfloor + 1 = \lfloor jm\rfloor - \lfloor (j-1)m\rfloor - 1 + 1 =$$

$$\lfloor jm\rfloor - \lfloor (j-1)m\rfloor.$$

□

Corollary 1. *The sequence* $(z_j)_{j\geq 0}$ *defined in* (3) *can be rewritten as*

$$z_j = a\big(\lfloor jm\rfloor + 1\big) - jn \qquad (j \geq 1), \tag{8}$$

with $z_0 = 0$.

Proof. The assertion results by combining relations (5) and (7). □

Now, we are able to state the main results of this section.

Theorem 1. *Consider two positive integers* a, n *with* $a < n$ *and* $GCD(a, n) = 1$. *Let* $(z_j)_{j\geq 0}$ *be the sequence defined in* (3) *and* $i \geq 1$ *the index such that* $z_i = 1$. *Then, due to* (8), *the inverse of* a *modulo* n *is given by*

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

with $m = n/a$.

Proof. Since $\mathrm{GCD}(a, n) = 1$, from the Bézouts's identity, there exists an index $i \geq 1$ such that $z_i = 1$. Indeed, without loss of generality, there exist a pair of positive integers g, i such that

$$1 = ga - in.$$

Fixing i, from the above equation, we obtain

$$g = im + \frac{1}{a} = \lfloor im \rfloor + \varphi_i + \frac{1}{a} = \lfloor im \rfloor + 1, \tag{10}$$

where we denote by

$$\varphi_j = jm - \lfloor jm \rfloor \qquad (j \geq 0), \tag{11}$$

the fractional part function of jm. In particular, the last equality of (10) holds true because both g and $\lfloor im \rfloor$ are positive integers. Thus, as φ_i and $\frac{1}{a}$ belong to $(0, 1)$ we can say that $\varphi_i + \frac{1}{a}$ must be equal to 1. Hence,

$$1 = a\big(\lfloor im \rfloor + 1\big) - in,$$

and, more specifically,

$$a\big(\lfloor im \rfloor + 1\big) \equiv 1 \pmod{n},$$

which implies

$$a^{-1} \equiv \lfloor im \rfloor + 1 \pmod{n},$$

where the last equality comes from Proposition 2. Finally, notice that $\lfloor im \rfloor + 1 < n$ (see Corollary 2) and this concludes the proof. □

3.2. Properties of the Sequence z_j

To better understand the nature of the sequence $(z_j)_{j\geq 0}$, we illustrate the following properties.

Proposition 3. *The sequence* $(z_j)_{j\geq 1}$ *defined in* (3) *is periodic with a period equal to* a.

Proof. For any $j \geq 1$, we have to prove that $z_{j+a} = z_j$. Let us proceed by induction on j. If $j = 1$,

$$z_{1+a} = a(\lfloor (1 + a)m \rfloor + 1) - (1 + a)n = aM + an - n - an = z_1.$$

Now, if the assertion holds true for $(j - 1)$, from relation (3), we may write

$$z_{j+a} = z_{(j-1)+a} + a\beta_{j+a} - n = z_{j-1} + a\big(\lfloor (j + a)m \rfloor - \lfloor (j + a - 1)m \rfloor\big) - n,$$

where

$$\lfloor (j + a)m \rfloor - \lfloor (j + a - 1)m \rfloor = \lfloor jm + n \rfloor - \lfloor (j - 1)m + n \rfloor = \lfloor jm \rfloor - \lfloor (j - 1)m \rfloor.$$

Therefore, we get

$$z_{j+a} = z_{j-1} + a\big(\lfloor jm \rfloor - \lfloor (j-1)m \rfloor\big) - n = z_{j-1} + a\beta_j - n = z_j.$$

□

Corollary 2. *The sequence* $(z_j)_{j\geq 0}$ *defined in* (3) *is less than* n*. In particular, the modular inverse defined in* (9) *is also less than* n*.*

Proof. From Proposition 3, we have to prove that $z_j < n$ for any $0 \leq j \leq a$. For this purpose, distinguish the following three cases:

(i) The $j = 0$ is trivial.

(ii) If $0 < j < a$, we have

$$z_j = a(jm - \varphi_j + 1) - jn = a(1 - \varphi_j) < a < n.$$

(iii) The case $j = a$ may be proved analogously to ii).

Finally, it is clear that the modular inverse defined in (9), i.e., $\lfloor im \rfloor + 1$, is less than n since $i \leq a - 1$. □

To see what was observed up to now, we shall consider a numerical example.

Example 4. *Choose* $a = 131$ *and* $n = 621$*. Obviously,* $GCD(a, n) = 1$ *that guarantees the existence of* a^{-1}*. It is obtained by* $i = 27$ *and*

$$131^{-1} = \left\lfloor 27 \cdot \frac{621}{131} \right\rfloor + 1 = 128,$$

modulo 621*. Figure 1 shows the behavior of the sequence* $(z_j)_{j\geq 0}$*. In particular, the blue line denotes the series when* $1 \leq j \leq 203$*, while those colored in red represent the entire sequence from two consecutive unitary* z_i*'s (circled in red), i.e.,* $i = 27$ *and* $i = 158$*. As proved, the series* $(z_j)_{j\geq 1}$ *is periodic with a period equal to* 131 *and any value less than* 621*.*

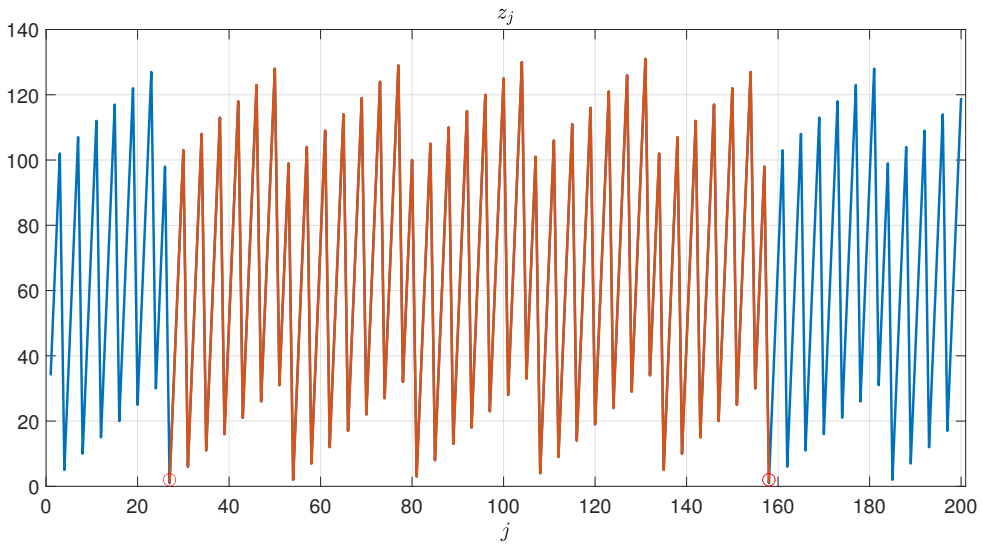

Figure 1. Sequence $(z_j)_{1\leq j\leq 203}$ when $a = 131$ and $n = 621$. The red line highlights the entire sequence $(z_j)_{j\geq 1}$ between two consecutive unitary z_i's (red circles).

3.3. Limitations and Future Challenges

A limitation of the proposed approach is that we have left the problem of determining the index i unsolved. In fact, by virtue of Theorem 1, we need to compute $z_i = 1$, such that

$$a\big(\lfloor im \rfloor + 1\big) - in = 1. \tag{12}$$

Observe that $\lfloor im \rfloor = im - \varphi_i$, where φ_i is defined by (11), and can be easily computed, as follows.

Proposition 4. *Let i be the solution of Equation (12); then, one has*

$$\varphi_i = \frac{a-1}{a}. \tag{13}$$

Proof. Equation (12) gives

$$1 = a\big(\lfloor im \rfloor + 1\big) - in = a\big(im - \varphi_i + 1\big) - in = a(1 - \varphi_i),$$

which implies Formula (13). □

Example 5. *With reference to Example 4, we have $m = 4.7405$ and $i = 27$. By computing φ_i directly from i, we obtain the value 0.9924, which coincides with that given by the a priori Formula (13).*

The knowledge of φ_i jointly with the periodicity information given by Proposition 4 suggests to solve the problem (12) by the simple algorithm described in Table 4.

Table 4. Pseudocode of a simple algorithm to solve (12).

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

Notice that the complexity of the algorithm just shown in Table 4 is $\mathcal{O}(a)$. Therefore, the above procedure is more convenient when $a << n$ (e.g., $a < \ln n$). In addition, when a is close to n, the algorithm in Table 4 is still better compared to the naive algorithm in Table 1 (since it involves simple multiplications instead of modular multiplications). Furthermore, Equation (9) represents a closed formula for the modular inverse, as does Equation (2), where the computational cost of the index i is $\mathcal{O}(a)$. This is less than $\mathcal{O}(n \ln n)$ of the Euler's phi function. These features are a clear advantage when n is large.

4. Conclusions

In this article, we have introduced the modulo operation and described the most common methods for computing the inverse modulo n. Hence, we have shown that, to solve the problem in Equation (12) through a closed formula, we need to investigate the properties of the sequence $(z_j)_{j\geq 0}$. The fact that the sequence $(z_j)_{j\geq 0}$ admits a simple explicit form which is periodic (for $j \geq 1$) helps us in understanding the features of $(z_j)_{j\geq 0}$. In particular, we have shown that the computational cost is $\mathcal{O}(a)$ versus $\mathcal{O}(n \ln n)$ of Euler's phi function. In terms of implementation, we suggest an algorithm for calculating a^{-1} using plain multiplications instead of modular multiplications. From a practical point of view, this approach is quite convenient because it has complexity $\mathcal{O}(a)$ compared to $\mathcal{O}(\ln n)$ of the extended Euclidean algorithm. This result is related to the characteristics of i, and, consequently, of a^{-1}. Next, research will focus on the determination of the index i such that $z_i = 1$.

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. and M.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: No applicable.

Informed Consent Statement: No applicable.

Data Availability Statement: No applicable.

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. U.S. Patent 4,405,829, 20 September 1983.
2. 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.
3. Mezher, A.E. Enhanced RSA cryptosystem based on multiplicity of public and private keys. *Int. J. Electr. Comput. Eng.* **2018**, *8*, 3949. [CrossRef]
4. 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 217; Springer: Berlin/Heidelberg, Germany, 2017; pp. 543–554.
5. 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]
6. 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]
7. 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]
8. Crandall, R.; Pomerance, C.B. *Prime Numbers: A Computational Perspective*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2006; Volume 182.
9. Rivest, R.L.; Shamir, A.; Adleman, L. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM* **1978**, *21*, 120–126. [CrossRef]
10. Verkhovsky, B. Overpass-Crossing Scheme for Digital Signature. In Proceedings of the International Conference on System Research, Informatics and Cybernetics, Baden-Baden, Germany, 30 July–4 August 2001; Volume 30.
11. ElGamal, T. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Trans. Inf. Theory* **1985**, *31*, 469–472. [CrossRef]
12. Rabin, M.O. *Digitalized Signatures and Public-Key Functions as Intractable as Factorization*; Technical Report; Massachusetts Inst of Tech Cambridge Lab for Computer Science: Cambridge, MA, USA, 1979.
13. Hoffstein, J.; Pipher, J.; Silverman, J.H.; Silverman, J.H. *An Introduction to Mathematical Cryptography*; Springer: Berlin/Heidelberg, Germany, 2008; Volume 1.
14. Verkhovsky, B. Enhanced Euclid Algorithm for Modular Multiplicative Inverse and Its Application in Cryptographic Protocols. *IJCNS* **2010**, *3*, 901–906. [CrossRef]
15. 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]
16. Yakymenko, I.; Kasianchuk, M.; Ivasiev, S.; Melnyk, A.; Nykolaichuk, Y.M. Realization of RSA cryptographic algorithm based on vector-module method of modular exponention. In Proceedings of the 2018 14th International Conference on Advanced Trends in Radioelecrtronics, Telecommunications and Computer Engineering (TCSET), Lviv-Slavske, Ukraine, 20–24 February 2018; pp. 550–554.
17. Ore, O. *Number Theory and Its History*; Dover Books on Mathematics Series; Dover: Mineola, New York, USA, 1988; ISBN 9780486656205.