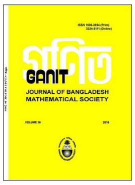
Available online at https://ganitjournal.bdmathsociety.org/

GANIT: Journal of Bangladesh Mathematical Society

GANIT *J. Bangladesh Math. Soc.* 44.2 (2024) 001–016
DOI: https://doi.org/10.3329/ganit.v44i2.78527

GANIT



Variants of RSA and Rabin Cryptosystems modulo $p^r q^s$

Md. Mohibul Hasan[1] and Sohana Jahan *[2]

[1,2] *Department of Mathematics, University of Dhaka, Bangladesh*

ABSTRACT

Cryptography is the technique to protect sensitive information from unauthorized persons by encryption. Cryptographers have invented various systems of cryptography to make it befitting. In the age of modern science, the use of mathematical theories has added a new dimension to cryptography. Prime factorization-based cryptography is widely used and effective. In 1977, Rivest, Shamir, and Adlerman proposed the first practical public key cryptosystem based on the prime factorization of large numbers known as the RSA cryptosystem. Later in 1979, Michael Oser Rabin developed a technique based on prime-factorization known as Rabin Cryptosystem. Several variants of these systems have been developed further by many famous mathematicians and computer scientists, aiming to increase security, reduce cost, time, and memory usage, and enhance overall performance. Tsuyoshi Takagi and Hugh C. Williams proposed such two famous variants. In this paper, we have first analyzed the traditional cryptosystems and existing variants. Identifying the security strength and field of improvement of these systems and their variants, we have proposed two new variants. The encryption-decryption techniques are described by employing them in several applications. The effectiveness of the proposed variants is demonstrated by a comparative analysis of these variants with others. Numerical experiments show that the proposed Rabin's variant performs almost the same as the base algorithm. However, the proposed variant of RSA algorithm reduces computational time significantly, approximately 91% reduction from traditional RSA and 90% reduction from multi-prime RSA.

Received: September 04, 2024 **Accepted:** October 13, 2024 **Published Online:** December 30, 2024

Keywords: Encryption; Decryption; Ctyptosystem; Integer factorization; Chinese Reminder Theorem.
AMS Subject Classifications 2024: 11T71 , 11A25, 94A60, 68P25.

1 Introduction

"Cryptography" refers to secure information and communication methods that are created using algorithms to create messages that are difficult to read [1, 2, 3, 4, 5]. It safeguards data and communication so that the information that is sent can be read by only the intended recipients. Cryptosystems are the deterministic algorithms utilized to create cryptographic keys, digital signatures, data privacy protection verification, online browsing, and private correspondence including emails and credit card transactions.
The word cryptography is derived from the Greek word *kryptos* meaning *hidden.* Encryption [6, 7] is the process of converting regular text into ciphertext and Decryption is the process of getting the original text back.

*Corresponding author: Sohana Jahan. *E-mail address:* sjahan.mat@du.ac.bd

In ancient times cryptography was based on some typical tools and techniques [8, 9] which were not difficult to break. Using mathematics in breaking these techniques made them vulnerable and cryptographers were supposed to take measures to keep their message secure. These leads cryptography to modernization day by day.

The theory of mathematics and computer science are major foundations of modern cryptography[10, 11, 12, 13]. Computational hardness is the premise that cryptographic methods are difficult for adversaries to crack in real-world scenarios [12, 14]. Although it is theoretically conceivable to breach a well-designed system, doing so is not practical in real life. These methods are referred to as "computationally secure" if they are well-designed. However, as computer technology progresses and theory evolves (such as improvements in integer factorization algorithms), these designs should be regularly reviewed and, if needed, modified. The finest theoretically breakable but computationally secure techniques are far easier to employ in practice than information-theoretically safe schemes, such as the one-time pad [15], which are presumably impossible to break even with infinite computer power.

1.1 Motivation

Cryptography involves a vast area of research. In the age of supercomputers, the most secure cryptography process depends on mathematical theories. Almost all the ancient [16] and analog cryptosystems are breakable at present. However, factoring large integers is still a difficult problem. So this particular type of cryptography can make the future [17, 18]. Much research has been done and is still ongoing [19, 20].

In 1978 Rivest-Shamir-Adleman introduced the foundation of public key cryptosystem [21]. This technique uses two large primes, an encryption key and a decryption key. The idea behind RSA cryptosystem was that though multiplying large prime numbers is simple, but factoring a large number into prime numbers is a great challenge. In 1979, Rabin introduced a similar technique using two primes of a particular class. In [22] Rabin used a fixed encryption key 2. The encryption process is quite similar to RSA but the decryption is different. Then in [23] Williams proposed another modification of RSA in 1980.

To reduce the computational time and overall cost of the RSA and Rabin scheme Takagi proposed a fast type of RSA in 1998. In [24] Takagi used two primes and one of them was taken multiple times. For fast computation, he used the concept of n-adic expansion in [25]. In integer factorization cryptosystems, the size of public and private keys is important in decreasing the cost and time of encryption and decryption. So this dynamic process needs a huge amount of research and proposing a technique that reduces the time and cost of encryption and decryption is much needed. So we have worked on the variants of RSA and Rabin's and proposed two variants that can reduce the size of encryption and decryption keys.

1.2 Our Contribution

In this research, we have worked on the prime factorization-based cryptosystems. We have worked on RSA and Rabin's and some of their variants. Our main contribution in this paper is

- We have proposed one variant of RSA modulo $p^r q^s$ to reduce the size of encryption and decryption keys as well as to reduce the cost and required time, where p, q are two primes and k, l are integers. The size of the decryption key got a remarkable reduction in our proposed variant compared with the original RSA and its existing variants.
- We have proposed a variant of Robin Cryptosystem with the same modulo as RSA. The proposed variant reduces the size of the decryption key. However, the computational time is almost the same as the traditional Robin and better than its variants Williams.

1.3 Paper Organization

The RSA and Rabin cryptosystems, which rely on integer factorization, are the primary subjects of this study. In the next section, we have discussed the basic concept of cryptography and the background study of cryptography. The mathematical theory behind integer factorization-based cryptography is discussed in the following section where we have discussed the textbook RSA and Rabin's. On the way, we have incorporated some most renowned variants of RSA and Rabin's. In section 3, we have proposed two variants, one of RSA and one of the Rabin's. We have applied our proposed approach and the existing approaches for the encryption and decryption of several messages and compared the computational time in section 4. We have concluded our article in section 5.

2 Background

In number theory, integer factorization means the process of breaking down a positive integer into the product of integers in number theory. Any positive integer larger than one is either prime or can be expressed as the product of primes. Though small composite numbers can be factored into by calculation, greater numbers require advanced factorization procedures [26, 27, 28, 29], especially utilizing computers. Whenever a factor arises, a prime factorization method usually checks to see if it is prime. There isn't an effective non-quantum integer factorization algorithm when the numbers are big enough. It hasn't been shown, nonetheless, that there isn't such an algorithm. A b-bit number n can be factored in O(b^k) time for any constant k. It should be noted that any method has not yet been developed to factor all integers in polynomial time, It is generally believed that such algorithms do not exist. Also, their presence or nonexistence has not been demonstrated. For this reason, the issue does not fall within class P (Possible). Though it hasn't been demonstrated, it is generally believed that even though the issue is obviously in class NP, it is not NP-complete.

2.1 RSA Cryptosystem

Rivest, Shamir, and Adlemans's RSA is the base of the modern asymmetric cryptosystem [13, 21, 30, 31, 32]. It is the first public key encryption. RSA cryptography encrypts messages using both public and private keys. To decrypt a message, the reverse of the encryption key is used. This quality is one of the reasons why RSA has become the most extensively used asymmetric algorithm [33, 34].

Algorithm 1: RSA
Key Generation : Let $n = pq$, Now define,

$$\kappa = \{(n, a) : GCD(a, \phi(n)) = 1\}.$$

where $\phi(n) = (p-1)(q-1)$ know as *Euler's Totient Function.* Select (n, e) and (n, d) from κ such that $ed \equiv 1 (\text{mod } \phi(n))$. Then (n, e) is the encryption key and (n, d) is the decryption key.

Encryption : For plaintext M, using the public key (n, e), calculate the encrypted text C by,

$$C = M^e \pmod{n}$$

Decryption: Use the decryption key $(n, d$ and compute the decrypted message.

$$M_1 = C^d \pmod{n}$$

Here, M_1 equals the plaintext M. The values (n, e) comprise the public key which is known to all, and the values (n, d) form the private key which is kept secretly. The RSA cryptosystem relies on the assumption that finding two big prime numbers is trivial; but computing the initial primes from the total or factoring is seen as impracticable owing to the time involved, even with today's supercomputers. This is because, the function $F : M \to C$ is a one-way trapdoor function because it can be computed quickly using the fast exponentiation method, but calculating its inverse, $F^{-1} : C \to M_1$, is challenging as it requires factoring n and computing $phi(n)$ for those who do not know the private key. Nonetheless, the computation of F^{-1} is as simple for those who know the private key.

Remark 1: The RSA trapdoor is made up of the four RSA parameters $\{d, p, q, \phi(n)\}$. Each of the four facts is equally significant. The RSA encryption is broken if one of them is known, as this divulges information about the remaining three. If RSA encryption is not utilized appropriately, it may be cracked without knowing about $\{d, p, q, \phi(n)\}$. Encrypting a single message using many exponents $e_1, e_2, ..., e_r$ allows anybody to retrieve the plaintext if the gcd is equal to 1. M can be obtained by anyone without factoring n or utilizing any of the trapdoor data $\{d, p, q, \phi(n)\}$. The proof is given for 2 exponents as follows:

Theorem 1. *If*

$$C_1 \equiv M^{e_1} \ (mod\ n)$$
$$C_2 \equiv M^{e_2} \ (mod\ n)$$

where $e_1 \neq e_2$ and $gcd(e_1, e_2) = 1$ then M can be recovered easily without knowledge of any secret key.

Proof : Since $\gcd(e_1, e_2) = 1$, then by the extended Euclid's algorithm $e_1x + e_2y = 1$ with $x, y \in Z$. Thus,

$$\begin{aligned} C_1^x C_2^y &\equiv (M^{e_1})^x (M^{e_2})^y \pmod{n} \\ &\equiv M^{e_1x+e_2y} \\ &\equiv M \pmod{n} \end{aligned}$$

Hence M is recovered.
So, for each given modulus, one should employ no more than one encryption exponent.

Remark 2: (Choice of e) Encryption can be done faster with a small exponent e, and decryption with a small exponent d. Of course, we can't choose both to be little since once one is chosen, the other is dictated by the congruence. This is not quite correct because $e = 1$ implies $d = 1$, implying that both d and e are tiny. Taking $e = 1$ is a terrible idea, as the plaintext and ciphertext are identical. We cannot use $e = 2$ because e must have multiplicative inverse and for that, it should be relative prime to $(p-1)(q-1)$. Thus, the least number for e is $e = 3$. Taking $e = 3$ is equally safe as taking a higher value of e.

2.1.1 Existing Variants of RSA

Being a highly valuable, computationally light, and extensively utilized public key cryptosystem, researchers are working on several variants of RSA algorithms to reduce the bit side of decryption key d.

Multi-prime RSA

Multiprime RSA was proposed by Collins, Hopkins, Langford, and Sabin in 1997 [35]. In this method, the authors choose n to be the product of $k > 2$ primes. This will speed up the encryption and decryption process. The algorithm is described below:

Algorithm 2: Multi-prime RSA
Key Generation: Let $n = p_1, p_2, ..., p_k$, $\phi(n) = (p_1 - 1) \cdot (p_2 - 1) \cdot ... \cdot (p_k - 1)$. As in the case of RSA e and d are such that $ed \equiv 1$ (mod $\phi(n)$.
Encryption: For the plaintext M compute the ciphertext C by

$$C \equiv M^e \pmod{n}$$

Decryption: Using the privet key (n, d) , the plaintext M can be retrieved by

$$M \equiv C^d \pmod{n}$$

Using the Chinese Remainder Theorem and parallel calculations (using conventional arithmetic), a ciphertext may be decrypted in a maximum of $\frac{3}{2r^3}(log_2(n))^3$ bit operations. This is the first advantage. It is shown that using the Chinese Remainder Theorem instead of single exponentiation speeds up the calculation three times in the case of two primes. Using three primes leads to an extra 1.9 speed improvement above the 2-prime case, for a total speed increase of around 5.7. Using four primes results in an overall speed improvement of around 8.9, which is 1.6 quicker than using three. The second benefit is space-related: by using the Chinese Remainder Theorem once again, the space needed for each decryption computation up to the last (recombining) step is reduced to $log_2 p_r$ space, where p_r is the greatest prime of the modulus. If all the primes are about $\frac{log_2(n)}{r}$ big (balanced primes), then the space required decreases with each additional prime added to the modulus.

Takagi Variant of RSA

Tsuyoshi Takagi [25] proposed an RSA-type cryptosystem that uses the modulo $n = p^k q$. This scheme enhances the performance of the RSA by reducing the size of the decryption key d as well as the decryption cost. The sizes of the secret primes p and q should be selected appropriately.

Algorithm 3: Takagi variant of RSA
Key Generation: Let $n = p^k q$ Determine e, d that satisfies $ed \equiv 1 \pmod{L\ }$ and $GCD(e, p) = GCD(e, L) = 1$ by computing $L = LCM(p-1, q-1)$. Then, the secret keys are (d, p, q, k), and the public keys are (e, n).

Encryption: For the plaintext M. ciphertext is calculated by,

$$C \equiv M^e \pmod{n}.$$

Decryption: With the secret key (d, p, q, k), solving $M_p \equiv M \pmod{p^k}$ and $M_q \equiv M \pmod{q}$ using Chinese Remainder Theorem (CRT) the plaintext M can be retrieved. In this case, M_p is calculated using the fast algorithm given in [25], and M_q is calculated using $M_q \equiv C^d \pmod{q}$.

2.2 Rabin Cryptosystem

As discussed in remark 2, the smallest possible value of e in RSA is 3. Micheal Oser Rabin [22] proposed a scheme based on $e = 2$. The advantage of the Rabin trapdoor function is that, unlike the RSA trapdoor function, its inversion has a mathematical proof [22] that indicates it is as difficult to solve as factoring integers. Rabin's algorithm is given in as follows:

Algorithm 4: Rabin
Key Generation : Let $n = pq$, where p and q satisfies

$$p \equiv q \equiv 3 \pmod{4}$$

Encryption : The encryption is done by

$$C \equiv M^2 \pmod{n}$$

Decryption : For the decryption, a system of congruences is obtained

$$M_p \equiv \sqrt{C}(modp)$$
$$M_q \equiv \sqrt{C}(modq)$$

The above system of congruences can be solved using the CRT to get the four solutions $\pm M_p, \pm M_q$ and one of these will be the actual plaintext M.
The Rabin's function disadvantage is that each output can be created by any one of the four potential inputs. Thus the decryption becomes more complicated to establish which of the four possible inputs was the real plaintext. Decrypting creates three false outputs in addition to the correct one. This is the Rabin cryptosystem's fundamental shortcoming and one of the reasons it has not found broad practical application. It is not difficult to predict whether the plaintext is intended to represent a text message; but, if the plaintext is intended to represent a numerical value, a disambiguation approach must be used to manage the situation. To get around this issue, you can use plaintexts with certain structures or add padding. Blum and Williams proposed a solution to the ambiguity of inversion: the two primes that are utilized can only be primes that are equivalent to 3 modulo 4. Additionally, the domain of the squaring is limited to the set of quadratic residues. These constraints remove the uncertainty by creating a trapdoor permutation from the squaring function.[36] Even though Rabin cryptography is not as commonly used as some other public-key algorithms, such as RSA, it is nonetheless an appealing subject for research in the field of cryptography and provides the broader context of data security and secure communication.

2.2.1 Williams Cryptosystem

In 1980, Hug William proposed a variant of Rabin's cryptosystem called Rabin's M^2 system. The algorithm proposed by William [23] is as follows

Algorithm 5: Williams

Key Generation : Let $n = pq$, where p and q are primes satisfying,

$$p \equiv 3 \pmod 8$$
$$q \equiv 7 \pmod 8$$

We select the value e such that $(e, \lambda(n)) = 1$, where $\lambda(n) = \text{LCM}(p-1, q-1\ )$

Encryption: Suppose M be a positive integer such that $2(2M+1) < n$ when $(2M+1|n) = -1$ and $4(2M+1) < n$ when $(2M+1|n) = 1$. Assume K be the set of all such M. For all $M \in K$, define

$$N = E_1(M) =$$

$$\begin{cases} 4(2M+1), & \text{when } (2M+1 \mid n) = 1 \\ 2(2M+1), & \text{when } (2M+1 \mid n) = -1 \end{cases} \tag{2.1}$$

$$\text{Then, } C = E_2(N) \equiv N^{2e} \pmod n$$

Decryption : For the decryption, $L = D_2(C) \equiv C^d (mod\ n)$

$$\text{Then, } M = D_1(L) =$$

$$\begin{cases} \frac{\frac{L}{4}-1}{2}, & \text{when } L \equiv 0 \pmod 4 \\ \frac{\frac{n-L}{4}-1}{2}, & \text{when } L \equiv 1 \pmod 4 \\ \frac{\frac{L}{2}-1}{2}, & \text{when } L \equiv 2 \pmod 4 \\ \frac{\frac{n-L}{2}-1}{2}, & \text{when } L \equiv 3 \pmod 4 \end{cases} \tag{2.2}$$

It should be noted that Williams encryption requires several calculations for decryption.

3 Proposed Approach

3.1 Proposed Variant of RSA

Let the moduli of the be RSA system $N = A^x \cdot B^y$ where, A & B are primes and the powers x, y are large. This will enhance memory usage. The algorithm of the scheme is described as follows:

Algorithm 6: Proposed variant of RSA

Key Generation: We choose two primes A and B and assume the moduli $N = A^x \cdot B^y$. We compute $\lambda = \lambda(N) = LCM((A-1)\cdot(B-1))$ and assume the encryption key e such that GCD$(e, A) = GCD$(e,B)=GCD$(e, \lambda) = 1$. Then we calculate d that satisfies $ed \equiv 1 \pmod{\lambda}$.
Now we take,

$$d_A \equiv d \pmod{(A-1)}$$
$$d_B \equiv d \pmod{(B-1)}$$

Thus we have the public key (N, e) and the privet key is (A, B, d_A, d_B, x, y).
Encryption: There is no difference in encryption with the original RSA. Let the plaintext be converted to an integer $1 \leq M \leq N$ which is necessarily coprime to N. We use the public key (N,e) and compute the ciphertext C as:

$$C \equiv M^e \pmod{N}$$

Decryption: We first decrypt $M_A \equiv C^d \pmod{A^x}$ and $M_B \equiv C^d \pmod{B^y}$ using the secret key (A, B, d_A, d_B, x, y). Then using CRT M can be recovered.

Here, M_A and M_B are computed by the fast algorithm using n-adic expansion described in [24]. For this, first we consider the $A-adic$ and $B-adic$ expansion of M_A and M_B respectively,

$$M_A = K_0 + AK_1 + A^2K_2 + ... + A^{x-1}K_{x-1} \pmod{A^x}$$
$$M_B = L_0 + BL_1 + B^2L_2 + ... + B^{y-1}L_{y-1} \pmod{B^y}.$$

As discussed in [24], at first K_0 and L_0 are calculated by ,

$$K_0 \equiv C^{d_A} \pmod{A}$$
$$L_0 \equiv C^{d_B} \pmod{B}$$

Then the blocks $K_1, K_2, ..., K_{x-1}$ and $L_1, L_2, ..., L_{y-1}$ can be decrypted by the following procedure.
Define a function $F_i(R_0, R_1, ..., R_i)$ as follows:

$$F_i(R_0, R_1, ..., R_i) = (R_0 + AR_1 + ... + A^iR_i)^e,$$

where $i = 0, 1, ..., x-1$ and $F_{x-1} = (R_0 + AR_1 + ... + A^{x-1}R_{x-1})^e$ is the function which is same as that encrypts the plaintext M_A. Reducing modulo A^{i+1},

we get,

$$F_i(R_0, R_1, ..., R_i) = F_{i-1} + A^iG_{i-1}R_i \pmod{A^{i+1}},$$

where $F_{i-1} = (R_0 + AR_1 + ... + A^{i-1}R_{i-1})^e$, $G_{i-1} = e(R_0 + AR_1 + ... + A^{i-1}R_{i-1})^{e-1}$. The values of $K_1, K_2, ..., K_{x-1}$ can be successively calculated by using the following relationship recursively. For $i = 1$, the following linear equation of R_1 has a solution of K_1:

$$C \equiv F_0(K_0) + AG_0(K_0)R_1 \pmod{A^2}.$$

Using the values of $K_1, K_2, ..., K_{i-1}$, F_{i-1}=$F_{i-1}(K_0, ..., K_{i-1})$ and G_{i-1}= $G_{i-1}(K_0, ..., K_{i-1})$ is obtained. The following linear equation of R_i is then solved by K_i:

$$C \equiv F_{i-1} + A^iG_{i-1}R_i \pmod{A^{i+1}}.$$

Because $GCD(K_0, A) = GCD(e, A) = 1$, it is evident that $(G_{i-l}, A) = 1$, allowing us to decrypt K_i uniquely. We may calculate $K_0, K_1, ..., K_{x-1}$, and then we can calculate $M_A \pmod{A^x}$. Similarly, we may calculate $L_0, L_1, ..., L_{y-1}$, and then we can calculate $M_B \pmod{B^y}$. Lastly, using the CRT, $M_A \pmod{A^k}$, and $M_B \pmod{B}$ values, the plaintext $M \pmod{A^xB^y}$ is also calculated.
Notably, we can evaluate $K_0, K_1, ..., K_{k-1}$ without using the secret exponent d as $d_A \equiv d \pmod{A-1}$ and $d_B \equiv d \pmod{B-1}$ hold, then $C^d \equiv C^{d_A} \pmod{A}$ and $C^d \equiv C^{d_B} \pmod{B}$ hold as well.

Problem1: Consider the plaintext message $M = 102908$. The message can be encrypted by using the proposed algorithm as follows:
Key Production: Choose the primes $A = 29$ and $B = 37$ with the exponents $x = 3, y = 5$.Then the moduli N becomes,

$$N = A^x \cdot B^y = 29^3 \cdot 37^5 = 1691229767273$$

Also,

$$\lambda(N) = LCM((A-1),(B-1)) = LCM(28,36) = 252$$

The encryption exponent e can be chosen as $e = 5$ since 5 is coprime with 252. Then, the solution of the congruence

$$5d \equiv 1 \pmod{252}$$

gives $d = 101$. Thus

$$\begin{aligned} d_A &\equiv d \pmod{(A-1)} \\ &\equiv 101 \pmod{28} \equiv 17 \pmod{28} \\ d_B &\equiv d \pmod{(B-1)} \\ &\equiv 101 \pmod{36} \equiv 29 \pmod{36} \end{aligned}$$

Hence the public key is $(1691229767273, 5)$ and the secret key is $(29, 37, 17, 29)$.
Encryption: At this step, take the plaintext $M = 102908$ and calculate the ciphertext C by,

$$\begin{aligned} C &\equiv M^e \pmod{N} \\ &\equiv 102908^5 \pmod{1691229767273} \\ &\equiv 1409436177262 \pmod{1691229767273} \end{aligned}$$

Now message 1409436177262 is sent.

Decryption: To decrypt the message find M_A & M_B such that,

$$\begin{aligned} M_A &\equiv 1409436177262^{101} \pmod{29^3} \\ M_B &\equiv 1409436177262^{101} \pmod{37^5} \end{aligned}$$

Using n-adic expansion,

$$\begin{gathered} M_A = K_0 + 29K_1 + 29^2 K_2 \pmod{29^3} \\ M_B = L_0 + 37L_1 + 37^2 L_2 + 37^3 L_3 + 37^4 L_4 \pmod{37^5}. \end{gathered}$$

using the procedures explained in above we have $K_0 = 16, K_1 = 10, K_2 = 6$ which gives $M_A = 5352$ and also $L_0 = 11, L_1 = 6, L_2 = 1, L_3 = 2, L_4 = 0$, then $M_B = 102908$. Then applying the CRT on the following congruences the original text M is,

$$\begin{aligned} Z &\equiv 5352 \pmod{29^3} \\ Z &\equiv 102908 \pmod{37^5} \end{aligned}$$

Finally, the decrypted message $Z = 102908$ is the same as the plaintext M.

3.2 Proposed Variant of Rabin Cryptosystem

In this section, we propose another variant of Rabin's which is slightly different from the previous one. We propose the moduli of the form

$$N = A^x B^y$$

where $A\&B$ are our primes such that $p \equiv 3 \pmod 4$ and $q \equiv 3 \pmod 4$. There remain only four square roots:- two square roots modulo A^x and two square roots modulo B^y. Using Hensel lifting these correspond to two square roots modulo A and two square roots modulo B respectively. As a result, we can employ precisely the same kind of redundancy schemes as conventional Rabin. We can expedite the computation of square roots using Hensel lifting [37] and the CRT. Thus, we expect a benefit to using Rabin this way. The proposed scheme:

Algorithm 7: Proposed variant of Rabin

Key Generation: First, we choose e two primes A, B, and then let $N = A^x B^y$. N is the public key and the primes A and B are kept secret.

Encryption: Assume the plaintext be M. We will calculate the ciphertext by

$$C \equiv M^2 \pmod N$$

Decryption: Let M_A and M_B be the solution of

$$X^2 \equiv C \pmod{A^x} \text{ and}$$
$$X^2 \equiv C \pmod{B^y}$$

respectively. After calculating M_A and M_B, we use CRT to calculate M because,

$$M_A \equiv \sqrt{C} \pmod{A^x}$$
$$M_B \equiv \sqrt{C} \pmod{B^y}$$
$$M \equiv \sqrt{C} \pmod{A^x B^y}$$

Now our main concern is to find M_A and M_B. So let the $A - adic$ and $B - adic$ expansion of M_A and M_B be

$$M_A = K_0 + AK_1 + A^2K_2 + ... + A^{x-1}K_{x-1} (\text{mod } A^x)$$
$$M_B = L_0 + BL_1 + B^2L_2 + ... + B^{y-1}L_{y-1} \pmod{B^y}$$

respectively. The first blocks K_0 and L_0 are the solutions of

$$X^2 \equiv C \pmod A$$
$$X^2 \equiv C \pmod B$$

respectively. Then we try to decrypt the other blocks $K_0, K_1, ..., K_{x-1}$ and $L_0, L_1, ..., L_{y-1}$. This can be done as we showed in the previous section. To find the block K_1 consider the linear equation modulo A^2,

$${K_0}^2 + 2AK_0X \equiv C \pmod{A^2}$$

After calculating $K_0, K_1, ..., K_{i-1}$, K_i can be decrypted uniquely by solving the linear equation ,

$$(K_0 + K_1 + ... + K_{i-1})^2 + 2A^i(K_0 + K_1 + ... + K_{i-1})X = C \pmod{A^{i+1}}$$

Thus, all plaintext blocks $K_0, K_1, K_2, ..., K_{x-1}$ can be decrypted. Similarly, we can find the plaintext blocks $L_0, L_1, L_2, ..., L_{y-1}$. Finally, we have got the plaintexts M_A and M_B which gives us the original plaintext M. The disadvantage of this scheme is that it gives results after deciphering. One of the four results is the original plaintext message. We have to add some redundancy schemes for unique decryption. It is as impossible to decipher the system's plaintext as it is to break factoring or the original RSA. So there is no concern about the security of this scheme. Also, the encryption and decryption are very fast. All the formulas for encryption-decryption of each of the discussed algorithms are given in table 3.1

Method	Variant	The Moduli N	The keys	Ecryption	Decryption
RSA	Traditional	$n = pq$	$ed = 1(\text{mod}(p-1)(q-1)$	$C = M^e \pmod n$	$M_1 = C^d \pmod n$
	Mulitiprime	$n = p_1 \dots p_k$	$ed = 1(\text{mod}(p_1-1)\dots(p_k-1)$	$C = M^e \pmod n$	$M_1 = C^d \pmod n$
	Takagi	$n = p^k q$	$ed = 1(\text{mod } LCM(p-1, q-1)$	$C = M^e \pmod n$	$M_p = M \pmod {p^k}$, $M_q = M \pmod q$
	Proposed	$n = A^x B^y$	$ed = 1(\text{mod } LCM(A-1, B-1)$	$C = M^e \pmod n$	$M_A = M \pmod {A^x}$, $M_B = M \pmod {B^y}$
Rabin	Traditional	$n = pq$	$p \equiv q \equiv 3 \pmod 4$	$C = M^2 \pmod n$	$M_p = \sqrt{C}(\text{mod } p)$, $M_q = \sqrt{C}(\text{mod } q)$
	Williams	$n = pq$	$p \equiv 3(\text{mod } 8),\ q \equiv 7(\text{mod } 8)$	$C \equiv N^{2e}(\text{mod} n)$	Decryption in Algorithm 5
	Proposed	$n = A^x B^y$	$A \equiv B \equiv 3 \pmod 4$	$C = M^2 \pmod n$	$M_A = \sqrt{C}(\text{mod } A^x)$, $M_B = \sqrt{C}(\text{mod } B^y)$

Table 3.1: Table of Formulas

Problem 2: Suppose a plaintext message $M = 10829$ is to be sent in a way such that the message is unreadable to anyone except the receiver. The sender can use our proposed variant of Rabin to encode his message and send it to the receiver.

Key Production: The sender can choose the primes $A = 31$ and $B = 33$ with the exponents $x = 3, y = 2$. Then the moduli N becomes,

$$N = A^x \cdot B^y = 31^3 \cdot 23^2 = 15759439$$

Encryption: To encrypt the message, compute

$$\begin{aligned} C &\equiv M^2 \pmod n \\ &\equiv 10829^2 \pmod{15759439} \\ &\equiv 6951168 \pmod{15759439} \end{aligned}$$

Decryption: Now for decryption we use n-adic expansion and compute,

$$\begin{cases} M_A \equiv \pm 10829(mod(31^3)) \\ M_B \equiv \pm 249(mod(23^2)) \end{cases}$$

This leads to 4 systems of congruences

$$\begin{cases} M \equiv 10829(mod 31^3) \\ M \equiv 249(mod(23^2)) \end{cases} \tag{3.1}$$

$$\begin{cases} M \equiv -10829(mod 31^3) \\ M \equiv 249(mod(23^2)) \end{cases} \tag{3.2}$$

$$\begin{cases} M \equiv -10829(mod 31^3) \\ M \equiv -249(mod(23^2) \end{cases} \tag{3.3}$$

$$\begin{cases} (M \equiv 10829)(mod 31^3) \\ M \equiv -249(mod(23^2) \end{cases} \tag{3.4}$$

These systems of congruences on solving using the CRT give, M=10829, M=1776631, M=15748610, and M=1398208 respectively. Among these 4 values, M=10829 is the expected original message.

4 Numerical Results

In this section, we have tested traditional RSA, traditional Rabin, and all of the variants with different sizes of the moduli n. We have generated several 2 to 16-digit numbers and compared the total encryption-decryption time of different cryptosystems. We ran 10-40 different instances for each of the number of different lengths and reported some representatives of the results. It is observed that for each of the algorithms, the required time for encryption-decryption increases with the bit length of moduli n. The results are shown in tables as well as using histogram graphs for each of the cryptosystems.

4.1 Technical Specification

All the tests were carried out using MATLAB 24.2.0.2729000 (R2024b). The technical specifications of the computer used for the computation is presented in Table.

Components	Specification
Ram	4 GB
Processor	Intel(R), Core(TM) i5-4210U CPU @ 1.70GHz 2.40 GHz
System type	64-bit operating system, x64- based processor

4.2 RSA and Its Variants

First, we compare the time taken to implement the traditional RSA and two of its variants - Multiprime RSA and Our proposed Scheme that we described in the earliest section.

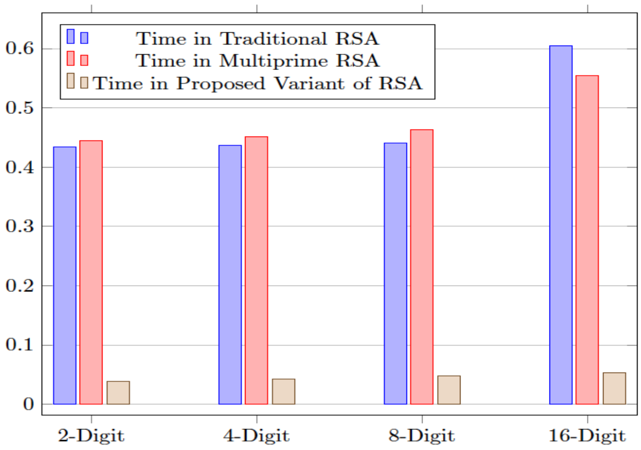


Figure 4.1: Time in Traditional RSA, Time in Multiprime RSA, Time in Proposed Variant of RSA

Variants	Length of N (number of digits)	The Moduli N	Ecryption - Decryption Time (s)
RSA	2	15	0.432721
	2	35	0.431319
	2	65	0.432187
	2	51	0.431129
	4	4757	0.436528
	4	1147	0.436312
	4	1517	0.436356
	4	3127	0.436510
	8	39601813	0.441366
	8	10017221	0.440123
	8	10169717	0.439871
	8	13859099	0.440019
	16	3941455467744143	0.604202
	16	1000000166758057	0.598356
	16	2505004802300129	0.602367
	16	2505023420943677	0.605128
Multiprime	2	70	0.445105
	4	5083	0.451680
	4	2387	0.448907
	4	1005	0.448231
	4	3965	0.450091
	8	10681031	0.462916
	8	29884301	0.467865
	8	68719771	0.466719
	8	99677881	0.468902
	16	3376192575450451	0.553894
	16	2881479796602431	0.551786
	16	5123911108009009	0.555723
	16	5782119654252079	0.556003
Proposed Variant	2	63	0.038185
	2	75	0.038713
	2	99	0.038765
	4	3025	0.043276
	4	1573	0.041762
	4	5491	0.043519
	4	2783	0.042987
	8	42599173	0.047758
	8	13409477	0.045823
	8	14386373	0.045811
	8	19915757	0.046109
	16	1173607064999641	0.0529195
	16	1274795891413019	0.0530137
	16	1835707307416979	0.0552131
	16	4546045013659463	0.0570816

Table 4.1: Time comparison between variants of RSA

Table 4.1 shows that the proposed algorithm reduces computational time significantly. Approximately 91% reduction from RSA and 90% reduction from multi-prime RSA. For Takagi's algorithm with $y = 1$ in our proposed variant, the encryption-decryption time lies between the time taken by multi-prime RSA and the proposed variant. The results are not included in this article. It can be seen from Table 4.1 as well as from figure 4.1 that as the value of n gets large, the computational time of the proposed variant differs significantly from other variants. For higher values of n, exploring the performance of these approaches leaves a scope for further study.

4.3 Rabin and its Variants

A similar comparison for the Textbook Rabin and two of its variants- Williams Cryptosystem and Our Proposed Scheme are shown in table

Variants	Length of N (number of digits)	The Moduli N	Ecryption - Decryption Time (s)
Rabin	2	21	0.469409
	2	77	0.469821
	4	3953	0.474206
	4	4757	0.474228
	4	5609	0.475006
	4	8549	0.475212
	8	32455613	0.499597
	8	64400429	0.501211
	8	45037457	0.499721
	8	75916333	0.501503
	16	2500005300002773	0.540895
	16	6400004000000621	0.541798
	16	3600002040000253	0.541005
	16	4900004900001161	0.541276
Williams	2	77	0.481578
	4	4757	0.497845
	4	4189	0.497586
	4	8549	0.498516
	4	2021	0.493982
	8	46416869	0.503511
	8	75916333	0.503871
	8	45927533	0.503523
	8	94887077	0.503897
	16	4609992424453861	0.531504
	16	4900009100002781	0.532812
	16	3600002040000253	0.529316
	16	7569042630047029	0.531672
Proposed Variant	2	63	0.468358
	4	9317	0.475541
	4	1127	0.472167
	4	2783	0.4726179
	4	8303	0.477213
	8	39651821	0.496770
	8	75355727	0.497129
	8	35806223	0.496113
	8	31885367	0.495678
	16	1776339597932567	0.531652
	16	4546045013659463	0.531802
	16	2557173760459463	0.531913
	16	5753519498148047	0.5320512

Table 4.2: Time comparison between variants of Rabins cryptosystem

In terms of computation, the proposed variant shows similar performance to the traditional Rabin and slightly better than its other variant Williams which can be observed in table 4.2. We have shown the results for the number n up to 16 digits due to the limitation of using a personal computer. The use of supercomputers for higher digit numbers could be interesting to explore.

5 Conclusion

The application of mathematical theories and the use of modern computers made cryptosystems much more secure at present. The difficulty of finding the prime factors of large composite numbers is such a mathematical theory. If sufficient processing power is available, it is now feasible to factor numbers with more than 250 decimal digits, and factoring a hundred decimal digits is a simple task thanks to recent significant advancements in the most well-known integer factorization algorithms. Though none of the algorithms run in polynomial time, the problem of integer factorization still looks challenging from a theoretical and practical standpoint—especially for numbers with more than 100 decimal digits.[27]

Although it is not flawless, cryptography is a strong tool for secure communication. A cryptographic system can be attacked in a variety of ways, and new attacks are continually being found. It is now a crucial component of cybersecurity, used to safeguard the integrity of data and guard against illegal access. Cryptography will continue to be essential for protecting data as the world becomes more computerized.
With a powerful enough quantum computer, the mathematical operations that the majority of conventional cryptography algorithms rely on could be broken. Many researchers are working on cryptanalysis of different cryptosystems [38]. As with the RSA, the proposed variant can be scrutinized for vulnerabilities. Different attacks on this variant can be implemented to test its strength compared to that of RSA and its other variants.

In this paper, we have discussed integer factorization-based cryptosystems such as the RSA and Rabin with some variants. Two new variants of RSA and Rabin's are proposed. The existing algorithms and the proposed algorithms are tested with different sizes of the moduli n. Several 2 to 16-digit numbers are generated and the performances are compared in terms of the total encryption-decryption time of different cryptosystems. Numerical experiments show that the proposed Rabin's variant per- forms almost the same as the base algorithm, whereas the proposed RSA variant outperforms the existing approaches and reduces computational time significantly, with approximately 91% reduction from traditional RSA and 90% reduction from multi-prime RSA.

References

[1] J. Hoffstein, J. Pipher, and J. Silverman, *An Introduction to Mathematical Cryptography.* 01 (2014).

[2] S. Yan, "Computational number theory and modern cryptography," *Computational Number Theory and Modern Cryptography*, 12 (2012).

[3] N. Sharma, P. Prabhjot, and H. Kaur, "A review of information security using cryptography technique," *International Journal of Advanced Research in Computer Science*, vol. 8, pp. 323–326, (2017).

[4] A. Mohammed and N. Varol, "A review paper on cryptography," pp. 1–6, 06 (2019).

[5] C. Paar and J. Pelzl, *Understanding Cryptography: A Textbook for Students and Practitioners.* Springer Berlin Heidelberg, (2009).

[6] R. Raimondo, "Encryption," 10 (2017).

[7] M. Wills, *Cryptography*, ch. 7, pp. 297–370. John Wiley Sons, Ltd, (2019).

[8] J. Schneider, "A brief history of cryptography: Sending secret messages throughout time," 01 (2024).

[9] H. Sidhpurwal, "A brief history of cryptography," 01 (2023).

[10] F. Maqsood, M. Ahmed, M. Mumtaz, and M. Shah, "Cryptography: A comparative analysis for modern techniques," *International Journal of Advanced Computer Science and Applications*, vol. 8, 01 (2017).

[11] A. Sidhu, "Analyzing modern cryptography techniques and reviewing their timeline (2023)," 03 (2023).

[12] W. Easttom, *Modern Cryptography: Applied Mathematics for Encryption and Information Security.* Springer International Publishing, (2020).

[13] H. Yu and Y. Kim, "New rsa encryption mechanism using one-time encryption keys and unpredictable bio-signal for wireless communication devices," *Electronics*, vol. 9, p. 246, 02 (2020).

[14] L. Y. Katz, Jonathan, "Introduction to modern cryptography," *Second Edition, Taylor & Francis Group*, (2015).

[15] T. Lugrin, *One-Time Pad*, pp. 3–6. 04 (2023).

[16] K. Manty, "A closer look at the enigma machine," 01 (2021).

[17] W. Diffie and M. E. Hellman, *New Directions in Cryptography*, p. 365–390. New York, NY, USA: Association for Computing Machinery, 1 ed., (2022).

[18] B. Sankhyan, "Review on symmetric and asymmetric cryptography," *International Journal for Research in Applied Science and Engineering Technology*, vol. 12, pp. 2934–2940, 03 (2024).

[19] D. Atkins, M. Graff, A. Lenstra, and P. Leyland, "The magic words are squeamish ossifrage," vol. 917, pp. 263–277, 01 (1994).

[20] M. Gardner, "Mathematical games – a new kind of cipher that would take millions of years to break," *Scientific American*, vol. 237, no. 2, pp. 120–124, (1977).

[21] R. L. Rivest, A. Shamir, and L. M. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 26, pp. 96–99, (1978).

[22] M. O. Rabin, "Digitalized signatures and public-key functions as intractable as factorization," tech. rep., USA, 1979.

[23] H. Williams, "A modification of the rsa public-key encryption procedure (corresp.)," *IEEE Transactions on Information Theory*, vol. 26, no. 6, pp. 726–729, (1980).

[24] T. Takagi, *Fast RSA-type cryptosystem Modulo pkq*, vol. 1462, pp. 318–326. 07 (2006).

[25] T. Takagi, "Fast rsa-type cryptosystems using n-adic expansion," in *Advances in Cryptology — CRYPTO '97* (B. S. Kaliski, ed.), (Berlin, Heidelberg), pp. 372–384, Springer Berlin Heidelberg, (1997).

[26] E. Kaltofen and V. Shoup, "Fast polynomial factorization over high algebraic extensions of finite fields," in *Proceedings of the 1997 International Symposium on Symbolic and Algebraic Computation*, ISSAC '97, (New York, NY, USA), p. 184–188, Association for Computing Machinery, (1997).

[27] K. Rabah, "Review of methods for integer factorization applied to cryptography," *Journal of Applied Sciences*, vol. 6, pp. 458–481, 02 (2006).

[28] C. Fontaine, B. Bhanu, M. Sherr, D. Hankerson, A. Menezes, T. Lange, J.-J. Quisquater, D. Samyde, G. Bleumer, M. De Soete, K. Sako, D. Boneh, Y. Desmedt, P. Zimmermann, F. Morain, M. Ward, A. Ochieano, F. Bauer, B. Kaliski, and C. Paar, *Elliptic Curve Method for Factoring*, pp. 401–403. 01 (2011).

[29] S. Rubinstein-Salzedo, *The Vigenère Cipher*, pp. 41–54. 09 (2018).

[30] S. Kute, C. Desai, and M. Jadhav, "Study on encryption decryption of rsa and elgamal," *L En-je Lacanien*, vol. 10, 02 (2023).

[31] R. Dakhni, S. Sandilkar, and D. Balaram, "Implementation of rsa algorithm for encryption and decryption recent technology trends in computer technology," vol. 3, pp. 2581–9429, 05 (2023).

[32] Z. Luo, R. Liu, and A. Mehta, "Understanding the rsa algorithm," 08 (2023).

[33] R. L. Rivest, "Critical remarks on "critical remarks on some public-key cryptosystems"by t. herlestam," *BIT Numerical Mathematics*, vol. 19, no. 2, pp. 274–275, (1979).

[34] T. Herlestam, "Critical remarks on some public-key cryptosystems," *BIT Numerical Mathematics*, vol. 18, no. 4, pp. 493–496, (1978).

[35] T. Collins, D. Hopkins, S. Langford, and M. Sabin, “Public key cryptographic apparatus and method,” October (2008).

[36] S. Bellare, Mihir; Goldwasser, “§2.3.5 a squaring permutation as hard to invert as factoring,” (2008).

[37] M. Avendaño, T. Krick, and A. Pacetti, “Newton-hensel interpolation lifting,” *Foundations of Computational Mathematics*, vol. 6, 10 (2005).

[38] S. R, P. Klnc, and K. Srm, “An improved cryptanalysis of multi-prime rsa with specific forms of decryption exponent,” *Cryptologia*, pp. 1–14, 01 (2024).