An Improved DNA Cryptography Using Residue Number System

Salaudeen Habib Lekan

Department of Computer Science, Federal Polytechnic Ohodo, Enugu, Nigeria.

DOI: https://dx.doi.org/10.47772/IJRISS.2024.806037

Received: 15 May 2024; Revised: 27 May 2024; Accepted: 31 May 2024; Published: 01 July 2024

ABSTRACT

In this information rich era, data protection is needed to ensure swift and secure communication through a digital medium. Data need to be protected from unauthorized access and transmitted to the intended receiver with confidentiality, availability, integrity and authenticity. Several schemes have been proposed over the years toward ensuring that data sent over a digital medium is difficult to understand by an intruder by hiding or transforming the data from a plain-text to a DNA based form which is sent as the cipher-text. Although, some of the schemes performed to a certain level but high computational time is the problem of most of these techniques. Therefore, this research presented a method that incorporate cryptography in securing the data using residue number system and DNA cryptography. Firstly, residue number system is used in order to reduce the computational time by encrypting plain-text into a residue form, using the moduli set {2n-1, 2n, 2n+1}, then convert the residue to a DNA format. The DNA cryptography was done to hide the existence of the encrypted DNA by using two different reference sequence randomly generated. The proposed scheme produces the cipher text in both an encrypted and encoded DNA based form, which takes less computational time and attracts less attention of intruders.

Keywords: DNA; Data; Computational time; Cryptography; Residue number system

INTRODUCTION

Communicating digitally has evolved to be a fundamental aspect of interaction between two ends in this generation, with a lot of internet-based field. Keeping communication secret is of high importance. So, the security of data passed over a network is a primary consign, which stretches to the confidentiality as well as integrity of the data, making it mandatory to protect against intruders or unauthorized access and use. In the quest to have a secured communication between two ends, the concept of cryptography and steganography came into light. Cryptography and steganography are usually interrelated and share the common aims and services of preserving the confidentiality, integrity, and availability of information, which are some of the most significant fields in computer security, [1]. Cryptography is an historical science that began in Egypt around 1900 B.C. with hieroglyphic writing, [2] It uses encryption to scramble the secret information in such a way that only the sender and the intended receiver can reveal it, [4] On the other hand, steganography began in ancient Greece around 440 B.C. [3]. It is the art of hiding data into a medium in a way that makes data unsuspicious. The hiding media used can be in the form of images, audios, videos and DNA which makes the data difficult to detect. From nature to science, the idea that genes themselves are made of information stimulated the research in molecular deoxyribonucleic acid (DNA). DNA molecule has three main advantages that make it an efficient medium for data hiding and transmission. [5] First of all, its high storage capacity; One trillion bits of binary data can be stored in one cubic decimeter of DNA solution. Secondly, DNA molecules show parallel computation, which means DNA based processes are capable of intense processing. DNA chains have large scale of parallelism and its computing speed could reach up to 1 billion times per second computations. Third, DNA based computers also have very less power

consumption, which is equal to one – billionth of a traditional computer. [6]

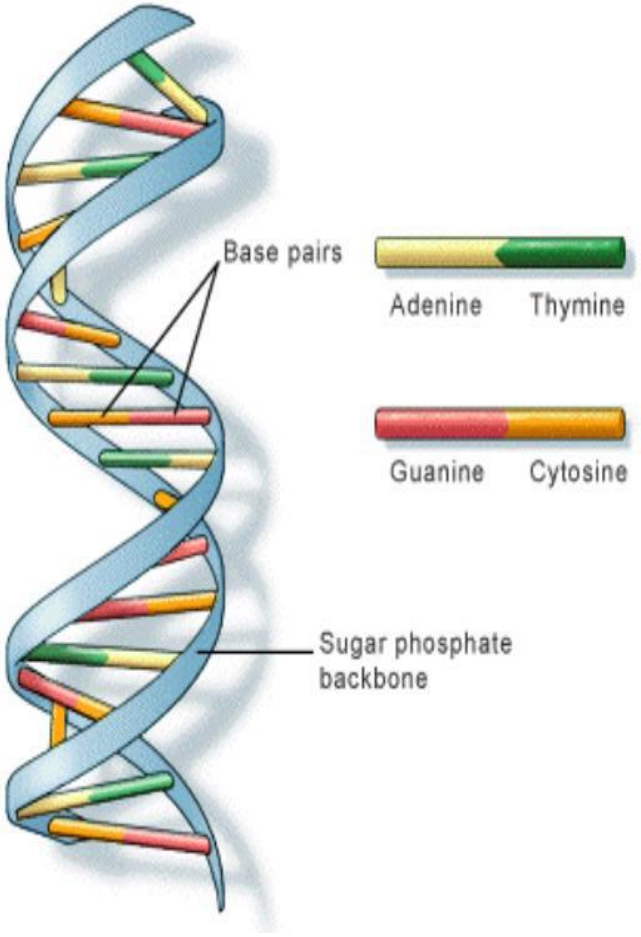

Fig. 1.1. DNA Structure

The DNA is the Deoxyribo nucleic acid which contains six smaller molecules in it. A sugar molecule called deoxyribose, a phosphate molecule, and four different nitrogenous bases (Adenine, Thymine, Cytosine and Guanine). The two long strands twisted around like a ladder to form a double helix model which is made up of sugar and phosphate group. [10].

This paper is organized as follows: section 2 discusses Residue Number System, section 3 states the Existing works, section 4 explains the structure of the proposed algorithm, section 5 shows the performance analysis and comparisons, section 6 states the choice programming language while section 7 is the conclusion of this work.

RESIDUE NUMBER SYSTEM (RNS)

Residue Number Systems (RNS) is an unweighted number system, which is usually employed in addition and multiplication dominated intensive applications such as fast Fourier transform, discrete Fourier transform, image processing, cryptography, digital filtering, and video coding. This is due to the RNS inherent features, such as carry free operations, parallelism, modularity, and fault tolerance. RNS based calculation requires reverse and forward conversions, which must be as fast as possible not to nullify the RNS advantages. [9].

Let $\{m_1, m_2, m_3\ldots, m_n\}$ be a set of positive integers all greater than 1. m_i is called a modulus, and then n-tuple set $m_1, m_2, m_3\ldots, m_n\}$ is called moduli set. Consider an integer number Y. For each of the modulus in

$\{m_1, m_2, m_3 \ldots, m_n\}$, we have yi = Y mod mi, (which will be denoted as $|Y|_{mi}$). Thus the number Y in this system is represented as $Y = (y_1, y_2, y_3 \ldots, y_n)$,$0 \leq y_i < mi$.[9].

Given the moduli set {7,8,9}, the number 150 can is depicted in RNS as:

$y_1 = |y|_{m1} = |150|_7 = 3$

$y_2 = |y|_{m2} = |150|_8 = 6$

$y_3 = |y|_{m3} = |150|_9 = 6$

Thus, the RNS representation of 150 is $(3,6,6)_{RNS(7|8|9)}$.

In order to avoid redundancy, the moduli set must be pair wise relatively prime. Thus gcd $(m_i, m_j) = 1$ for $i \neq j$, where gcd stands for greatest common divisor of m_i ,m_j.

Let $M = \prod_{i=1}^{n} mi$, then the RNS representation is unique for any integer M is called the dynamic range. [9]

Decimal to Residue (D/R) converter (encoder) is needed in order to convert a decimal number to RNS representation.

Forward Conversion

Forward conversion is done by a forward converter which decomposes a weighted binary number into a residue represented number with regards to a moduli set. It is the conversion from a conventional representation to a residue representation by dividing the number X by each of the given moduli and then collecting their remainder, (Gbolagade & Cotofana, 2008).

Taking the example, moduli set {3, 4, 5}, 4 is depicted in RNS as: Therefore, the depiction of 4 in RNS as:

$xi = |X|m_i$ (2.1)

$x_1 = |x|_{m1} = |4|_3 = 1$; $x_2 = |x|_{m2} = |4|_4 = 0$; $x_3 = |x|_{m3} = |4|_5 = 4$

therefore, the RNS representation of 4 is $(1, 0, 4)_{RNS(3|4|5)}$

Reverse Conversion

The conversion from residue to a conventional representation (either binary or decimal representation) is known as reverse conversion. The two most widely used techniques of reverse conversion are the Chinese Remainder Theorem (CRT) and Mixed Radix Conversion (MRC) [9].

Conversion by Chinese Remainder Theorem (CRT)

For a moduli set with and a dynamic $\{m_1, m_2, \ldots, m_k\}$ with gcd $(m_i, m_j) = 1$ for $i \neq j$ and a dynamic range $M = \prod_{i=1}^{n} mi$, the residue number $\{x_1, x_2, \ldots, x_k\}$ can be converted into the decimal number X if the moduli set are co-prime [9].

$X = |\sum_{i=1}^{k} mi|mi^{-1}\ xi|mi|m$ (2.2)

$M = \prod_{i=1}^{n} mi$ and $Mi = \frac{M}{mi}$ and Mi^{-1} is the multiplicative inverse of M*i* with respect to m*i*.

Example: Finding the decimal equivalent of the RNS number (0, 0, 4) with respect to the moduli set {3,4,5}.

Solution:

M = $\prod_{i=1}^{n} mi$ = 3 x 4 x 5 = 60

$M_1 = \frac{M}{m1} = \frac{60}{3}$ = 20 then = $M1^{-1}$ = 2

$M_2 = \frac{M}{m2} = \frac{60}{4}$ = 15 then = $M2^{-1}$ = 3

$M_3 = \frac{M}{m3} = \frac{60}{5}$ = 12 then = $M3^{-1}$ = 3

X = |0 x 20 x 2 + 0 x 15 x 3 + 4 x 12 x 3|60

= |0 + 0 +144|60

= 24

In this paper, limitations in [7] and [8] were pointed out and then proposed an improved DNA cryptography scheme using residue number system. Firstly, we encrypt the text data to a residue form with respect to the moduli set {2n-1, 2n, 2n+1}, which have a common factor of 2. This simplifies the encryption process due to its low complexity compared to the delayed chaotic neural network used in [8] and also increases the permutation and combination that enhanced the robustness of the security when compared to [7].

EXISTING WORKS

Dr. R. Surrendering et.al. [15] discusses the significance of the Internet of Things (IoT) in various applications due to its ability to establish effective communication between interconnected devices. The potential threat of unauthorized access to sensitive information during storage raises concerns

about data confidentiality, integrity, and privacy. To address these concerns the paper introduces a novel approach named DNA-based Elliptic Curve Cryptography technique with RedFox Optimization algorithm for clustering (RF-DECC). This approach is aimed at enhancing data security in fog computing. The process involves identifying cluster heads using the RedFox optimization technique. Once clustering is completed, the cluster head initiates the data encryption process by applying DNA-Elliptic Curve Cryptography (ECC) to encrypt the data of cluster members. ECC is highlighted as a lightweight public key cryptography algorithm, and the article underscores that using DNA in conjunction with ECC adds complexity to the encryption process.

In [16] a system is proposed that demonstrate its effectiveness in withstanding attacks. The technique offers the ability to retrieve the original data (audio, video, image, or text files) without any alterations during the decryption process.

The paper addresses the limitations of conventional cryptographic methods in securing online activities and introduces DNA cryptography as an innovative approach. By combining the Trellis algorithm with DNA sequences and random key generation, the technique aims to enhance security and resist attacks, ensuring that original data can be securely retrieved after decryption.

Al-Harbi, et.al. [17], investigates the most recent data hiding techniques based on DNA steganography, including the highly improved DNA-based steganography technique, the data hiding using double DNA sequences method, and the enhanced DNA-based steganography technique. The strengths and weaknesses

of these techniques are discussed. Additionally, the security of these techniques is analyzed based on several security parameters that measure the quality of DNA steganography with respect to many factors, including, but not limited to, cracking probability, blindness, modification rate and expansion rate, and layers of security. The goal of the comparison between the investigated techniques is to highlight the advantages and disadvantages of the existing data hiding algorithms and to motivate future research in this field. Moreover, the paper evaluates the discussed techniques based on some parameters, including capacity, payload, and bit per nucleotide (bpn). The result shows that the enhanced DNA-based steganography technique hides 2 bpn, whereas the highly improved method can hide on average 1.46 bpn, which is higher than data hiding using double DNA sequences method can hide. The paper also presents suggestions for how each technique can be optimized to achieve a higher security level for hiding data within DNA sequences.

OA Al-Harbi, et.al. [11], investigates the most recent data hiding techniques based on DNA steganography, including the highly improved DNA-based steganography technique, the data hiding using double DNA sequences method, and the enhanced DNA-based steganography technique. The strengths and weaknesses of these techniques are discussed. Additionally, the security of these techniques is analyzed based on several security parameters that measure the quality of DNA steganography with respect to many factors, including, but not limited to, cracking probability, blindness, modification rate and expansion rate, and layers of security. The goal of the comparison between the investigated techniques is to highlight the advantages and disadvantages of the existing data hiding algorithms and to motivate future research in this field. Moreover, the paper evaluates the discussed techniques based on some parameters, including capacity, payload, and bit per nucleotide (bpn). The result shows that the enhanced DNA-based steganography technique hides 2 bpn, whereas the highly improved method can hide on average 1.46 bpn, which is higher than data hiding using double DNA sequences method can hide. The paper also presents suggestions for how each technique can be optimized to to achieve a higher security level for hiding data within DNA sequences.

El-Latif, et.al. [12], suggested a method which has two rounds of encryption. This scheme is the same as the existing technique named the Data Encryption Standard (DES) algorithm. In this method, two keys are used for encoding the plaintext. These two keys are made up of the elliptic curve cryptography (ECC), and Gaussian kernel function (GKF) and another key is created on random based injective mapping on the second characters repeated in the first key. At last, the encryption message arbitrarily hides in the second DNA sequence based on the numbers from GKF.

Sohal, et.al. [13], introduced a new method with the cryptographic technique. In this technique, client-side data is encrypted before storing it in the cloud. This is a symmetric-key cryptography scheme which uses DNA cryptography. Apart from presenting the thorough design of this approach, and comparing it with the present symmetric-key algorithms (DNA, AES, DES, and Blowfish), the experimental results show that this method leaves behind the traditional algorithms based on ciphertext size, encryption time, and throughput. Hence this new method is much more efficient and performs better.

Tiwari, et.al. [14], recommended a scheme in which the DNA mapping technique was offered for ECC. In this method the DNA code is random, and non- repetitive subsections are allotted to alphabets. Then these alphabets are used for encoding and decoding at the two ends. This scheme was effectively employed and used in real- time internet of things devices.

P. Malathi, et.al. [7], modifies the insertion algorithm to decrease the cracking probability of the fake DNA sequence. The algorithm uses two different keys. The first key **(K1)(K1)**is a number in the range of 0 to 255, which is used to XOR the last character in the message (*M*); the result will be XORed with the character preceding the last one in the M, and so on. Accordingly, the first key is used to encrypt the message. The second key **(K2)(K2)**is randomly generated and is used to divide the DNA sequence into same-length segments. The resulting cipher characters are inserted as binary bits one by one at the beginning of each segment. Then, the binary sequence is converted into DNA bases. The second key is preferred to be

a small number so that the DNA sequence has a minimum length while hiding the secret message.

S.S. Roy, et.al. [8] proposed a new method using delayed chaotic neural network with a posterior DNA cryptography. The binary sequence needed to perform XOR operation with message blocks is generated from chaotic neural network. The permutation of the plaintext and the number of epochs is also based on the chaotic neural network. It is difficult for any cryptanalyst to determine the actual parameters of the encryption method and decrypt the DNA cipher sequence. Without knowing all the parameters i.e. input, delay function of the chaotic neural network as well as the position bits for using permutation operation. The scheme is slower for use in practical applications. In case of online media file transmission, the scheme would not provide efficient solution.

PROPOSED WORK

Improving the existing system towards providing a better means through which secure communication can be ensured from one end to the other, a new scheme was proposed. The concept of residue number system was adopted to encrypt the corresponding ASCII value of the character entered, through computation of forward conversion. The scheme contains one DNA lookup table which is used for substitution when encoding the encrypted message into a DNA sequence. with the DNA sequence implementing an insertion method, the data encoded in DNA base is merge together with two different reference DNA sequence randomly generate to form a cipher DNA sequence which is sent to the receiver.

The reverse conversion, which is the decryption process makes use of the Chinese remainder theorem (CRT) method, which first returns the cipher-text back from RNS form to ASCII form which is then converted to the original message after the system has make use of the lookup table to convert the DNA sequence back to it binary form and then to it equivalent residue form.

This will be achieved using the developed system as shown in Figure 4.1.

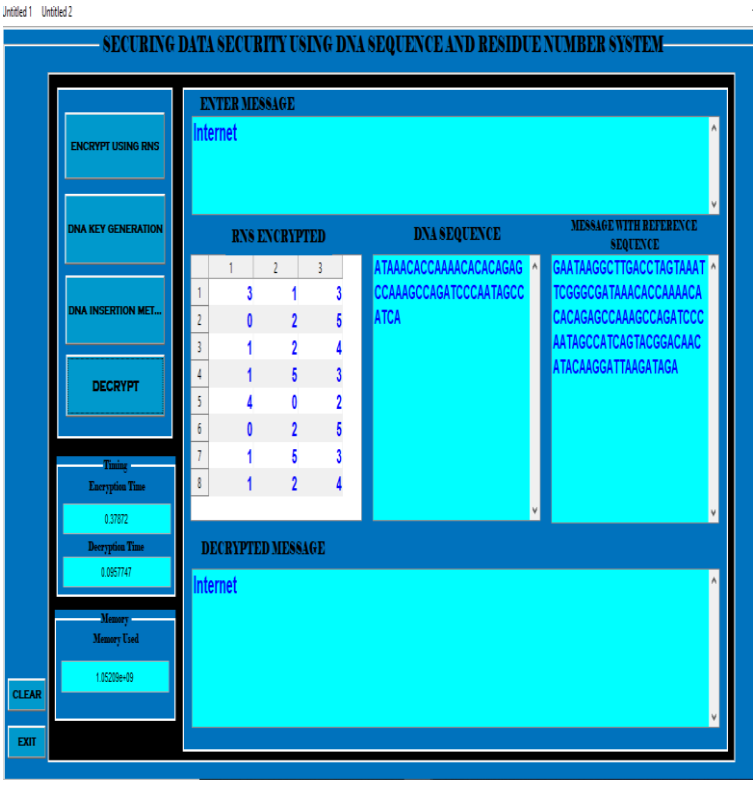

Figure 4.1: System Interface

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS |Volume VIII Issue VI June 2024

Table 4.1: DNA lookup table

DNA Base	Binary code
A	**00**
C	**01**
G	**10**
T	**10**

The system accepts the text that is entered in the text field. Thereafter, the user will be required to click the "Encryption using RNS" button, which will trigger the system to perform both the conversion of the text entered to ASCII form and forward conversion. After the encryption, the user will be required to click the "DNA Key Generation" button in order to convert the encrypted message to a DNA sequence. Then the user is required to click the "DNA Insertion method" to encode DNA sequence into a reference DNA to generate a cipher DNA sequence. To decrypt the cipher text, user only needs to click the "Decryption" button, enabling the system to reverse the process to produce the plain text.

Algorithm for encoding

Step 1**:** Given Plain-text **M** and Convert **M** into their respective ASCII numbers (in decimal format) are grouped into blocks.

Step 2**:** The ASCII numbers are converted into residues with respect to the moduli set {2n-1, 2n, 2n+1}.

Step 3**:** The residues are converted into 4-bit binary number (0's and 1's).

Step 4**:** The sequence of binary numbers is broken in pairs. The pairs could be 00, 01, 10, 11. These pairs are converted into a DNA sequence using Table 4.1.

Step 5**:** Based on Step 4, the DNA sequence as **S** is created.

Step 6**:** Two reference sequence **R** and **Q** are generated randomly online out of 163 million available on bioinformatics.org.

Step 7**:** DNA sequence **S** is inserted in-between the two reference sequence **R** and **Q** to create Cipher DNA **G**

Step 8**:** Return **G** and send **G** to the receiver.

Example of encoding technique for message 'We'

Step 1: Convert the plain text to it equivalent ASCII values

W 87

e 101

Step 2: Convert the ASCII value to residue with respect to the moduli set {2n-1, 2n, 2n+1}, where value of n = 2.

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS |Volume VIII Issue VI June 2024

ASCII	**Residue**
87	2 3 3
101	1 5 3

Step 3: Convert the residue to a 4 bit's

Residue	**4 bits binary**
2 3 3	0010 0011 0011
1 5 3	0001 0101 0011

Step 4: Convert the 4 bit's binary into DNA base using DNA base table from table 4.1 to form DNA 'A'.

4 bits binary	**DNA base**
0010 0011 0011	AG AT AT
0001 0101 0011	AC CC AT

Step 5: Based on Step 4 the DNA sequence **S** is created by listing the DNA base in linear.

S = 'AGATATACACAT'

Step 6: Generate two reference 30 bases long DNA sequence '**R**' and '**Q**' randomly. Consider the sequence '**GAATAAGGCTTGACCTAGTAAATTCGGGCG**' and '**GTACGGACAACATACAAGGATTAAGATAGA**' as the two reference sequence 'B' and 'C' respectively.

Step 7: Insert the encrypted DNA 'A' in-between reference DNA sequence '**R**' and '**Q**' and concatenate them to form cipher DNA '**G**'.

Concatenate 'R', 'S' and 'Q'

R	**GAATAAGGCTTGACCTAGTAAATTCGGGCG**
S	**AGATATACCCAT**
Q	**GTACGGACAACATACAAGGATTAAGATAGA**

Cipher DNA sequence

G	**GAATAAGGCTTGACCTAGTAAATTCGGGCGAGATATACCCATGTACGGAC AACATACAAGGATTAAGATAGA**

Algorithm for decoding

Step 1: The Cipher DNA **G** is received from sender.

Step 2: DNA sequence **S** is extracted from the cipher DNA **G.**

Step 3: DNA sequence **S** is converted into binary and splitted into 4-bits.

Step 4: The binary numbers are converted into residue (decimal).

Step 5: The residues are converted into ASCII numbers (decimal) using Chinese Remainder Theorem.

Step 6**:** ASCII numbers are converted back to Plain-text **M**.

Step 7**:** Return **M**.

Example of decoding technique for message 'We'

Step 1: Cipher DNA '**G**' is received.

Step 2: Extract the DNA '**S**' from cipher DNA '**G**' by counting 30 from the left and 30 from the right and extract the middle DNA sequence

GAATAAGGCTTGACCTAGTAAATTCGGGCG'AGATATACCCAT'GTACGGCAACATACAAG GATTAAGATAGA

Step 3: Convert the DNA 'A' to 4 bits binary using the DNA base table.

DNA base			**4 bits binary**		
AG	AT	AT	0010	0011	0011
AC	CC	AT	0001	0101	0011

Step 4: Convert 4 bits binary to residue.

4 bits binary			**Residue**		
0010	0011	0011	2	3	3
0001	0101	0011	1	5	3

Step 5: Convert the residue to ASCII using Chinese Remainder Theorem.

The CRT is given by

$$X = |\sum_{i=1}^{k} mi|mi^{-1}\, xi|_{mi}|_m \quad (4.1)$$

$\prod_{i=1}^{n} mi$ and $M_i = \frac{M}{mi}$ and Mi^{-1} is the multiplicative inverse of M_i with respect to m_i.Using the moduli $m_1 = 5$, $m_2 = 6$, $m_3 = 7$, the dynamic range is;

$M = \prod_{i=1}^{N} m_i$ = 5 x 6 x 7 = 210

$M_1 = \frac{M}{m1} = \frac{210}{5} = 42$, $M_2 = \frac{M}{m2} = \frac{210}{6} = 35$, $M_3 = \frac{M}{m3} = \frac{210}{7} = 30$

$M_1^{-1} = 3$, $M_2^{-1} = 5$, $M_3^{-1} = 4$,

For the first row of the cipher-text with residue {2, 3, 3} the reverse conversion is;

$|x|_{210} = |(2 \text{ x } 42 \text{ x } 3) + (3 \text{ x } 35 \text{ x } 5) + (3 \text{ x } 30 \text{ x } 4)|_{210}$

$= |252 + 525 + 360|_{210}$

$= |1137|_{210} = 87$

For the next row of the cipher-text with residue {1, 5, 3} the reverse conversion is;

$|x|_{210} = |(1 \text{ x } 42 \text{ x } 3) + (5 \text{ x } 35 \text{ x } 5) + (3 \text{ x } 30 \text{ x } 4)|_{210}$

$= |126 + 875 + 360|_{210}$

$= |1361|_{210} = 101$

Step 6: Convert the ASCII value to it corresponding Alphanumeric value and concatenate to form the plain text.

ASCII **Alphanumeric**

87 ⟶ W

101 ⟶ **e**

Step 7: Hence the message “We” is retrieved.

Cracking Probability

It is the total probability to predict the confidential information hidden inside the reference DNA sequence. The attacker needs the following information to crack the secret message hidden in the reference DNA.

The message is encrypted using residue number system with respect to specific moduli set, specific arrangement of the moduli set and a specific value of n. This measure is the first information for the intruders to break the hidden information in the DNA sequence, and thus the probability of guessing the moduli set is

$$\frac{1}{(120^n - 1)} \qquad (4.2)$$

and the probability of guessing the arrangement of the moduli set is

$\frac{1}{6}$ (4.3)

and the probability of guessing the specific value of n used is

$\frac{1}{(3^n-1)}$ (4.4)

The size of the reference DNA available is about 163 million. This is another information the intruder will need to crack the secret message. Thus, the probability to predict reference DNA sequence twice is

$\frac{1}{(1.63 \times 10^8)\hat{}2}$ (4.5)

The binary coding of A, C, G, T gives different combinations of two, thus the probability to guess binary coding is

$\frac{1}{24}$ (4.6)

Thirdly, the size of the message and prefix DNA is another information available to the intruder to crack the hidden message and the probability of finding message and reference DNA sequence is

$\frac{1}{(n-1)\hat{}2}$ (4.7)

The message and DNA are segmented and this provides another information to the intruder and thus the probability of guessing the segmentation of the DNA is

$\frac{1}{2^s-1}$ (4.8)

and the probability of guessing segmentation of message is

$\frac{1}{2^m-1}$ (4.9)

Hence the total probability to find the message hidden in the DNA sequence using the proposed scheme is

$\frac{1}{(120^b-1)} X \frac{1}{6} X \frac{1}{(3^d-1)} X \frac{1}{(1.63 \times 10^8)\hat{}2} X \frac{1}{24} X \frac{1}{(n-1)\hat{}2} X \frac{1}{2^s-1}$ (4.10)

Here,

- d is the number of bits in the moduli set.
- n is the number of bits in the cipher DNA sequence.
- m is the number of bits in the secret message.
- s is the number of bits in the reference DNA sequence.

PERFORMANCE ANALYSIS AND COMPARISONS

Several words with different character length were encrypted using the previous schemes and the proposed scheme, both the encryption and decryption time was analyzed and evaluated. After the evaluation of the schemes, as shown in Table 5.1 and Table 5.2, the proposed scheme was compared with the existing scheme to check for success attained.

Table 5.1: Comparison in term of Encryption Time

CHARACTER (size)	P. Malathi (2017) (ms)	S.S. Roy (2017) (ms)	PROPOSED SCHEME (ms)
Design (6)	2.1321	3.1655	0.2786
Message (7)	2.9102	4.4316	0.3062
Internet (8)	3.2140	4.5224	0.3436
Database (8)	3.1452	4.9636	0.3049
Cryptography (12)	4.9801	6.3321	0.4020
Generational (12)	4.8431	6.3431	0.3077

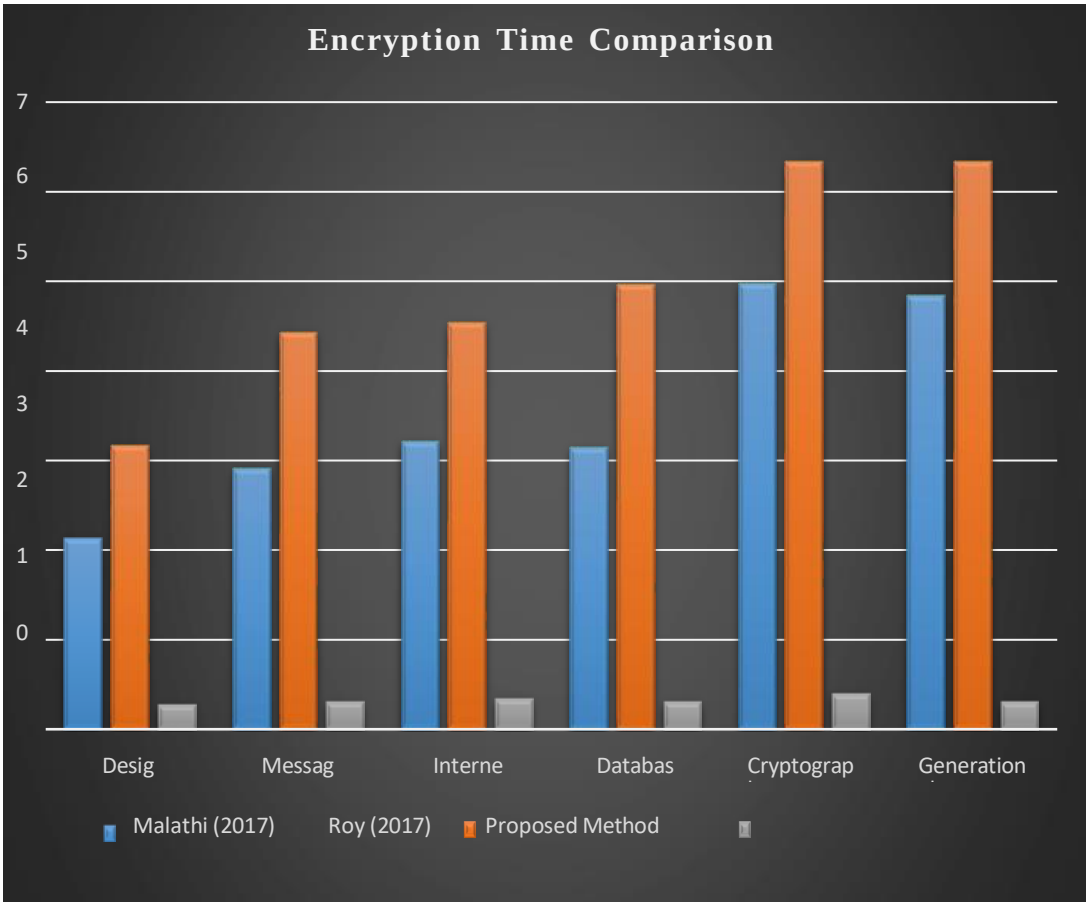

Figure 5.1: Comparison in term of Encryption Time

Table 5.2: Comparison in term of Decryption Time

CHARACTER (size)	P. Malathi (2017) (ms)	S.S. Roy (2017) (ms)	PROPOSED SCHEME (ms)
Design (6)	0.4152	1.1094	0.0851
Message (7)	0.9136	1.6641	0.1267
Internet (8)	1.4132	1.8351	0.0882
Database (8)	1.1246	2.0093	0.0912
Cryptography (12)	2.0148	3.3241	0.1387
Generational (12)	1.9241	3.4132	0.0916

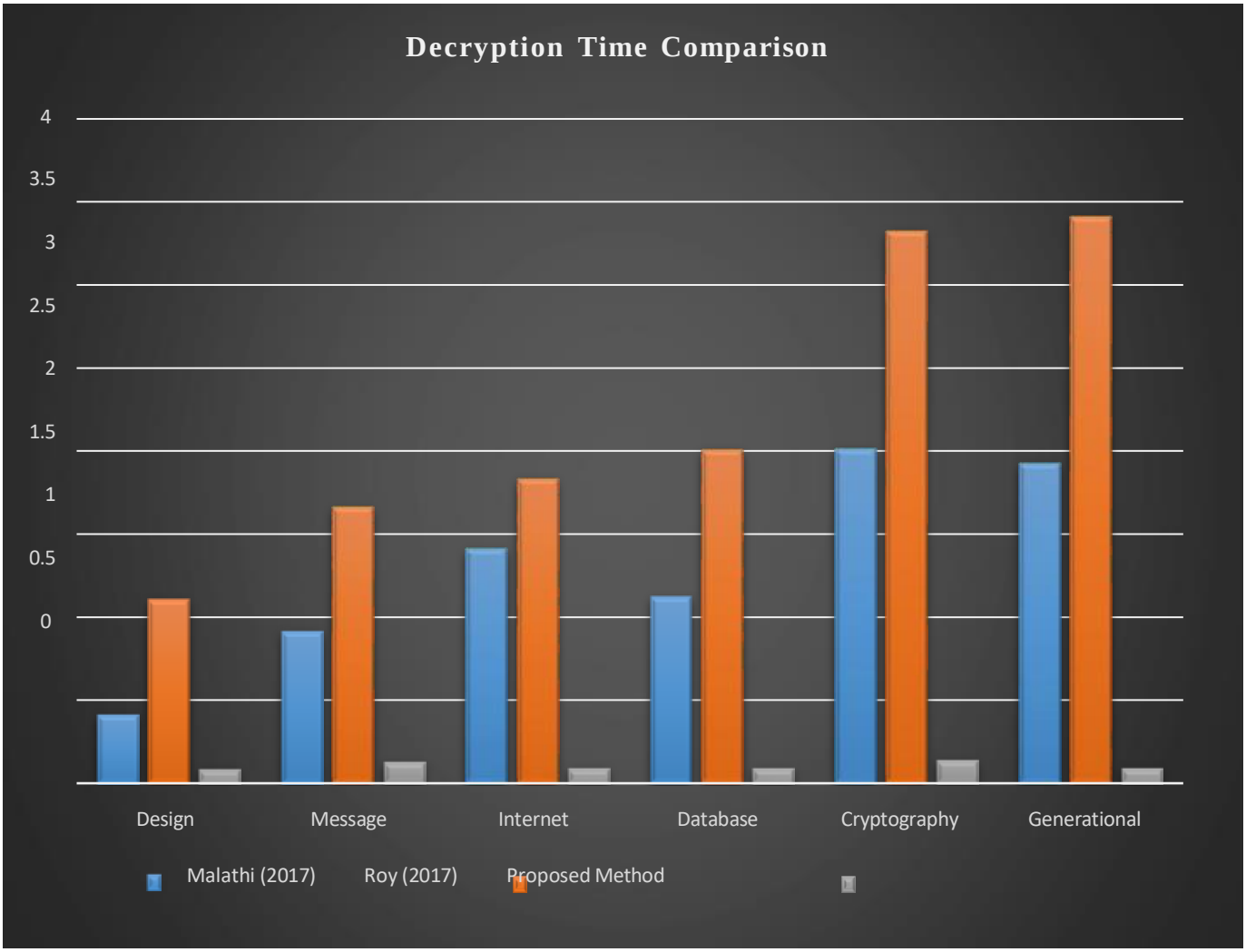

Figure 5.2: Comparison in term of Decryption Time

The results of the experiment carried out shows that the proposed system maintain a relatively low computational time regardless of the character length of the text being encrypted whereas previous schemes computational time keeps increasing base on the size of character being encrypted.

CHOICE OF PROGRAMMING LANGUAGE

The entire system was developed using MATLAB. MATLAB is a high-level programming language and interactive environment for numerical computation, visualization, and programming. Using MATLAB, you can analyze data, develop algorithms, and create models and standalone applications. The language, tools, and in-built math functions enable you to explore multiple approaches and reach a solution faster than with spreadsheets or traditional programming languages, such as C/C++ or Java. MATLAB programming language is also object-based programming language and it is integrated with MATLAB Compiler Runtime for the purpose of creating applications that will work outside of MATLAB integrated environment, thus making room for applications to run without installing MATLAB.

CONCLUSION

The proposed scheme has provided a better means of transmitting message securely and faster over the network while the message maintains an extremely high level of secrecy, proving to outperform it

predecessors in term of computational time to produce results and still maintaining the secrecy attached to the message. The result of the cipher-text produced in the proposed scheme is very complex and to crack it, there are many probabilities that should be tried, such as: finding the sequence of the algorithms used, determining the moduli set, determining the value of n for the moduli set, determining the arrangement of the DNA lookup table and also determining the first and the second reference sequence chosen randomly.

REFERENCE

1. Krishnan RB, Thandra PK, Sai Baba M (2017) An overview of text steganography. In: 2017 4th international conference on signal processing, communication and networking (ICSCN). IEEE
2. Sokół B, Yarmolik VN (2005) Cryptography and steganography: teaching experience. Enhanced methods in computer security, biometric and artificial intelligence systems. Springer, Boston,
3. Vinodhini RE, Malathi P, Gireesh Kumar T (2017) A survey on DNA and image steganography. 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS). IEEE
4. Selvaraj D (2017) Development of a secure communication system based on steganography for mobile devices. p 3
5. Samiha M., Ahmed S., Khaled N. (2017). Utilizing DNA Strands for Secured Data-Hiding with High Capacity. International Journal of Interactive Mobile Technology, Vol. 11, No 2.
6. Ahsan O., Muhammad I.F. (2015). DNA Cryptography Algorithms and applications. HiTech University.
7. Malathi P., et.al. (2017). Highly Improved DNA Based Steganography. 7th International Conference on Advances in Computing & Communications, ICACC, Procedia Computer Science 115, 651–659.
8. Roy S.S., et.al. (2017). A Novel Encryption Model for Text Messages using Delayed Chaotic Neural Network and DNA Cryptography. 20th International Conference of Computer and Information Technology (ICCIT).
9. K.A. Gbolagade and S.D. Cotofana (2008). A residue to binary converter for the{2n+2;2n+1;2n} moduli set. Proceedings of 42nd Asilomar Conference on Signals, Systems, and Computers, pp. 1785-1789.
10. Khalifa A. LSBase: A key encapsulation scheme to improve hybrid crypto-systems using DNA steganography. In 8th IEEE International Conference on Computer Engineering & Systems (ICCES) 2013; 105-110.
11. O.A. Al-Harbi, W.E. Alahmadi, A.O. Aljahdali (2020). Security analysis of DNA based steganography techniques. Springer Nature journal, SN Applied Sciences 2:172.
12. E.I. Abd El-Latif, M.I. Moussa (2019). Information hiding using artificial DNA sequences based on Gaussian kernel function. Journal of Information and Optimization Sciences ISSN: 0252-2667. 2169-0103.
13. Sohal, Sharma (2018). BDNA-A DNA inspired symmetric key cryptographic technique to secure cloud computing. Journal of King Saud University – Computer and Information Sciences.
14. H.D. Tiwari, J.H. Kim (2018). Novel Method for DNA- Based Elliptic Curve Cryptography for IoT Devices. ETRI Journal, Volume 40, Number 3.
15. P. K. R. Dr. R.Surendiran, "A Fog Computing Approach for Securing IoT Devices Datausing DNA-ECC Cryptography," *DS Journal of Digital Science and Technology,* vol. 1, no. 1, pp. 1, 2, 2022.
16. E. B. K. Rama Devi, "An Enhancement in Data Security Using Trellis Algorithm with DNA Sequences in Symmetric DNA Cryptography." *Wireless Personal Communications,* p. 1, 202.
17. Al-Harbi O.A., Alahmadi W.E. & Aljahdali A.O. (2020). Security analysis of DNA based steganography techniques. *Springer Nature journal, SN Applied Sciences* 2:172.