International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878, Volume-8 Issue-5, January 2020

# A Novel Transposition Encryption Emerging DNA Strand with Random Permutation

**Shakir M. H. Al-Farraji, Huda K. Saadeh**

***Abstract: The fast and wide development in information technology and data handling has made researchers to be interested in the data confidentiality and to think about finding new and complex methods in data encryption. In this paper, a novel method for encryption is introduced. The idea beyond this proposed method is to take the advantages of the DNA computing and merged it with the concept of the classical transposition cipher by applying the random permutation in order to design this novel encryption method. The emerging DNA computing with the random numbers for transposition produced a highly complex ciphertext that make the task of cryptanalysis more sophisticated and difficult to analyze and break it.***

***Keywords: DNA Computing, DNA Encryption, Random Permutation, Transposition Cipher.***

## I. INTRODUCTION

With the huge revolution in today's technologies and their effects on contemporary systems such as cloud, IoT, and wide spread of media usage, a major key to the success of many applications and systems that depend on these technologies is the security triangle that involves Confidentiality, Integrity, and Availability referred to as CIA triad [1], [2]. To avoid confusion of this term with the Central Intelligence Agency, another term is used as AIC triad which is referred to Availability, Integrity and Confidentiality. Integrity means safeguarding information against improper modification or destruction in order to keep it accurate and authentic. Availability is making the data and system available any time needed by authorized people [3]. General speaking the confidentiality is equivalent to privacy in which it concentrates to avoid reaching sensitive data to unauthorized people [4]. There are two methods to provide confidentiality services, one of them is Steganography, where data is hidden within another media [5], [6]. The other common method that ensures confidentiality is cryptography, in which data encryption is performed to generate ciphers. This paper proposes a cryptography method, which is a novel method to encrypt data by emerging the Deoxyribonucleic Acid (DNA) computing and random permutation that gives a very high complexity.

Ciphers can be generated by using two main categories of cryptographic technique: Symmetric and Asymmetric [7]. Symmetric techniques require that both the sender and the receiver of confidential information must share a secret key [8]. This secret key is used to encrypt a plaintext at the sender side to produce the ciphertext. And the same key is used to decrypt the ciphertext at the receiver side in order to obtain the original plaintext [1]. Keys are very important to cipher resistance against cryptanalysis attacks according to Kerckhoff's principle [9]. Keys secrecy plays the main role for protecting ciphers in the time that algorithms are supposed to be known to public. For that reason, powerful encryption techniques are the ones that maintain in its design a key generator core to produce randomize keys. The String Base Random Permutation patent method can be used to produce randomize keys that can be used to enhance the power of all types of ciphers [10]**.**

Asymmetric techniques are based on using two public and private keys. While the former is known by the users, the latter is kept secret by his owner [7]. In this cryptographic methodology, both confidentiality and non-repudiation services can be applied in cryptosystems. Preferences between asymmetric and asymmetric techniques depend on the system's requirements. On one hand, asymmetric techniques provide wide foundation for authentication and digital signature methods. On the other hand, they require more computational resources which is not practical in most cases. On the contrary, to asymmetric techniques, symmetric techniques are more efficient and used to encrypt large amount of information [1], [7].

Cryptographic techniques can be also categorized according to their methods of implementation. While Asymmetric techniques are based on mathematical foundations. Classical symmetric techniques are based on substitution and transposition. Substitution methods include substituting plaintext characters into other characters. Transposition methods reorder character's positions in the plaintext to constitute the cipher. Both, substitution and transposition are implemented using substitution and transposition tables generated to achieve powerful ciphers [1]. Powerful ciphers have two important properties [11]: First, high confusion which means that the relation between the cipher and the key is weak, therefor, any changes in the key value will cause huge variances in the cipher. Second, high diffusion which represents a weak relation between the cipher and the original plaintext, where small changes in plaintext result in huge changes in the cipher. Confusion and diffusion are both important to protect ciphers from different cryptanalysis attacks to uncover the key or the original messages [1].

Revised Manuscript Received on January 15, 2020

**Shakir M. H. Al-Farraji,** Computer Science Department, Faculty of Information Technology, University of Petra, Amman, Jordan. Email: shussain@uop.edu.jo

**Huda K. Saadeh,** Department of Information Security, Faculty of Information Technology, University of Petra, Amman, Jordan. Email: hsaadeh@uop.edu.jo

Cryptography have evolved by using contemporary techniques such as Quantum, Biometric, and DNA techniques to strengthen the cipher against cryptanalyst attacks [1], [5], [7]. DNA Molecule is famous for its double helix-coiled two chains, each consists of nucleotides units. Each of these units is composed of a deoxyribose sugar, a phosphate group and one of Cytosine (C), Guanine (G), Adenine (A), and Thymine (T) nitrogen-containing nucleobases [5]. In DNA nucleotides from each double-stranded chain is bonded together in a pairing rule where A bonds with T and C bonds with G as shown in Figure 1 [8]. The patterns appear in the strand reflects massive genetic information for functioning, growth, and production of new generations. Some regions in the DNA strand called the coding regions are translated to proteins and they represent the functionality of the DNA. Whereas, other noncoding regions are not translated. Central dogma is a process of protein production in which DNA information is converted into Ribonucleic acid (RNA) information known as transcription. Transcription uses DNA as a base template but substitutes Thymine (T) with uracil (U). After transcription, RNA information is translated into proteins [5]. Polymerase Chain Reaction (PCR) is another known DNA computing process used for DNA amplification for its simplicity [3], [12]. DNA amplification widely impacts forensics, medicine as well as other research fields [3], [12].

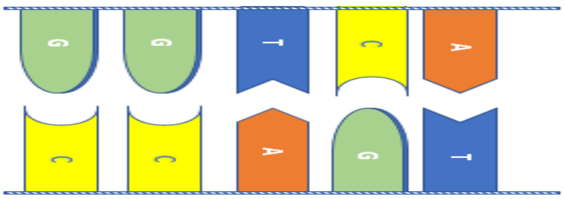

**Fig. 1. DNA strands bonds**

DNA Bio-Computing techniques opened the door into wide variety of problem solving, such as using DNA computing in solving the Hamiltonian path problem [5]. This solution was found and proofed by Leonard Adelman back in 1994 [13]. Adelman's study shows how DNA computing can be used in solving problems efficiently because of its both parallel processing and massive information storage capabilities [8], [14]. Authors of [5] have categorized DNA computing research into three research categories: classical such as cryptography, natural such as nanotechnology, and those which support fundamental research such as bio-molecular chemistry and DNA chips.

For security purposes, researchers have studied the DNA power in cryptography and steganography techniques for its ability to store high amount of data and that converting data into/from DNA is more convenient than current techniques [8], [11], [14]. Most of these techniques transform each two binary bits in the data stream into one of the DNA (A, T, C, G) nucleotides mentioned earlier. According to this, binary transformations could be implemented using one of eight encoding rules as mentioned in [15], taking into consideration that A is complementary to T, and C is complementary to G. Table I is an example of one of these encoding rules, where binary 00 and 01 is encoded into A and C respectively, which implies that T and G encode 11 and 10 respectively. A byte of binary data is represented by a sequence of four DNA nucleotides. For example, 00100011 is represented by the sequence AGAT according to rule described in Table 1.

This paper focuses on keeping information secrecy using DNA cryptography by emerging the DNA computing in the transposition cipher with a random permutation in order to produce a powerful encryption method.

**Table I: Binary-DNA transformation**

| Binary bits | DNA Nucleobases |
|---|---|
| 00 | A |
| 11 | T |
| 01 | C |
| 10 | G |

The rest of the paper is organized as follows: Section 2 presents an overview of literature techniques. The proposed technique is explained in Section 3. In Section 4, three examples are presented to illustrate how the encryption is done using the proposed method and to show the power of this method. Finally, the conclusion of the work is given in Section 4.

## II. LITERATURE REVIEW

Using DNA for information security has been studied in the literature, and different techniques have been designed and implemented [16]. These techniques used steganography, substitution and transposition in securing information [16]. Some of these techniques required costly laboratories, equipment, and preparations [17]. Pseudo DNA cryptography techniques have compensated this issue by simulating DNA computing methods such as PCR and central dogma, and then adding artificial sequences to become more related to digital computing [5], [17]. Moreover, DNA cryptography has been studied to enhance classical symmetric encryption techniques such as Vigenere, and Playfair ciphers [18], [19]. As well as, it has been combined with modern ciphering techniques such as DES and AES to empower the resulting cipher [17], [20].

Among the starters of DNA steganography is [21] in 1999, were DNA steganography used for hiding data into microdots. Later, many researchers have used DNA for hiding information such as [22], [23], where in the former a secret primer sequence is used to uncover information from DNA. The later has kept the functionality of the original DNA while hiding information in DNA sequences [21]. [16] has doubled the amount of information hidden by using both DNA strands, taking into consideration strands dependencies, and generating a key with the same size of the reference strand using the other strand.

Using DNA in cryptography has been first tested in 1999. [24] has designed a cryptosystem based on one-time pad implemented using two methods: The first one generates one-time pad using DNA substitution technique. In this technique, the key is chosen from a library of the massive number of long DNA sequences, each can map a plain word into a cipher word randomly. The second technique is based on Bit-wise XOR with shared secret one-time pad random key. The cipher in [7] is generated in two phases where the first phase uses a secret key, and the final phase uses DNA sequences generated based on random keys.

Encoding tables-based researches are extensively found in literature, where encoding tables for DNA sequences are used to encode binary data. Many authors have enhanced this approach by using dynamic encoding tables [25] rather than using pre-defined sequences from international libraries [26] such as European Bioinformatics Institute (EBI) database. The technique in [25] is among the DNA cryptography techniques which is based on encoding tables generated dynamically for character set representations. The authors have proposed an encryption and decryption algorithm based on dynamic encoding table where each letter is encoded with four DNA nucleotides digits (A, T, C, G). At the beginning of each encoding session, different codes are generated in the encoding table. After that, two halves of the plaintext are encrypted using two encoding tables, one generated by the sender and the other half is encoded by the receiver generated table. A similar approach has been proposed in [3] where the data is transformed into Unicode then into Hexadecimal digits in order to be converted into a DNA sequence combined with randomly generated DNA key.

Securing cloud information has got huge attention of researchers because the environment has more probability to be attacked [17], [27], [28]. Combining traditional encryption techniques such as DES, AES, and BlowFish with enhanced DNA cryptography has been proposed in [17]. In this technique, a DNA symmetric key technique has been proposed to encrypt client data on client sites before being transmitted into the cloud. This technique has been compared with other traditional techniques and showed that it outperforms with regards to cipher size and time efficiency. [27] has proposed a secure data encryption technique for IoT cloud platforms using DNA cryptography and Huffman coding for key generation.

DNA encryption using transposition is found in literature by many researchers [14], [29]. [14] has proposed a parabolic transposition in which circular data arrangement is used, then converted into DNA sequence and is sent to the receiver as the cipher. This algorithm depends on using the number of column and rows as encryption secret key. Authors of [29] have proposed an encryption algorithm by using spiral transposition where data is arranged using 8X8 matrix. Then each byte of binary data is substituted with DNA four nucleotide sequence according to pre-generated 256-entry coding table which also acts as the secret key.

DNA encoding with permutation and chaos techniques has been used in efficient image encryption [11], [15], [30]. Both substitution and permutation components have been found in [11] to encrypt images. A hyper- chaotic system is proposed to generate a random sequence to be used in nucleotide substitution and in generating hyper-image of permutation matrices. Image data in [15] is transferred into a DNA sequence based on random encoding, then a self-adaptive permutation is used to generate the final cipher image. In [30], images are transformed into one-dimensional sequence, then pixels are shuffled using chaotic and DNA based permutation, afterward a DNA-diffusion process is used to generate the cipher. Efficiency and simplicity of DNA and Elliptic Curve (EC) cryptography are combined with [31]. In this technique, image pixels are first mapped into EC points, each point and the random key values are converted into DNA sequence, then encrypted using a DNA chosen operation to generate the cipher. In [6] DNA steganography has been improved using Hyperelliptic curve cryptography, where both cover and secret images are converted into DNA nucleotides, based on nucleotides triple encoding table, and combined. Then, hyperelliptic curve encryption is used to encrypt the combined image after mapping its pixel values into points.

A DNA-Based Cryptographic Key Generation is presented by [32] for generating a strong cryptographic key that is applicable for symmetric ciphering applications.

## III. PROPOSED METHOD

This paper proposes a novel cipher method that emerges the DNA computing, random permutations, and the concept of transposition to obtain a very high complexity of encryption. It is a symmetric block cipher method needs one initial key that is used to generate number of random permutations equal to the number of input plaintext block. The random permutation size is four times of the data block size. The input data (plaintext) block is converted to DNA base and we consider it as a DNA strand. Converting plaintext to DNA base is done by taking each character and convert its ASCII value to 8-bits, then each two successive bits is converted into its equivalent decimal value that each value represents DNA base as follows (00 to A, 01 to C, 10 to G, 11 to T).

Now the plaintext block is representing the DNA strand. For each DNA strand, all its elements are rearranged (permuted) according to the generated random permutation. The result represents the cipher DNA strand (ciphertext block). The description of random permutation generator, encryption, and decryption are explained in this paper.

### A. Permutation Generation

Many existing methods, offering producing random numbers which depend on a seed in order to generate these numbers. In this paper, the method String Base Random Permutation (SBRP) [10], is a patent that is used to generate random permutations as many as number of DNA strands of input data.

### B. Encryption Method

Data is converted to DNA base in order to encrypt it. Reading plaintext and convert each character into ASCII value and then convert the ASCII value to 8-bits which is called a byte. Take each byte and divide it into four 2-bits and then convert it into values of the range of 0 to 3. These values are the DNA bases known as: adenine (A), cytosine (C), guanine (G) and thymine (T). The DNA Encryption process involves three steps:

Step 1: generate random permutations using input key

Step 2: convert input data (plaintext) to DNA strand (block)

Step 3: encrypt DNA strand.

Step 3 can be described by the cipher algorithm

### C. Cipher algorithm (encryption)

Given the following

N: permutation/DNA strand size

P: permutation

DNAB: block of DNA strand

DNAC: block of DNA ciphered strand
For each DNA strand
For i = 1 to N
DNAC[i] = DNAB[P[i]]

### D. Decryption Method

The DNA cipher strand is rearranged by an inverse permute using the generated permutation. The decryption process involves three steps:

Step 1: generate random permutations using the same input key that is used in encryption

Step 2: decrypt DNA cipher strand.

Step 3: convert DNA strand to character by taking every four successive DNA bases and convert it to one byte.

Step 2 can be described by the cipher algorithm (decrypt)

### E. Cipher algorithm (decrypt)

Given the following
N: permutation/DNA strand size
P: permutation
DNAC: block of DNA ciphered strand
DNAB: block of DNA strand
For each DNA ciphered strand
For i = 1 to N
DNAB[P[i]] = DNAC[i]

## IV. ILLUSTRATED EXAMPLES AND RESULTS

Let takes three examples to show the process and the power of this proposed encryption method.

### A. Example 1

Initial key: confidentiality
Permutation size: 40
Plaintext:
"The word cryptography comes out from ancient Greek. It is a mishmash of two word"
Plaintext block size = 10
DNA strand (block) size = 40
Number of permutations needed = 8
All Permutations

P1: 31 23 15 33 19 30 17 13 27 18 9 14 10 7 2 1 5 26 29 34 4 0 24 3 20 8 11 28 16 22 21 6 25 35 37 39 12 38 36 32

P2: 5 20 2 11 10 16 9 21 27 25 17 37 19 12 15 36 31 32 23 38 33 39 30 35 13 6 18 22 14 28 7 8 1 3 26 0 34 4 29 24

P3: 33 23 30 31 13 15 18 19 14 17 7 27 1 9 26 10 34 2 29 5 24 20 4 11 0 16 3 21 8 25 28 37 22 12 6 36 35 32 39 38

P4: 34 0 26 3 1 8 7 28 14 22 18 6 13 35 30 39 33 38 23 32 31 36 15 12 19 37 17 25 27 21 9 16 10 11 2 20 5 24 29 4

P5: 31 23 15 33 19 30 17 13 27 18 9 14 10 7 2 1 5 26 29 34 4 0 24 3 20 8 11 28 16 22 21 6 25 35 37 39 12 38 36 32

P6: 5 20 2 11 10 16 9 21 27 25 17 37 19 12 15 36 31 32 23 38 33 39 30 35 13 6 18 22 14 28 7 8 1 3 26 0 34 4 29 24

P7: 33 23 30 31 13 15 18 19 14 17 7 27 1 9 26 10 34 2 29 5 24 20 4 11 0 16 3 21 8 25 28 37 22 12 6 36 35 32 39 38

P8: 34 0 26 3 1 8 7 28 14 22 18 6 13 35 30 39 33 38 23 32 31 36 15 12 19 37 17 25 27 21 9 16 10 11 2 20 5 24 29 4

**DNA Plaintext (8 blocks/strands)**

GGGAGCCAGCGGACAAGTGTGCTTGTACGCGAACA
AGCAT
GTACGTCGGTAAGTGAGCTTGCGTGTACGCAGGTA
AGCCA
GTCGACAAGCATGCTTGCTGGCGGGTATACAAGCT
TGTGG
GTGAACAAGCGCGTACGCTTGCTGACAAGCAGGCT
CGCAT
GCCGGCGGGCTCGTGAACAAGAGTGTACGCGGGCG
GGCCT
ACTCACAAGACGGTGAACAAGCCGGTATACAAGCA
GACAA
GCTGGCCGGTATGCCAGCTGGCAGGTATGCCAACA
AGCTT
GCGCACAAGTGAGTGTGCTTACAAGTGTGCTTGTA
CGCGA

**DNA Ciphertext Block (8 blocks/strands)**

CAATGGACTGAAGAGGGCCGGGCGTATGCGCGTAT
CAACA
AGATCGTAGCGCTGGGTAAGTGCTATAGCCGGCGG
TAATC
GGCAGGTAACTGCAGCTCAACTGTGCTTATCGTGG
GTAGG
TATATGCACAGCGGGTGAGACCCTCAGGAAAGTGG
TCCCC
AAGTGAGTTTGAGGCCGGCGGGCCGGCGCACGTGA
CCGCG
GGTCCATAAAACTGAGCCAGCATAAAAGCACACGA
GAGAC
GGCAAGAAGCAGCGGCTTCGCTACGCCTCTTGCGG
GAATT
CAACAGAACTGTAGGATGGGCCTTCTGGAGGATGG
TCTCT

**Ciphertext characters**

?¬-I?{?6ê??äI«?WJëU
??¥?¥?Fáf?LâRLDb!? ?iöq?~j

**Ciphertext (ASCII value in decimal)**

67 161 224 138 150 166 206 102 205 4 35 108 153 234 194 231 50 90 107 13 164 172 30 73 208 123 159 54 234 202 204 228 73 171 136 87 74 2 235 85 11 139 248 165 166 165 166 70 225 102 173 76 1 226 82 76 2 68 98 33 164 32 146 105 246 113 151 126 106 15 65 32 123 40 234 95 122 40 235 119

**Plaintext (ASCII value in decimal)**

84 104 101 32 119 111 114 100 32 99 114 121 112 116 111 103 114 97 112 104 121 32 99 111 109 101 115 32 111 117 116 32 102 114 111 109 32 97 110 99 105 101 110 116 32 71 114 101 101 107 46 32 73 116 32 105 115 32 97 32 109 105 115 104 109 97 115 104 32 111 102 32 116 119 111 32 119 111 114 100

The correlation between the elements of the plaintext and ciphertext is 0.0442 which is very low, and Figure 2 shows this fact for this example. Also, the repetition of the same plaintext element does not have to be the same ciphertext element. For example, the letter ‘e’ appears in plaintext 5 times at positions 3, 26, 42, 48, and 49 while the encrypted this letter appears in the ciphertext 5 times with different values. Table II shows the letter ‘e’ in the plaintext and its corresponding values in the ciphertext.

**Table II: Values of the letter ‘e’ in plaintext and ciphertext**

| Position letter ‘e’ | 3 | 26 | 42 | 48 | 49 |
|---|---|---|---|---|---|
| Value of letter ‘e’ in plaintext | 101 | 101 | 101 | 101 | 101 |
| Value of ciphertext | 228 | 123 | 139 | 70 | 225 |

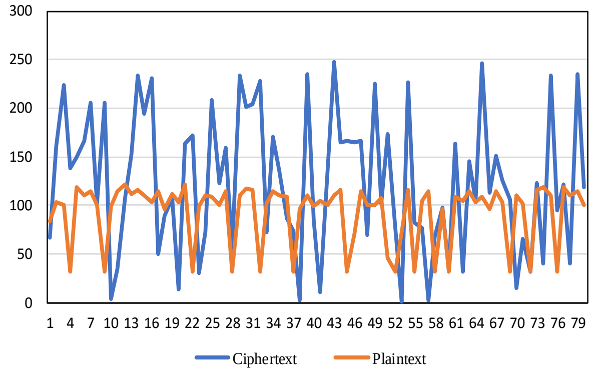


**Fig. 2. Matching between ciphertext and plaintext elements (Example 1)**

To show the power of this novel method, let takes another example

## B. Example 2

Initial key: This test
Permutation size: 40
Plaintext block size = 10
DNA strand (block) size = 40
Number of permutations needed = 8

**Plaintext**

“This test This test This test This test This test This test This test This test “

**Ciphertext**

???*f? ?&?°J ?9â)+z??(V+-
¡;?f°¢æZû:?  (Oªn"Ü£?ç??$??J`âhú=?mZ????JtòJ?²(±

The plaintext contains a repetition of the string “This test “, eight times in order to make the plaintext. Also, the same string “This test”, is used as an initial key for this example. The result was amazing because the correlation between the ciphertext and the plaintext (-0.094) which is very low. Figure 3 shows there is no pattern for the ciphertext of the repeated words in the plaintext.

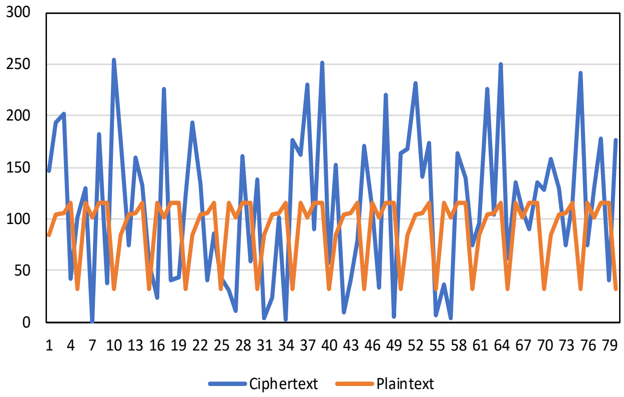


**Fig. 3. Matching between ciphertext and plaintext elements (Example 2)**

## C. Example 3

Changing one bit in the initial key.
Let us take the same key (This test) and change only one bit to become (Thhs test), and then encrypt the same plaintext as shown in Table III. The bit “1” for the old key is changed to bit “0” for the new key, both bits are written in red and bold.

**Table III: keys representation in binary**

| key | text | binary representation |
|---|---|---|
| old key | This test | 01010100 01101000 0110100**1** 01110011 0100000 01110100 01100101 01110011 01110100 |
| new key | Thhs test | 01010100 01101000 0110100**0** 01110011 0100000 01110100 01100101 01110011 01110100 |

The new Key: Thhs text
All Perms:
P1: 20 23 28 34 37 32 39 38 15 35 29 27 21 22 30 33 14 25 11 10 5 8 19 3 9 2 24 26 7 12 16 0 31 4 6 13 36 1 17 18
P2: 14 9 30 24 21 7 29 16 15 31 39 6 37 36 28 17 20 18 23 1 34 13 32 4 38 0 35 12 27 26 22 2 33 3 25 8 10 5 11 19
P3: 31 26 6 12 36 0 17 4 18 13 1 34 23 32 20 38 28 35 37 27 39 22 15 33 29 25 21 10 30 11 14 19 9 5 24 8 7 3 16 2
P4: 21 23 30 1 14 18 9 17 24 36 7 6 16 31 2 26 3 12 8 0 5 4 19 13 11 34 10 32 25 38 33 35 22 27 39 37 15 28 29 20
P5: 3 11 2 10 16 25 7 33 24 22 9 39 14 15 30 29 21 20 23 28 1 37 18 27 17 35 36 38 6 32 31 34 26 13 12 4 0 5 8 19
P6: 39 38 15 35 29 27 20 37 28 1 23 18 21 17 30 36 14 6 9 31 24 26 7 12 16 0 2 8 3 19 11 5 10 4 25 13 33 34 22 32
P7: 22 39 32 38 34 35 13 27 4 37 5 1 19 18 8 17 0 36 12 6 26 31 24 9 7 14 16 30 2 21 3 23 11 28 10 20 25 29 33 15
P8: 12 24 0 7 8 16 19 2 5 3 4 11 13 10 34 25 32 33 22 15 39 29 38 20 35 28 27 23 37 21 1 30 18 14 17 9 36 31 6 26

**Data (Plaintext):**

This test This test This test This test This test This test This test This test

**DNA Plaintext**

GGGAGCCAGCCGGTATACAAGTGAGCGGGTATGTG AACAA GGGAGCCAGCCGGTATACAAGTGAGCGGGTATGTG AACAA GGGAGCCAGCCGGTATACAAGTGAGCGGGTATGTG AACAA GGGAGCCAGCCGGTATACAAGTGAGCGGGTATGTG AACAA GGGAGCCAGCCGGTATACAAGTGAGCGGGTATGTG AACAA GGGAGCCAGCCGGTATACAAGTGAGCGGGTATGTG AACAA GGGAGCCAGCCGGTATACAAGTGAGCGGGTATGTG AACAA GGGAGCCAGCCGGTATACAAGTGAGCGGGTATGTG AACAA

**DNA Cipher Block:**

TCCATGGGTGAATAAGAAAGGGTGGCGGGCAGCTA CAGAC CATTACGCAGAAGTGGATCAAGAAAGTGACGCGGG GTGGC CCACATGAAGGTACAGACGTAGTGGCGAAGGGTAG CGATG AAAATGGCACGGCAGAGACGAGGGGGTTCAGTGAC TCATG AGGGACGCACAGGTGTGGGACACAGCGAATAATAT CGTGG CCGGTTCGGAGAAAAGGTGTCGACGGTCGGAAAAC ATAGG AGGAGCAGAAGGACCAGTTGATGTGAGATCGTGAG CCCAG GAACCGAAGATGGGTACGGCATAGGTAGCTTCACA GAGGG

**Ciphertext (ASCII value in decimal)**

212 234 224 194 2 174 154 146 113 33 79 25 32 186 52 32 46 25 170 233 81 56 43 18 27 46 152 42 201 142 0 233 26 72 134 42 175 75 135 78 42 25 18 187 168 68 152 48 205 186 90 246 136 2 187 97 173 160 4 202 40 146 10 20 190 59 136 219 137 82 129 96 142 172 105 50 178 125 18 42

**Plaintext (ASCII value in decimal)**

84 104 105 115 32 116 101 115 116 32 84 104 105 115 32 116 101 115 116 32 84 104 105 115 32 116 101 115 116 32 84 104 105 115 32 116 101 115 116 32 84 104 105 115 32 116 101 115 116 32 84 104 105 115 32 116 101 115 116 32 84 104 105 115 32 116 101 115 116 32 84 104 105 115 32 116 101 115 116 32

**Ciphertext**

?êà????q!Oº4 .ªéQ8+.?*É? éH?*?K?N*»?D?0?ºZö?»a? ?(? ?;???R?`?¬i2²}*

The produced ciphertext was drastically changed compared with the first ciphertext that obtained by using the original key. Table IV shows the ASCII values of the corresponding ciphertext elements after changing one bit of this key. The correlation between the plaintext and the new ciphertext is -0.0533 which is very low. Figure 4 shows matching between the two ciphertext and the same plaintext. In addition, the cipher algorithm has a non-linearity which satisfies confusion. After changing one bit of the used key, the new different permutations are generated, and the diffusion is very clear that the ciphertext has been extremely changed as shown in Table IV and Figure 4.

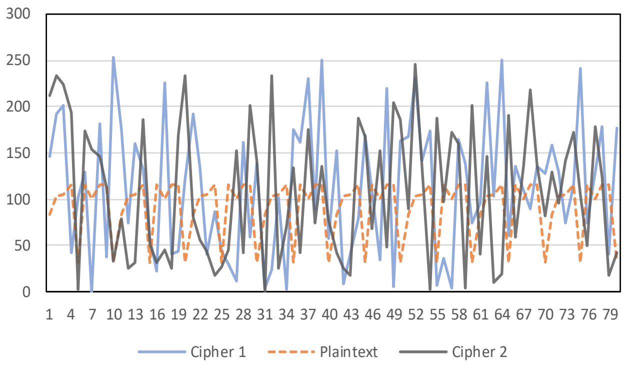


**Fig. 4. Matching between the two ciphertexts and same plaintext elements (Example 3). Cipher 1 using the key (This test). Cipher 2 using the key (Thhs test).**

The result of the encryption for the given examples is shown in Table V. The produced ciphertext in the table show all the printable elements while it might contain another element which is not printable because the ciphertext has all the possible elements for the ASCII code (0-255) decimal value. The proposed method encrypts the plaintext element and produce an element with a possible value 0-255. The plaintext in row 2 is encrypted with the given key and the same plaintext in row 3 is encrypted by taking the same key in row 2 and changing only one bit in order to show the power of the result. The proposed encryption method produced a ciphertext in row 3 which is completely different from the ciphertext in row 2. Changing one bit of the key and using it to encrypt the same plaintext produced a considerable changing in the ciphertext. Also, as shown in table V there is a weak correlation between the plaintext and the ciphertext. The correlation is weak with a value 0.044, -0.094, and -0.053 for row1, row2, and row3 respectively.

**Table IV: Two ciphertexts of same plaintext using one key and changing one bit in the same key to produce another ciphertext (comparing the ASCII code of corresponding ciphertext)**

| Cipher 1 | 146 | 193 | 202 | 4 2 | 102 | 129 | 0 | 182 | 3 8 | 254 | 176 | 7 4 | 160 | 132 | 57 | 2 3 | 226 | 4 1 | 4 3 | 122 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Plain** | 8 4 | 104 | 105 | 115 | 3 2 | 116 | 101 | 115 | 116 | 2 | 8 4 | 104 | 105 | 115 | 32 | 116 | 101 | 115 | 116 | 3 2 |
| Cipher* 2 | 212 | 234 | 224 | 194 | 2 | 174 | 154 | 146 | 113 | 3 3 | 7 9 | 2 5 | 3 2 | 186 | 52 | 3 2 | 4 6 | 2 5 | 170 | 233 |

| | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 193 | 134 | 40 | 86 | 43 | 30 | 11 | 161 | 59 | 138 | 4 | 24 | 102 | 2 | 176 | 162 | 230 | 90 | 251 | 58 |
| 84 | 104 | 105 | 115 | 32 | 116 | 101 | 115 | 116 | 32 | 84 | 104 | 105 | 115 | 32 | 116 | 101 | 115 | 116 | 32 |
| 81 | 56 | 43 | 18 | 27 | 46 | 152 | 42 | 201 | 142 | 0 | 233 | 26 | 72 | 134 | 42 | 175 | 75 | 135 | 78 |

| | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 152 | 9 | 40 | 79 | 170 | 110 | 34 | 220 | 5 | 163 | 168 | 231 | 141 | 174 | 7 | 36 | 4 | 164 | 139 | 74 |
| 84 | 104 | 105 | 115 | 32 | 116 | 101 | 115 | 116 | 32 | 84 | 104 | 105 | 115 | 32 | 116 | 101 | 115 | 116 | 32 |
| 42 | 25 | 18 | 187 | 168 | 68 | 152 | 48 | 205 | 186 | 90 | 246 | 136 | 2 | 187 | 97 | 173 | 160 | 4 | 202 |

| | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 74 | 96 | 226 | 104 | 250 | 61 | 136 | 109 | 90 | 136 | 128 | 158 | 129 | 74 | 116 | 242 | 74 | 130 | 178 | 40 | 177 |
| 32 | 84 | 104 | 105 | 115 | 32 | 116 | 101 | 115 | 116 | 32 | 84 | 104 | 105 | 115 | 32 | 116 | 101 | 115 | 116 | 32 |
| 202 | 40 | 146 | 10 | 20 | 190 | 59 | 136 | 219 | 137 | 82 | 129 | 96 | 142 | 172 | 105 | 50 | 178 | 125 | 18 | 42 |

** Ciphertext after one bit changed in key*

**Table V: Encryption plaintext using a key with its correspond ciphertext**

| N o | Plaintext | key | Ciphertext | Correlation |
|---|---|---|---|---|
| 1 | The word cryptography comes out from ancient Greek. It is a mishmash of two word | confidentiality | ?¬-I?{?6ê??äI«?WJëU ??¥?¥?Fáf?LâRLDb!? ?iöq?~j | 0.044 |
| 2 | This test This test This test This test This test This test This test This test | This test | ???*f? ?&?°J ?9â)+z??(V+- ¡;?f°¢æZû:? (Oªn"Ü£?ç??$??J`âhú=?mZ????JtòJ?²(± | -0.094 |
| 3 | This test This test This test This test This test This test This test This test | Thhs test | ?êà????q!Oº4 .ªéQ8+.?*É? éH?*?K?N*»?D?0?ºZö?»a? ?(? ?;???R?`?¬i2²}* | -0.053 |

## V. CONCLUSION

A novel DNA encryption introduced in this paper. This method introduced the concept of the traditional well known of the transposition cipher and because of the drawback of this classical method, the DNA computing and the random permutation are emerged to produce this novel method. The drawback of the classical transposition method is related to the nature of the natural language that has a statistical distribution of the language characters which is kept after encryption. This helps the cryptanalysis to break the cipher in an easy way. The new method does not maintain the statistical analysis of the characters after encryption even for the same letter appears more than one time in the plaintext. This method produces ciphertext that is hard to break by cryptanalysis using the statistical analysis. Also, another method such as brute-force need to work for N! where N is the size of DNA block/permutation.

## REFERENCES

1. Mandrita Mondal, Kumar S. Ray, "Review on DNA Cryptography", arXiv:1904.05528 , pp. 1-31, 2019.
2. S. Mathew, G. Saranya, "Advanced biometric home security system using digital signature and DNA cryptography", in International Conference on Innovations in Green Energy and Healthcare Technologies (IGEHT), Coimbatore, India, 2017.
3. Bahubali Akiwate, Latha Parthiban, "A Dynamic DNA for Key-based Cryptography," in International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS), Belgaum, India, India, 2018.
4. Ashish Kumar Kaunda, A.K Verma, "DNA Based Cryptography: A Review," International Journal of Information and Computation Technology, vol. 4, no. 7, pp. 693-698 , 2014.
5. El tantawi, Mostafa Reda; Abd El Shafouk Ali, and Amr Mohamed, "A proposed Algorithm using DNA-based Cryptography and Steganography Techniques," Computing & Information Systems, vol. 14, no. 3, pp. 43-55, 2010.
6. P. Vijayakumar, V. Vijayalakshmi, and G. Zayaraz, "An Improved Level of Security for DNA Steganography Using Hyperelliptic Curve Cryptography," Wireless Personal Communications, vol. 89, no. 4, p. 1221–1242, 2016.
7. Bonny B. Raj, J. Frank Vijay, and T. Mahalakshmi, " Secure Data Transfer through DNA Cryptography using Symmetric Algorithm," International Journal of Computer Applications, vol. 133, no. 2, pp. 19-23, 2016.
8. Hariram S, Dhamodharan R, "A Survey on DNA Based Cryptography using Differential Encryption and Decryption Algorithm," Journal of Electronics and Communication Engineering (IOSR-JECE), vol. 10, no. 5, pp. 14-18, 2015.
9. G.R. Blakley, "Twenty Years of Cryptography in the Open Literature," in Proceedings of the 1999 IEEE Symposium on Security and Privacy, Oakland, CA, USA, 1999.
10. Shakir M. Hussain Al-Farraji, "Method and System Generating String Base Random Permutation". USA Patent 10.496.377, 3 12 2019.
11. Shine Zhang, Tiegang Gao, "An image encryption scheme based on DNA coding and permutation of hyper-image," Multimedia Tools and Applications, vol. 75, no. 24, p. 17157–17170, 2016.
12. Md. Fakruddin, Khanjada Shahnewaj Bin Mannan, Abhijit Chowdhury, Reaz Mohammad Mazumdar, Md. Nur Hossain, Sumaiya Islam, and Md. Alimuddin Chowdhury, "Nucleic acid amplification: Alternative methods of polymerase chain reaction," Journal of Pharmacy & Bioallied Sciences, vol. vol.5(4), p. 245–252, Oct-Dec 2013.
13. Adleman LM, "Molecular computation of solutions to combinatorial problems," Science, vol. 266, no. 11, pp. 1021 - 1024, Nov. 1994.

14. Shipra Jain, Vishal Bhatnagar, "A Novel Ammonic Conversion Algorithm for Securing Data in DNA using Parabolic Encryption," Information Resources Management Journal, vol. 28, no. 2, pp. 20-31, 2015.
15. Chen Junxin, Zhu Zhi-liang, Zhang Li-bo, ZhangYushu, Yang Ben-qiang, "Exploiting self-adaptive permutation-diffusion and DNA random encoding for secure and efficient image encryption," Signal Processing, vol. 142, pp. 340-353, 2017.
16. Samiha Abdelrahman Mohammed Marwan, Ahmed Shawish, Khaled Nagaty, "Utilizing DNA Strands for Secured Data-Hiding with High Capacity," International Journal of Interactive Mobile Technologies, vol. 11, no. 2, pp. 88-98, 2017.
17. Manreet Sohal, Sandeep Sharma, "BDNA-A DNA inspired symmetric key cryptographic technique to secure cloud computing," Journal of King Saud University – Computer and Information Sciences, 2018.
18. Ahmad Sharaieh, Afaf Edinat, Shakir AlFarraji, "An Enhanced Polyalphabetic Algorithm on Vigenerecipher with DNA-Based Cryptography," in IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA), Aqaba, Jordan, 2018.
19. Mona Sabry, Mohammed Hashem, Nazmy Taymoor, and Mohamed Essam Khalifa, "A DNA and Amino Acids-Based Implementation of Playfair Cipher," International Journal of Computer Science and Information Security, vol. 8, no. 3, pp. 19219-19226, 2010.
20. Mona Sabry, Mohammed Hashem, Nazmy Taymoor, and Mohamed Essam Khalifa, "Design of DNA-based Advanced Encryption Standard (AES)," in IEEE Seventh International Conference on Intelligent Computing and Information Systems (ICICIS), Cairo, Egypt, 2015.
21. C. Clelland, V. Risca and C. Bancroft, "Hiding messages in DNA microdots," Nature, vol. 399, p. 533–534, 1999.
22. Leier A., Richter C., Banzhaf W., Rauhe H., "Cryptography with DNA binary strands," Biosystems, vol. 57, no. 1, pp. 13-22, 2000.
23. Chin-Chen Chang, Tzu-Chuen Lu, and Ya-Fen Chang, "Reversible data hiding schemes for deoxyribonucleic acid (DNA) medium," International Journal of Innovative Computing, Information and Control ICIC, vol. 3, pp. 1145-1160, 2007.
24. A. Gehani, T. LaBean and J. Reif, "DNA-based Cryptography," in Aspects of Molecular Computing, Berlin, Heidelberg, Springer, Berlin, Heidelberg, 2003, pp. 167-188.
25. Noorul Hussain, UbaidurRahman, Chithralekha Balamurugan, and Rajapandian Mariappan, "A Novel DNA Computing based Encryption and Decryption Algorithm," in International Conference on Information and Communication Technologies (ICICT 2014), Kochi, India, 2015.
26. A. R. Pushpa, "A new technique for data encryption using DNA sequence," in International Conference on Intelligent Computing and Control (I2C2), Coimbatore, India, 2017.
27. N. Harish Kumar, Rajshekhar M. Patil, G. Deepak, and B. M. Murthy "A Novel Approach for securing data in IoTcloud Using DNA Cryptography and Huffman Coding Algorithm," in International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), Coimbatore, India, 2017.
28. Prajapati Ashishkumar B., Prajapati Barkha, "Implementation of DNA Cryptography in Cloud Computing and Using Socket Programming," in International Conference on Computer Communication and Informatics (ICCCI -2016), Coimbatore, INDIA , 2016.
29. Shipra Jain, Vishal Bhatnagar, "A Novel DNA Sequence Dictionary method for Securing Data in DNA using Spiral Approach and Framework for DNA Cryptography," in IEEE International Conference on Advances in Engineering &Technology Research (ICAETR - 2014), Unnao, India, August 01-02, 2014.
30. Rasul Enayatifar, Abdul Hanan Abdullah, Ismail Fauzi Isnin, Ayman Altameem, and Malrey Lee, "Image encryption using a synchronous permutation-diffusion technique," Optics and Lasers in Engineering, vol. 90, pp. 146-154, 2017.
31. Salma Bendaoud, Fatima Amounas, and El Hassan El Kinani, "A New Image Encryption Scheme Based on Enhanced Elliptic Curve Cryptosystem Using DNA Computing," in NISS19 Proceedings of the 2nd International Conference on Networking, Information Systems & Security, Rabat, Morocco, 2019.
32. Shakir M. Hussain, and Hussein Al-Bahadili, "A DNA-Based Cryptographic Key Generation Algorithm", The 2016 International Conference on Security and Management (SAM'16), Las Vegas, 2016.

## AUTHORS PROFILE

**Shakir M. Al-Farraji,** (shussain@uop.edu.jo) He received his B.A. degree in statistics from University of Al-Mustansiriyah, Iraq, in 1976 and M.Sc. degree in Computing and Information Science from Oklahoma State University, USA, in 1984. In 1997 he received his Ph.D. degree in Computer Science from University of Technology, Iraq. From 1997 to 2008 he was a faculty member at Applied Science University, Jordan. Currently, he is an associate professor at Petra University, department of computer science, Faculty of Information Technology, Jordan. His research interest covers encryption, key generation, authentication, and data compression. He received a USPTO patent for generating random permutation and currently working on DNA based cryptography in encryption, authentication, and digital signature. He is a member of ACM.

.

**Huda K. Saadeh,** (hsaadeh@uop.edu.jo) She received her B.A. degree in Computer Science from University of Jordan, Jordan, in 2000 and M.Sc. degree in Image Processing from University of Jordan, Jordan, in 2006. In 2018 she received her Ph.D. degree in Information and Network Security from University of Jordan, Jordan. From 2001 to 2018 she was a faculty member at University of Petra, Information Technology Faculty, Computer Science Department. Currently she is an assistant professor at University of Petra, department of Information Security, Faculty of Information Technology, Jordan. Her research interest covers encryption, intrusion detection, IoT, and Network Security, Networking Architectures.