





DNA Based Cryptography

Archana Gahlaut^(✉) , Amit Bharti , Yash Dogra ,
and Puneet Singh 

Department of Computer Science, Atma Ram Sanatan Dharma College,
University of Delhi, New Delhi, India
archana.gahlaut@gmail.com, amitbharti43@gmail.com,
yashdogra31@gmail.com, puneet.singh0902@gmail.com

Abstract. A new emerging research topic in the field of Information storage, security and cryptography is DNA based cryptography. DNA is known to carry information from one generation to other and is turning out to be very promising for cryptography. The storage capacity, vast parallelism of DNA are used for cryptographic purposes. In this paper, we will talk about progress of DNA cryptography, discuss DNA computing, and propose a method for DNA based cryptography through 3 phases where we will first encrypt the plaintext through our proposed encryption algorithm and then prepare a desired DNA sequence ready to send.

Keywords: DNA based cryptography · DNA computing · AYP algorithm

1 Introduction

As the security threats are increasing day by day, information security is a major concern today. Data needs to be encrypted while transmitting in order to ensure the security of data. Cryptography is the practice of hiding information. Cryptographic techniques help in ensuring the security of such sensitive information. DNA cryptography is a new and promising technique in the field of cryptography. In DNA cryptography, information carriers are the DNA nucleotides (denoted by the letters A, C, G and T). The main advantage of DNA cryptography is high storage capacity of DNA, vast parallelism, lower power consumption with extraordinary performance.

1.1 DNA (Deoxyribonucleic Acid)

DNA is a molecule which is present nearly in all living organisms. It transmits the genetic information required for the growth, development, and functioning of all living organisms (including viruses). DNA is found mostly in the nucleus of the cell (Nuclear DNA) but a small proportion of it is also located in the mitochondria (Mitochondrial DNA). The genetic information stored in DNA molecule is in the form of code which consists of four chemical bases namely Adenine (A), Guanine (G), Cytosine (C), and Thymine (T). The above mentioned chemical bases pair up with each other in a way

such that each base is having a specific partner. So overall we have two pairs which are as follows:

1. 'A' always pair up with 'T' and vice versa.
2. 'C' always pair up with 'G' and vice versa.

And they form units which are known as Base Pairs. Each base is also linked to two other molecules namely Sugar molecule, Phosphate molecule. Unitedly a base along with the sugar and the phosphate molecule are known as Nucleotide.

1.2 Structure of DNA

Nucleotides are organized in two long strands which form the spiral known as Double Helix. Thus the structure of DNA is commonly known as the Double Helix Structure. The Double Helix is somewhat similar to a ladder in which the sugar and phosphate molecules are forming the vertical side pipes and the base pair forming the steps of the ladder (Fig. 1).

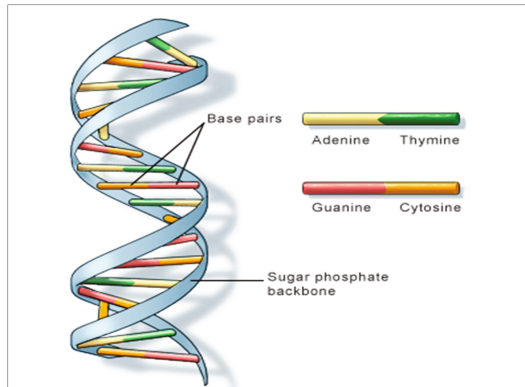


Fig. 1. Double Helix Structure of DNA

1.3 DNA Computing

DNA computing is an emergent field of the computing in which we use the concept of biochemistry, molecular biology hardware along with DNA in place of conventional silicon based computer technology. Leonard Adleman of the University of Southern California in 1994 was the first one known for the creation and initiation of this field. Adleman exhibits a proof-of-concept to solve the Seven-Point Hamiltonian path problem which is using DNA molecule as a means of computation [1].

In the year 2002, researchers of the Weizmann Institute of Science in Rehovot, Israel, uncovers a programmable molecular computing device made up of DNA molecules and enzymes in place of Silicon Microchips [2]. On April 28, 2004, Yaakov Benenson, Ehud Shapiro, Uri Ben-Dor, Binyamin Gil, and Rivka Adar at the Weizmann Institute declared in the journal Nature that they had developed a DNA computer attached to an input and output module which would be theoretically efficient of

identifying cancerous activity in a cell, and delivering an anti-cancer drug upon recognition [3]. In January 2013, researchers were able to store a set of Shakespearean sonnets, an audio file of Martin Luther King, Jr.'s speech "I Have a Dream" and a JPEG picture on DNA digital data storage [4]. In March 2013, researchers bring into existence a transistor (a biological transistor) [5].

2 DNA Cryptography (Related Work)

In 1994, Adleman used molecular computation to solve the combinatorial problems such as "Hamiltonian path" problem and laid the foundation of DNA computing [1].

In 2003, Jie Chen used DNA cryptographic approach which included one-time pad, molecular theory, and performed encryption and decryption of a 2-dimensional image [6].

In 2004, Ashish Gehani used molecular approach and one-time pad and laid the foundation of DNA cryptography [7]. According to Vernam and Shannon, who are the inventor of one-time pad, it has perfect secrecy. They proposed a method which uses DNA chip and one-time pad for encryption and decryption. Hence it is very difficult to guess any encrypted message for the adversary.

In 2013, Monica Borda and Olga Tornea proposed a DNA based cipher based on DNA indexing [8].

3 Experimental Work

3.1 AYP Algorithm

Input: Plaintext, Key String

Output: Cipher text

Rule to be followed to encrypt the given plaintext using AYP – algorithm

- (a) First create the key matrix
The key matrix here is of 5×5 order therefore we can accommodate only 25 alphabets of English letters (as a convention i and j are considered the same while encrypting).
 - To create the key matrix first both ALICE and BOB needs to agree on a secret key string.
 - After agreeing on a secret key string start writing the key diagonally in the matrix. (Starting from the main diagonal and filling the diagonals of lower triangle and then filling the diagonal of the upper triangle if key length is more than the length of the main diagonal. Also keep in mind that only distinct characters of the key string needs to be filled in the matrix.)
 - After filling the secret key sting the key matrix fill the remaining cell of the matrix (if any) in the alphabetical order starting form the first empty cell in the matrix.

We can use any pattern while filling the key Matrix but both sender and receiver must use the same pattern while creating the key matrix.

- (b) Locate the character of the plaintext in the key matrix one by one. And for each plaintext character we have to select 2 character from the matrix and these 2 character are the encrypted character for that particular plaintext character.

Here we have a total of 2 cases and further each case have sub-cases also.

Case 1. (Single diagonal possible) plaintext character is at one of the 4 corner position of the matrix (Fig. 2).

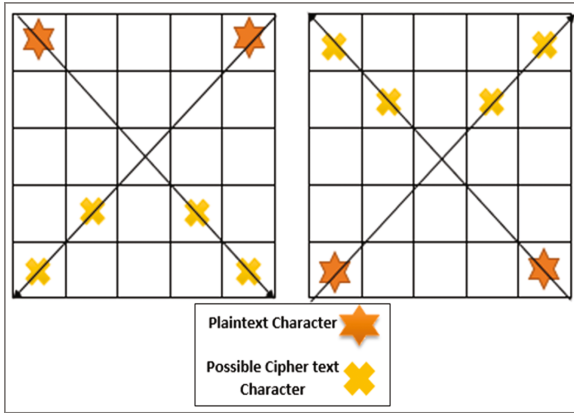


Fig. 2. Case 1

Case 2. (2 diagonals are possible)

Always move in one step away from the plaintext character on the diagonal.

2.1 Two diagonals are possible and both the diagonal are intersecting at the cell containing the plaintext character itself (Fig. 3).

Odd occurrence of the plaintext character means if the plaintext character comes for the

$$1\text{st } 3\text{rd } 5\text{th} \dots \dots \dots (2n + 1)\text{th}$$

Then we need to follow the 1st diagram

Even occurrence of the plaintext character means if the plaintext character comes for the

$$2\text{nd } 4\text{th } 6\text{th} \dots \dots \dots (2n)\text{th}$$

Then we need to follow the 2nd diagram.

Remark 1. 0th occurrence here has no meaning.

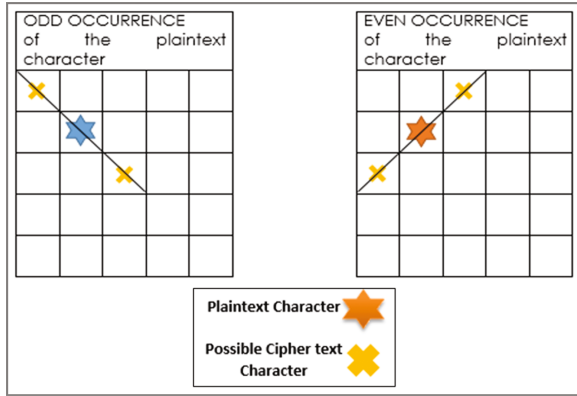


Fig. 3. Case 2.1

2.2 Two diagonal are still possible but one of the diagonals have only one character for that particular plaintext character.

Also we can say that here both the diagonal form right angle with the plain text character at the vertex of the right angle (Fig. 4).

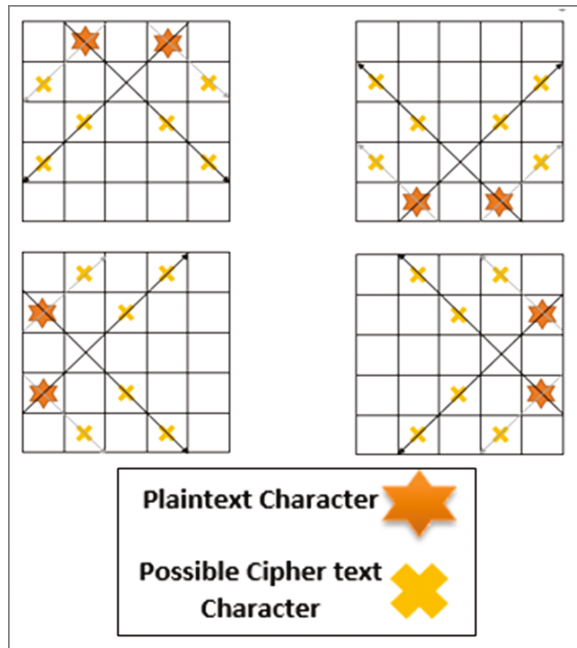


Fig. 4. Case 2.2

2.3 Exactly same case as above with one difference that here any one out of the two Diagonals can be chosen.

Also we can say that both diagonals form a right angle with plaintext character at the vertex of the right angle and also the plaintext character is in one of the following positions

- 1st row middle column.
- 5th row middle column.
- 1st column middle row.
- 5th column middle row (Fig. 5).

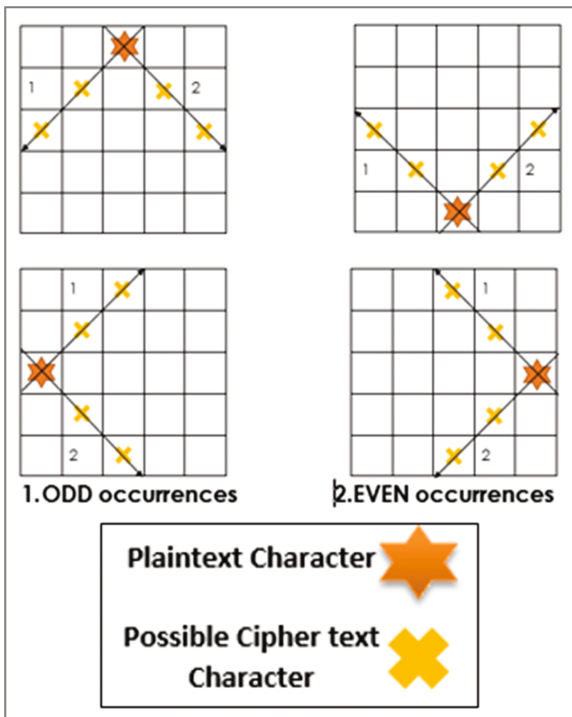


Fig. 5. Case 2.3

4 DNA Based Data Encryption and Hiding Using 3 Phases

Phase I. Encrypting the data using the above specified AYP algorithm.

In this we are encrypting the plaintext using the AYP algorithm to produce the intermediate text that is further encrypted in the upcoming phases.

For this we need following things

- (a) Plaintext: Data to be encrypted.
- (b) Secret key String: Both sender and receiver need to agree upon a secret string that is known only to them and this string is used to create the Key matrix.

So the basic idea or Aim of this phase is to encrypt the plaintext using the AYP algorithm thereby creating the intermediate text ready to be used in the upcoming phases.

Phase II. Increasing security (making resistant to statistical attack).

Basic idea behind introducing this phase is to disturb frequency of 2 letter string (digrams) and 3 letter string (trigrams) thereby making it resistant to statistical attacks.

Let's say we have a plaintext string as "eeste" and suppose that the letter "e" happens to be in one of the position marked in the Fig. 6.

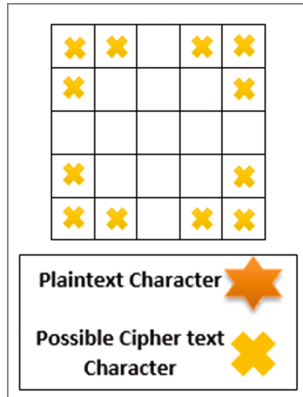


Fig. 6. If the plaintext character is in one of the positions marked above

Then doesn't matter - how many times the letter "e" comes OR whether it is the odd occurrence or even occurrence of letter same 2 letter will be used to encrypt the letter "e".

This happens so because in these cases we don't have any choice to choose from we have either only one diagonal OR two diagonals with one containing only one letter (character).

So if we don't apply phase II it will be very easy for the attacker to guess at least that the letter in the plaintext at these position must be same.

So we recommend applying well known traditional ciphers such as double-transposition/transposition cipher to the intermediate text produced after the Phase I just to jumble the intermediate text characters so that guessing won't be that easy.

Phase III. Hiding the encrypted data into the reference dna sequence using and insertion technique [9]

For this phase we will first convert the encrypted text obtained after the phase 2 to the Binary form. Then by using the Binary coding rule this binary form of data in converted into a DNA sequence (Table 1).

Table 1. Binary Coding Rule

BASE	BINARY CODE
A	00
C	01
G	10
T	11

Now we will use the above encrypted DNA sequence and the reference DNA sequence as input to the insertion technique. This insertion technique was originally introduced in [10] and was also modified in [9]. Now we have further modified this technique in accordance with our algorithm.

This technique is explained as follows.

1. First divide both encrypted DNA sequence and the reference DNA sequence into segments where each segments contains a random number of DNA nucleotides so the segments are not fixed in length.

(Random length of each segments is decided by the random number generator which seeds are passed and these seeds are known secretly to both sender and receiver.)

2. Next we insert each segment of the encrypted DNA sequence before the segments of reference DNA sequence respectively finally we get a faked DNA sequence with the encrypted DNA sequence hidden.

Finally we present our algorithm and we call this as AYP- Insertion algorithm

The proposed AYP-Insertion algorithm can be summarized in the following steps and figure below is the block diagram that illustrates this algorithm in a better way.

Input: Plaintext P, Secret key string for AYP key matrix, Secret seeds, Key for Double Transposition cipher, Reference DNA sequence.

Output: Faked DNA sequence with hidden plaintext in it.

1. First make the key matrix using the method described in the AYP Algorithm using the secret key string.
2. Encrypt the plaintext using the AYP algorithm to produce the intermediate text to be further encrypted by the Phase II of the AYP-Insertion Algorithm.
3. Now apply the Double transposition cipher to further encrypt the intermediate text. After this step we finally get out cipher text.
4. Convert the cipher text obtained from above in binary form using 4 - bits coding (*By first converting the whole cipher text in ASCII and then converting it into the binary from where each of the digits is converted to 4 bit binary number. Ex. 6 is converted to 0110
1 is converted to 0001*).
5. After converting the cipher text into binary from use binary coding scheme to convert the same into a DNA sequence. For simplicity let’s call this encrypted DNA sequence be DS and length of DNA sequence be represented as [DS].

6. Now generate the Random number sequence $r_1, r_2, r_3, r_4, \dots$ using the random number generator seed R and another random number sequence $k_1, k_2, k_3, k_4, \dots$ using the random number generator seed K
7. Then find the smallest integer “t” such that

$$\sum_{i=1}^t r_i > [DS]$$

8. Now divide the DS into segments with length $r_1, r_2, r_3, r_4, \dots, r_{t-1}$ for the simplicity we denote these segment by the $DS_1, DS_2, DS_3, DS_4, \dots, DS_{t-1}$ and let the residual part be DS_t
9. Now divide the reference DNA sequence into segments of length $k_1, k_2, k_3, k_4, \dots, k_{t-1}$ and truncate the residual part of the REF. For simplicity we denote these segments by the $REF_1, REF_2, REF_3, REF_4, \dots, REF_{t-1}$
10. Insert each DS_i , where $1 \leq i \leq t - 1$ of DS before each segment of Reference DNA string i.e., before REF_i .
11. Finally put the DS_t in the end of the sequence to produce the Faked DNA sequence with the encrypted data hidden.

Let’s denote this faked DNA sequence be C.

To use this method what we require is

1. Secret Key String for AYP Key Matrix [SK].
2. Key for the Double Transposition cipher [DT].
3. Two random number seeds [R] and [K].

So the Faked DNA sequence C can be send in public channel while [SK], [DT], [R], [K], [T] need to be sent on secure channel between sender and the receiver.

The receiver should follow the following algorithm to recover the hidden data and perform the subsequent decryption:

1. Generate two number sequences $r_1, r_2, r_3, r_4, \dots$ and $k_1, k_2, k_3, k_4, \dots$
By using the same random number generator and the seeds [R] and [K] respectively.
2. Now find the largest integer “n” such that

$$\sum_{i=1}^n r_i + k_i \leq [C]$$

And divide the [C] into segments with lengths

$$r_1 + k_1, r_2 + k_2, r_3 + k_3, \dots, r_n + k_n$$

And the remaining part of C is denoted as DS_{n+1}

3. For each segment i , $1 \leq i \leq n$ of C extract the first r_i bits called DS_i and concatenate all DS_i where $i \leq j \leq n + 1$ to be DS .
4. Convert DS into binary form and group into groups of 4 bits each and convert it into the corresponding decimal digit.

After this we get a string containing digit only. Now divide this string obtained into groups containing 2 digits only. Now convert each group to its corresponding English alphabet characters.

Remark 2. Here we are assuming that plaintext consists only uppercase letters just for simplicity. We can also do this encryption decryption process if the plaintext is a mixture of both upper case and lower case with a little bit of manipulation.

5. Now apply decryption process of Double Transposition Cipher.
6. After that to Decrypt further the receiver needs to construct the AYP Key Matrix.
7. Now again group the obtain text into groups contain 2 letters each. And locate the letters of each group in the AYP Key Matrix and just follow reverse of the steps that are followed while selecting the encrypted characters for a particular plaintext character.

5 Conclusion

In this paper, we discussed new chapter in information security. i.e. DNA Based Cryptography. The storage capacity, vast parallelism of DNA are its main advantages for cryptographic purposes. We presented an Encryption algorithm named AYP Algorithm. We then used this algorithm in DNA based cryptography which we discussed through the 3 phases where in phase 1 we are using AYP algorithm and we can clearly see that number of characters in the intermediate text after Phase I is getting double. This helps us to increase the number of possible ways thereby making the brute force attack more difficult.

References

1. Adleman, L.M.: Molecular computation of solutions to combinatorial problems. *Science* **266** (5187), 1021–1024 (1994)
2. Lovgren, Stefan (2003-02-24). Computer Made from DNA and Enzymes. National Geographic. Retrieved 2009-11-26
3. Benenson, Y., Gil, B., Ben-Dor, U., Adar, R., Shapiro, E.: An autonomous molecular computer for logical control of gene expression. *Nature* **429**(6990), 423–429 (2004)
4. DNA stores poems, a photo and a speech|Science News
5. Bonnet, Jerome, Yin, Peter, Ortiz, Monica E., Subsoontorn, Pakpoom, Endy, Drew: Amplifying Genetic Logic Gates. *Science* **340**, 599–603 (2013)
6. Jie, C.: A DNA-based bio molecular cryptography design. *Proc. IEEE Int. Symp.* **3**, III–822 (2003)
7. A. Gehani, T.H. LaBean, J.H. Reif: DNA-based cryptography. *DNA Based Computers V*. Providence **54**, 233–249 (2000)

8. O. Tornea, .B.E. Monica: Security and complexity of a DNA-based cipher. In Roedunet International Conference (RoEduNet), 11th IEEE International Conference (2013)
9. Atito, A., Khalifa, A., Rida, S.Z.: DNA-based data encryption and hiding using playfair andinsertion techniques. *J. Commun. Comput. Eng.* **2**(3), 44–49 (2012)
10. Shiu, H.J., et al.: Data hiding methods based upon DNA sequences. *Inf. Sci.* **180**, 2196–2208 (2010)