



International Journal of Emerging Technologies in Computational and Applied Sciences (IJETCAS)

www.iasir.net

A New DNA Encryption Technique for Secure Data Transmission

Sanjeev Dhawan¹, Alisha Saini²

¹Faculty of Computer Science & Engineering, ²Research Scholar
Department of Computer Science & Engineering, University Institute of Engineering and Technology,
Kurukshetra University, Kurukshetra-136 119, Haryana, INDIA.
E-mail (s): rsdhawan@rediffmail.com, alisha191@rediffmail.com

Abstract: DNA Cryptography is one of the emerging techniques in today's world of science. It uses the biological element known as DNA alongwith cryptography. One of the major drawbacks of DNA Cryptography field is that a lot of research and work is required to reach a position in which it can be implemented and made useful for practical purposes. There is a need that knowledge from traditional cryptography and DNA technology should be exchanged and a cryptosystem could be devised so that benefits from both the fields can be enjoyed. The DNA technology focuses on experiments. There are very few technologies which have been developed and well accepted in the last few years. Some of these accepted key technologies are polymerase chain reaction (PCR), DNA synthesis and DNA digital coding. Various encryption algorithms are devised based on these technologies which results in excellent security of the data. The main focus of this paper is to provide with an encryption decryption algorithm using the above mentioned technologies with secure strength, bringing failure to the intruder effort to break the cipher.

Keywords: Cryptograph, Decryption, DNA, Encryption, DNA Computing, DNA Cryptography

I. Introduction

DNA cryptography is encrypting or hiding a data in terms of DNA sequences, which can be done using several DNA technologies with the biochemical methods. The vast parallelism and extraordinary information density characteristics of DNA molecules are explored for cryptographic purposes such as encryption, authentication, signature, and so on. DNA cryptography might become of practical significance in the context of labelling organic and inorganic materials with DNA 'barcodes' [1]. DNA computing has its own application in the field of huge information storage, massive parallel processing, and low energy consumption. DNA computational logic can be used in cryptography for encrypting, storing and transmitting the information, as well as for carrying out computation. DNA cryptography has shown its effectiveness in the field of secured data transmission. DNA is a medium for ultra-compact information storage. Currently it is in the development phase and requires a lot of work and research to reach a mature stage. DNA as a means of cryptography has high technical laboratory requirements and computational limitations, as well as the effort demanding extrapolation means so far. DNA-based Cryptography is an approach to guarantee highly secure environment. Various encryption schemes have been designed by using the technologies of DNA synthesis, PCR amplification and DNA digital coding including the theory of traditional cryptography. There are various algorithms for encryption of messages in DNA cryptography using DNA base: Bi-serial DNA encryption algorithm, IDEA algorithm for making it more secure, Image encryption algorithm based on DNA self-assembly technology.

As Prabhu and Adimoolam [2] claim an encryption algorithm which is based on number conversion, DNA digital coding, PCR amplification and XOR operation. In this scheme two primer pairs were used as the key which were not independently designed by the sender or receiver. The traditional encryption method and DNA Digital Coding were used to preprocess operation to get completely different cipher text from the same plaintext, which can effectively prevent attack from possible word as PCR primers. This encryption scheme provided a double layer security but a single imperfection in the design of encryption scheme could allow successful attacks. The cost of encryption was also very high with process of techniques. According to Rakheja [3] a more secure algorithm using DNA computing on international data encryption algorithm (IDEA) was designed. A layer of DNA cipher was added over the basic IDEA algorithm. The cipher now will be in form of DNA sequence which will even hide very existence of the underlying IDEA algorithm. Key space was extended a bit to make it more immune to cryptanalytic attacks.

II. Biological Background

The biological background of DNA cryptography consists of the biological elements that work as the backbone of the whole scheme. These elements are described as:

A. DNA

A Deoxyribonucleic Acid (DNA) is the genetic content of the cell found in every living organism. The main task of the DNA is to carry the genetic information from parents to their offspring. DNA is a double helix structure consisting of four bases: Adenine (A), Thymine (T), Cytosine (C), and Guanine (G). The pairing will be done in such a way that Adenine will always pair up with Thymine and Cytosine will always pair up with Guanine.

B Amino Acid

The DNA sequences after pairing are matched with the amino acid sequence. Combination of any three nucleotide of DNA results in an element of the codon table and there is a corresponding amino acid for it. Based on the building of DNA codon, the amino acid is made [5].

III. Biomolecular Computation

Recombinant DNA techniques have been developed for a wide class of operations on DNA strands. There has recently arisen a new field of research known as DNA computing, that makes use of recombinant DNA techniques for carrying out computation [6]. Recombinant DNA operations were shown to be theoretically sufficient for universal computation [7]. Biomolecular computing (BMC) methods have been proposed to solve difficult combinatorial search problems such as the Hamiltonian path problem [8], using the vast parallelism available to do the combinatorial search among a large number of possible solutions represented by DNA strands. For example [9] and [10] proposed biomolecular computing methods for breaking the Data Encryption Standard (DES). While these methods for the solution of hard combinatorial search problems might succeed for fixed sized problems, they are ultimately limited by their volume requirements, which may grow exponentially with input size. For example, DNA is appealing media for data storage due to the very huge amount of data can be stored in compact volume. It vastly exceeds the storage capacities of conventional electronic, magnetic, optical media. A gram of DNA contains about 10^{21} DNA bases, or about 108 tera-bytes. Hence, a few grams of DNA may have the probability of storing all the data stored in the whole world.

IV. Proposed Algorithm

The encryption process in this case is little different from the encryption schemes used till now. In those encryption schemes DNA technologies like PCR amplification, XOR operation etc were used. Primer pairs were added to the data which were used as keys. And after the encryption is finished when it comes to sending of data, the encrypted data is compressed. But in this proposed algorithm of DNA encryption/decryption, no primer pairs are used and no compression technique is applied. The proposed encryption algorithm firstly will generate a substitution array after entering all the five parameters. Firstly decide the FINAL_ARRAY size. It can be of any size, depending on the need of the user. From the chosen subset, select only those numbers which will give the specific remainder, remainder when divided by the modulus (this will increase the security level). Call this array of number as Substitution_Array. After this give input on which encryption is to be done and obtain ASCII value for it. Then perform ASCII code to numerical value conversion. This will convert the ASCII characters into their numerical values. The proposed scheme presents with a symmetric algorithm which will provide with secure strength, bringing failure to the intruder effort to break the cipher. In this, the main task is to grant security to the cipher text by using five parameters (discussed below) which will work as key. This will include a substitution array which will play a major role in giving the strength to the algorithm. The encryption and the decryption algorithm will be made public. These five parameters of the key will give secure strength to the algorithm. This will bring failure to the intruders' effort in breaking the algorithm. The algorithm will proceed with both the sender and the receiver knowing the value of all the five parameters of the key.

$$\text{Key} = \{\text{Final_Array_Size}, \text{Starting_Number}, \text{Ending_Number}, \text{Modulus}, \text{Remainder}\}$$

Where

- Final_Array_Size is the size of the array until which the numbers are to be generated.
- Starting_Number is the number to start within the Final_Array_Size number.
- Ending_Number is the number to end within the Final_Array_Size number.
- Modulus is the number which will extract only those numbers to be used for substitution which give the remainder.
- Remainder is the number left when the number between the Starting_Number and Ending_Number are divided by the modulus.

To know about all the five parameters correctly will be a difficult task for the intruder. So this algo is assured to be secure. This algorithm will successfully perform encryption and decryption for all the 256 ASCII characters.

V. Encryption

The encryption process in this case is little different from the encryption schemes used till now. In those encryption schemes DNA technologies like PCR amplification, XOR operation etc were used. Primer pairs were added to the data which were used as keys. And after the encryption is finished when it comes to sending of data, the encrypted data is compressed. But in this proposed algorithm of DNA encryption/decryption, no primer pairs are used and no compression technique is applied. The proposed encryption algorithm firstly will generate a substitution array after entering all the five parameters. Firstly decide the FINAL_ARRAY size. It can be of any size, depending on the need of the user. From the chosen subset, select only those numbers which will give the specific remainder, remainder when divided by the modulus (this will increase the security level). Call this array of number as Substitution_Array. After this give input on which encryption is to be done and obtain ASCII value for it. Then perform ASCII code to numerical value conversion. This will convert the ASCII characters into their respective numeric code. This conversion is very simple. We just have to add the character into the command window and press ok. It will automatically gives out the resultant respective numeric code. This conversion is very simple. We just have to add the character into the command window and press ok. It will automatically gives out the resultant respective numeric values. Substitute the numbers from the generated substitution array sequentially on a one-to-one basis at the place of character of the plain text. Now divide the number chosen with the ASCII value of the plain text character and calculate the Quotient and the Remainder and store it in the Array. The message will now be the series of Quotient followed by the Remainder. Apply Base 10- Base 4 conversion on the generated series of Quotient and Remainder. With this a string of Hyphen_Indices will be generated which will represent the indices of the hyphen that are used to separate the characters in the input after base conversion. After generating base 4, replace all the 0,1,2,3 in the series by A, T, C, and G respectively to get the DNA sequence. The DNA sequence will now be converted to amino acid sequence. For amino acids with more than one possible nucleotide codon, this function will randomly selects the codon corresponding to that particular amino acid [4]. According to the above mentioned assumptions, the proposed encryption algorithm consists of following steps:

STEP 1: Generate all the numbers in the Final_Array by entering the value for the Array size, starting and ending number, Modulus and the Remainder once. Call this array as SUBSTITUTION_ARRAY.

STEP 2: Convert each letter into its numerical value using ASCII code.

STEP 3: Substitute the generated numbers sequentially on a one-to-one basis at the place of character of the plain text.

STEP 4: Now divide the number chosen with the ASCII value of the plain text character and calculate the Quotient and the Remainder and store it in the Array.

STEP 5: The message will now be the series of Quotient followed by the Remainder.

STEP 6: Apply Base 10- Base 4 conversion on the generated series of Quotient and Remainder.

STEP 7: Replace all the 0,1,2,3 in the series by A, T, C, and G respectively.

STEP 8: The DNA sequence is converted to amino acid sequence.

Note: The Encrypted message is now ready to forward to the receiver for communication.

VI. Decryption

From the sender's side we get the encrypted data. There is no need of applying decompression algorithm to recover compressed data as used before in other encryption schemes. This is because we are getting the cipher in amino acid form in which the data is already in very small sequence. We will get a sequence of amino acid in the form of cipher from the sender. This amino sequence if firstly converted to DNA sequence. In general, the mapping from an amino acid to a nucleotide codon is not a one-to-one mapping. After we get DNA sequence, the A, T, C, G in sequence are replaced by 0, 1, 2 and 3 respectively i.e. DNA digital coding is converted into binary code. The generated series will now go through hexadecimal conversion i.e. base4 – base10 conversion after getting the Hyphen_Indices string. Now we get the series of quotient followed by the remainder. Once this series is got, a loop is run for all the elements. Each ASCII value is multiplied and the generated remainder is added to it. Now compare it with the number in the SUBSTITUTION_ARRAY. The ASCII valued character that matches with the number in the SUBSTITUTION_ARRAY will be stored in a separate array and will be read as the final message. In case anyone of the key is wrong, then there missing of data or improper form of data. So, there are chances of maintaining more secure of data. Another thing to be kept in mind is that, chances are there when any number of the SUBSTITUTION_ARRAY happens to be zero. Special care should be taken in that case. The decryption algorithm consists of following steps:

STEP 1: The amino acid sequence is converted to DNA sequence.

STEP 2: The nucleotide sequence of the DNA i.e. A, T, C and G is replaced by 0, 1, 2 and 3 respectively.

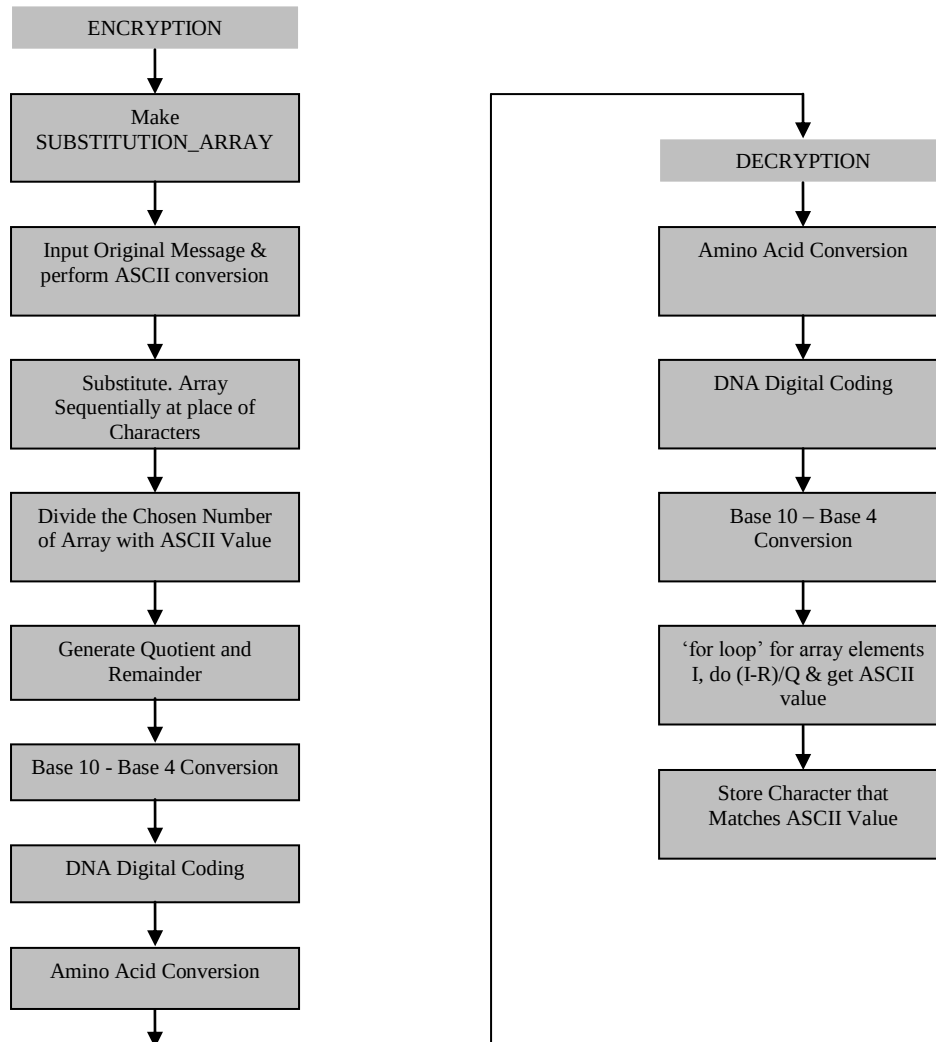
STEP 3: Base 4 – Base10 conversion is applied on the generated sequence to get the quotients followed by the remainders.

STEP 4: A loop is run by division of each array element after subtracting it from the remainder. The generated number is matched with the ASCII valued character

STEP 5: The ASCII valued character that matches with the number in the SUBSTITUTION_ARRAY will be stored in a separate array and will be read as the final message.

Note: The Decrypter would take the Amino Acid sequence.

Figure 1 process of encryption and decryption.



VII. Results

The implementation of the DNA Cryptography algorithm for encryption and decryption algorithm presented above have been carried out using PC with Intel Pentium Dual Core Processor running on 1.60 GHz. The proposed algorithms are designed by using Java as programming language and Eclipse for providing user interface. The plaintext after going through encryption and decryption process has successfully been recovered. After encrypting the plaintext and running the cipher, the original text is generated. In the first snapshot, we first entered the keys, which are passwords known only to the one who encrypt and the one who decrypt the text. In second snapshot, we have the plaintext and then the cipher (amino acid string) generated after applying encryption algorithm. The third snapshot shows the generation of plaintext from the amino acid string after applying decryption algorithm. The following assumptions are made about the substitution Array

- Let the Final_Array_Size is 2000. This array size can be varied to higher or lower values.

- *Starting_Number* = 860 and *Ending_Number* = 1200.
- *Modulus* = 25 and *Remainder* = 2.

Snapshot 1 Entering the Keys

A. Encryption

After giving the input to the encryption dialog box, the ASCII value of the corresponding character will be generated. When we had entered the array size along with the modulus, remainder and starting and ending number, the substitution array would have been generated. Now ASCII value of each character will be divided sequentially with the array element. This will give out the quotient and remainder after division as shown in table 1 below:

Table 1 Encryption Process

Character	ASCII value	Substitution Array	Division	Quotient	Remainder
C	67	877	877/67	13	6
O	79	902	902/79	11	33
M	77	927	927/77	12	3
P	80	952	952/80	11	72
U	85	977	977/85	11	42
T	84	1002	1002/84	11	78
E	69	1027	1027/69	14	61
R	82	1052	1052/82	12	68

The message now will be the series of Quotient followed by the Remainder.

$\{13,11,12,11,11,11,14,12,6,33,3,72,42,78,61,68\}$

Applying Base 10 – Base 4 conversion on the generated series.

$\{31,23,30,23,23,23,32,30,12,201,3,1020,222,1032,331,1010\}$

Hyphen_Indices string will be:

$\{2,5,8,11,14,17,20,23,26,30,32,37,41,46,50\}$

Replacing 0,1,2,3 with the DNA bases A, T, C, and G respectively.

GTGCGACGCGCGGCGATCCATGTACACCCTAGCGGTTATA

Amino acid sequence: *V1G2R1A3A3I1HV2H1PS5GYAX*

CIPHERTEXT = V1G2R1A3A3I1HV2H1PS5GYAX

Snapshot 2 Encryption

B Decryption

After getting the amino acid series from the sender as the cipher text, the decryption process will start at the decryption end. Amino acid sequence will be converted to DNA sequence

$V1G2R1A3A3I1HV2H1PS5GYAX = GTCGGACGCGCGCGATCCATGTACACCCTAGCGGTTATA$

Now the string of hyphen indices is given to the processor

$\{2,5,8,11,14,17,20,23,26,30,32,37,41,46,50\}$

DNA Digital coding is applied, which will convert DNA bases A, T, C, and G to 0, 1, 2 and 3 respectively

$\{31,23,30,23,23,23,32,30,12,201,3,1020,222,1032,331,1010\}$

Now Base 10 - Base 4 conversion is applied which will give the following sequence

$\{13,11,12,11,11,11,14,12,6,33,3,72,42,78,61,68\}$

A loop is run for all the elements generated in the Substitution_Array and each array element is subtracted from its respective remainder divided by its respective quotient. And the respective character will be generated as shown in table 2 below:

Table 2 Decryption Process

Quotient	Remainder	'for loop' for array elements I, do (I-R)/Q & get ASCII value	Character
13	6	67	C
11	33	79	O
12	3	77	M
11	72	80	P
11	42	85	U
11	78	84	T
14	61	69	E
12	68	82	R

Snapshot 3 Decryption

VIII. Conclusion and Future Work

In this paper, we designed a DNA encryption/decryption algorithm which will provide more security from the algorithms which are in existence till now. Apart from providing double layer security with complexity of biological difficult problem and cryptography computing difficulties, we can say this algorithm will provide a two fold computing difficulty as well. And due to this reason this encryption algorithm possesses high confidential strength. The presentation of the algorithm is illustrated with the help of a small example using Java and eclipse for user interface. The main disadvantage of this algorithm is that this algorithm will take its Starting_Number from 255 onwards. This is to avoid zero quotient or quotient in decimals. The cost of encryption will be high with process of techniques. The future work comprises of analyzing the performance of the algorithm designed to various cryptanalytic attacks.

IX. References

- [1] Andre Leier, Christoph Richter, Wolfgang Banzhaf, Hilmar Rauhe, "Cryptography with DNA binary strands", 0303-2647:00:\$, Elsevier Science Ireland Ltd, PII: S0303-2647(00)00083-6 – 2000.
- [2] D.Prabhu ,M.Adimoolam, "Bi-serial DNA Encryption Algorithm (BDEA)", CoRR abs/1101.2577 -2011.
- [3] Pankaj Rakheja, "Integrating DNA Computing in International Data Encryption Algorithm (IDEA)", International Journal of Computer Applications (0975 – 8887) Volume 26– No.3 -July 2011.
- [4] <http://www.mathworks.in/help/toolbox/bioinfo/ref/aa2nt.html> Convert amino acid sequence to nucleotide sequence - MATLAB".

- [5] Kang Ning. "A Pseudo DNA Cryptography Method". March 2009 .
- [6] John H. Reif. Paradigms for biomolecular computation. In C. S. Calude, J. Casti, and M. J. Dinneen, editors, *Unconventional Models of Computation*. Springer, 1998.
- [7] Thomas Head. Splicing schemes and DNA. *Lindenmayer Systems; Impact on Theoretical computer science and developmental biology*, pages 371{383, 1992.
- [8] L. M. Adleman. Molecular computation of solutions to combinatorial problems. *Science*, 266:1021{1024, November 1994.
- [9] Dan Boneh, Christopher Dunworth, and Richard J. Lipton. Breaking DES using a molecular computer. In Eric B. Baum and Richard J. Lipton, editors, *DNA Based Computers*, volume 27 of *DIMACS: Series in Discrete Mathematics and Theoretical Computer Science*. American Mathematical Society, 1996.
- [10] Sam Roweis, Erik Winfree, Richard Burgoyne, Nickolas V. Chelyapov, Myron F. Goodman, Paul W. K. Rothmund, and Leonard M. Adleman. A sticker based architecture for DNA computation. In *Proceedings of the Second Annual Meeting on DNA Based Computers*, held at Princeton University, June 10-12, 1996., volume 44 of *DIMACS: Series in Discrete Mathematics and Theoretical Computer Science*, pages 1- 30, American Mathematical Society, May 1996.