International Journal of Engineering & Technology, 7 (2.6) (2018) 141-146

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research Paper

An efficient 2-Step DNA symmetric cryptography algorithm based on dynamic data structures

Jossy P. George[1], Joseph Varghese Kureethara[2]*

[1]Christ Institute of Management, Lavasa, Pune, India.
[2]CHRIST (Deemed to be University)
**Corresponding author E-mail: frjoseph@christuniversity.in*

Abstract

The security of text has become highly demanding in today's fast growing networking world. DNA computing is one of the emerging technologies in the arena of huge data storage and parallel computation. A single gram of DNA holds 5.5 petabytes of data. This leads to the increased risk in data communication. DNA in computers is mapped to human genome. Thus, the sequence of nucleotide base constructs the foundation of uniqueness. In this paper, a new scheme acronymed as –'Cryptography on DNA Storage'-CDS is provided. It performs the DNA data encryption in just two-step by using random private key for each letter in the plaintext and parallel swapping of the resultant text in small clusters. It is discussed keeping the time and space complexity of the algorithm in concern.

Keywords: *Security; Cryptography; Bio-Encryption; Bio-computing; DNA; DNA Cryptography*

1. Introduction

The growing use of internet across the globe, has led to the increase in threat posed by the data shared across in data communication. Data communication forms the lifeline in networking. The need to secure data is apparent, making it one of the major areas of concern.

Data can be made secure by encoding it to the other form or hiding it in another larger text such as steganography or encryption using some cryptological protocol. Security becomes essential in order to prevent it getting the data in hands of those who could misuse it. Thus, white collar jobs have become popular and highly paid with the increase in the use of networks for data communication especially when the sensitive data is sent across with less cost and much speeder time than traditional physical information carrier.

Security in networks was brought in picture for accountability, impartiality, precision, and confidentiality. It can very well restrict the unauthorized entry by others and can secure the transactions. High security measures can be implemented on online financial transactions. It can protect one's anonymity or can prove one's identity.

The security in networks refers to – 'securing the information in potentially aggressive surroundings – is a critical factor in the development of information-based methods in industry, business, and administration'[7]. Cryptography is one such process in information security for the data communications.

The need of security arises from both the internal and the external network attacks. The privacy of all communications is at any place at any time with the communications remaining private and protected and controlled, on access to information by accurately identifying users. The respective system also constitutes the reasons for security need.

2. Design Rationale

From Caesar's cipher to quantum cryptography, secret communication has come a long way. This is the era of bio-cryptography. We now give some of the basic information regarding the cryptosystem we propose in this work.

2.1. Cryptography

The need of security in networks has emerged with high rise in data communication and transmission. One of the popular ways in prevention of data intrusion by third-party is achieved by cryptography. Prior to transmission, the plaintext is converted to encrypted text, known as the ciphertext. Regaining the scrambled text back to the original text is practised by the process called decryption. The involvement of key is a major concern in cryptography. It is the key which determines the level of data security achieved. Key can be either symmetric or asymmetric in nature.

2.2. Deoxyribonucleic Acid

Deoxyribonucleic Acid is popularly acronymed as 'DNA'. It is a double-helix molecular structure which stores genetic information. Thus, it forms essential part of inheritance characteristics which are carried forward to further generations. It is basically comprised of three things known as nucleotides.
The four bases in DNA [5] are A (Adenine), C (Cytosine), T (Thymine) and G (Guanine). They carry genetic information. Fig 1 shows expanded diagram of double stranded DNA structure.

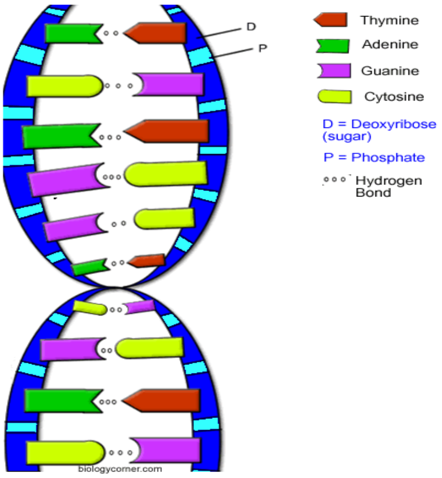

Fig. 1: DNA structure [5]

DNA is also known as "the blueprint of life". The reason behind this is that it comprises the code, or commands for structuring the organism and confirming how the organism functions correctly [5]. The translation of information into proteins is done by ribonucleic acid (RNA). Fig 2 shows how message is transmitted to protein.

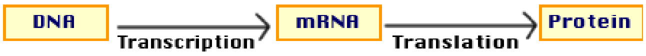

Fig. 2: Message translation

Information can be transferred only from DNA to a protein. It never can be reversed [3]. It is the storing of base information in protein that helps the successful transmission of information. The nucleotide base sequence with A, C, T and G are huge with redundant sub-pattern within itself. There are several nucleotide bases in cells, which define functionality for the organism to operate.

Only the first word in a title must be capital and other word should be in small case. Author details must not show any professional title (e.g. Managing Director), any academic title (e.g. Dr.) or any membership of any professional organization (e.g. Senior Member IEEE).

2.3. DNA computing

DNA computing is an inspiration drawn from biomedical stream. The structure exactly maps to the human genome. Thus, it can be either single stranded or double stranded. It not only has huge amount of information storage but also it provides parallel processing for the same. Due to both capabilities it is expected to a become super-computer in near future.

3. Literature Survey

In this section, different DNA cryptographic techniques and their salient features are explained.

Sadeg [10] proposed an encryption algorithm based on DNA structure. It includes methods that generate the ideas from the methods of transcription (transfer from DNA to mRNA) and translation (from mRNA into amino acids). Cui *et al*.[1] and Gahlaut *et al*.[5] propose encryption schemes using DNA technology. This method is created by using the technologies of DNA synthesis, PCR amplification, DNA digital coding as well as the theory of traditional cryptography. By applying the special function of primers to PCR amplification, the primers and coding mode are used as the keys of the scheme. The traditional encryption method and DNA digital coding are used to pre-process the plaintext, which can effectively prevent attack from a possible word as PCR primers.

Wang *et al*. [12] provided cryptography scheme for DNA encryption which would use RSA which comes with asymmetric key and shall be connected to DNA computing technique.

Pramanik *et al*. [8] provided one-time pad key for the encryption of ssDNA data. If the first bit is 1 in plaintext, it would compute long random 10-mer oligonucleotide from the sequence end and takes its Watson-Crick complement. Then it attaches sequential random integer, attaches it to the DNA ciphertext and transmits to receiver using open channel communication.

Yunpeng *et al*. [14] provided an Index-Based Symmetric DNA Encryption Algorithm which adopts the method of block cipher and index of string. This cryptosystem encrypts the DNA sequence based plaintext.

Kalsi *et al*. discuss about DNA Cryptography and Deep Learning using Genetic Algorithm with NW algorithm for Key Generation [4].

Wang *et al*. [13] suggest an information hiding on DNA steganography where the sender uses DNA steganography to send one part. If the microdot is assumed or contaminated, the message will be encrypted and decomposed repeatedly until the microdot is not assumed or contaminated. If the microdot is not assumed or contaminated, the corresponding part will be publicly sent.

Tornea *et al*. [11] have provided work on security and complexity of a DNA-Based cipher, it uses key space, cryptanalysis and statistical measurements for evaluating security. However, it measures complexity by theoretical and practical measurements of time.

Martin *et al*., [6] provided an undecryptable symmetric encryption, that allows a device to symmetrically encrypt a device without being able to decrypt it and nor any attacker that composes it.

Roy *et al*. [9] proposed an improved symmetric key cryptography with DNA-based strong cipher, which has a unique ciphertext generation procedure as well as a new key generation procedure.

The need of security arises from both the internal and external network attacks. The privacy of all communications, at any place at any time with the communications remaining private and protected and control on access to information by accurately identifying users and the respective system also constitute the reasons for security need.

4. Basic Definitions and Equations

4.1. Symmetric-key cryptography

Symmetric-key cryptography refers to encryption methods in which both the sender and receiver share the same key.

4.2. RNA

Ribonucleic acid (RNA) transmits genetic infor-mation from DNA to proteins produced by the cell [5].

4.3. Protein

Proteins are large, complex molecules that play many critical roles in the body. They do most of the work in cells and are required for the structure, function, and regulation of the body’s tissues and organs [5].

4.4. Transcription

Transcription is the name given to the process where the information in a gene in a DNA strand is transferred to an RNA molecule [5].

4.5. Translation

Translation is the final step on the way from DNA to protein. It is the synthesis of proteins directed by a mRNA template. The information contained in the nucleotide sequence of the mRNA is read as three letter words (triplets), called codons [5].

4.5. DNA Cryptography

It is a cryptographic technique in which each letter of the alphabet is converted into a different combination of the four bases that make up the human deoxyribonucleic acid (DNA). A piece of DNA spelling out the message to be encrypted is then synthesized, and the strand is slipped into a normal fragment of human DNA of similar length. The end result is dried out on paper and cut into small dots. As only one DNA strand in about 30 billion will contain the message, the detection of even the existence of the encrypted message is most unlikely [3].

We now give the basic equations.

4.6. Key Computation

Symmetric key = Random Number + Letter Frequency …………. (1)

4.7. Encryption Formula

Ciphertext Character = Plaintext Character XOR Key XOR Limit Indicator …………… (2)

4.8. Key Encryption

Encrypted key = 2’s complement {FF- HEX(DEC(random number)) } ………….. (3)

4.9. Key Decryption

Decrypted key = DEC{FF - HEX(1’s complement (Encrypted key in binary - 1))} …… (4)

4.9. Decryption Formula

Plaintext Character = Ciphertext Character XAND Key XAND Limit Indicator ………….. (5)

5. Proposed System

5.1. Encryption Algorithm

Using data structure, the below given encryption algorithm processes plaintext to ciphertext. The proposed system shall consider each character in plain text, store its frequency and limit indicator (initially set to 1) in the linked list. When frequency count exceeds 255, the limit indicator is incremented by 1 setting frequency to 1. This is done so to prevent generation of invalid cipher letter.

Input: Plaintext
Data Structure: Dynamic Linked-List
Output: Ciphertext

Step 1: Generate random number between 1-5. The number 5 has been taken as high limit due to five types of characters in plain text : vowels, consonants, digits, special symbols and punctuation.

Step 2: For encryption of plaintext generate the private key. Key in our algorithm shall be different for each letter. As, combination of random number with current letter frequency:-

a. Current distinct letter frequency can be computed by maintaining its count in dynamic linked list.

b. Compute private key using eq. (1)

The letter, its frequency and limit indicator values as a structure shall be stored in a node of the linked-list. Where, Limit indicator shall be set to 1 initially.

Step 3: Encrypt the plaintext with key by performing XOR operation between them. On XOR operation between them the ASCII limit cannot be maintained for worst case. Thus, the resultant is to XORed with Limit indicator so it would restore the ASCII limit also. For each and every letter in plain text compute ciphertext character using eq. (2).
Considering the worst case where a single letter is stored recursively in DNA. The ASCII limit shall also be affected by frequency of letter due to huge data size. Thus, in this step, the ASCII limit is kept in check for such worst cases (whenever, frequency count >255) by incrementing the Limit Indicator count by 1 and setting frequency limit as 1 for that character.
Step 4: IF the index of string is more than 1 THEN
Step 5: Sequentially combine n (above Generated Random Number) letters into one forming a word.
Step 6: Perform swapping of letters in each word where, word size is more than 1, when

```
    IF random number is EVEN,
    Swapping of second character with fourth character,
    ELSE, (if random number is ODD)
    Swapping of first character with third character
    ENDIF.
    ENDIF.
```

Step 7: Convert each word into binary and store information in nucleotide base of ddDNA with
A=>00
C=>01
T=>10
G=>11 and form the sequence.
Step 8: Repeat the steps 2 and 3 for entire text.
The below table 1 provides the summary of the 2-Step encryption.
Step 9: Perform key encryption using eq. (3). If the value is not 3-digit pad 0 in front of it.
Step 10: Send this key along with text. In following format
| Encrypted Key in dna base sequence | Cipher Text sequence|

5.2. Decryption Algorithm

Using data structure, the below given encryption algorithm processes. Using data structure, the below given decryption algorithm processes ciphertext to plaintext. The proposed system shall consider each character in cipher text, store its frequency and limit indicator (initially set to 1) in the linked list. When frequency count exceeds 255, the limit indicator is incremented by 1 setting frequency to 1. This is done so to prevent generation of invalid plain letter.

Input: Ciphertext
Data Structure: Dynamic Linked-List
Output: Plaintext

Step 1: Receive the cipher text from strand
Step 2: Convert consecutive four bases i.eA,C,T ,G into binary and from binary convert it to ASCII equivalent.
Step 3: Extract key for initial character from cipher text of 3-digit.
Step 4: Key Decryption using eq. (4)
Step 5: Repeat step 2 & compute key for respective letter using (1)

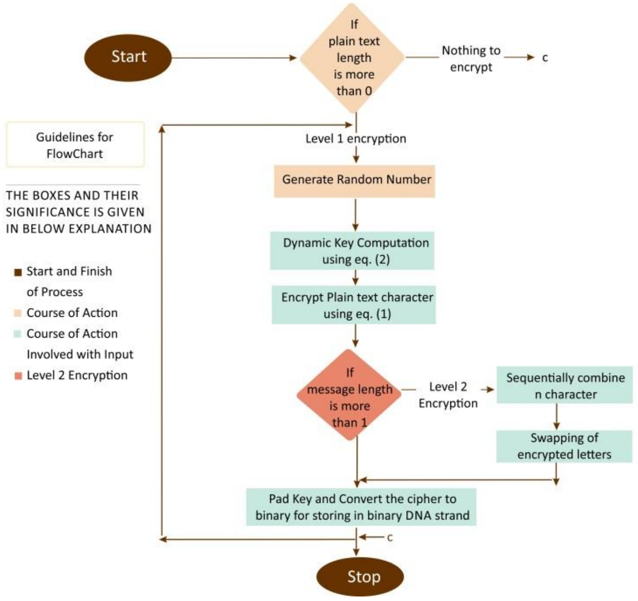
Start
If plain text length is more than 0
Nothing to encrypt
c
Guidelines for FlowChart
THE BOXES AND THEIR SIGNIFICANCE IS GIVEN IN BELOW EXPLANATION
■ Start and Finish of Process
■ Course of Action
■ Course of Action Involved with Input
■ Level 2 Encryption
Level 1 encryption
Generate Random Number
Dynamic Key Computation using eq. (2)
Encrypt Plain text character using eq. (1)
If message length is more than 1
Level 2 Encryption
Sequentially combine n character
Swapping of encrypted letters
Pad Key and Convert the cipher to binary for storing in binary DNA strand
c
Stop

Fig. 3: CDS Encryption Flow

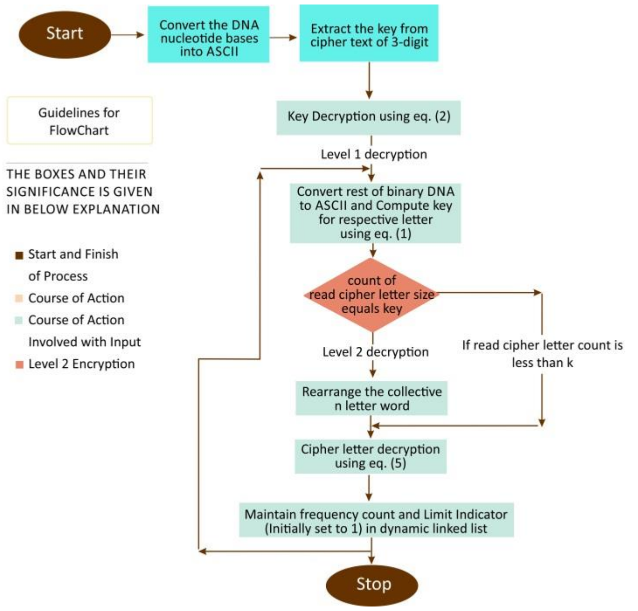
Start
Convert the DNA nucleotide bases into ASCII
Extract the key from cipher text of 3-digit
Guidelines for FlowChart
THE BOXES AND THEIR SIGNIFICANCE IS GIVEN IN BELOW EXPLANATION
■ Start and Finish of Process
■ Course of Action
■ Course of Action Involved with Input
■ Level 2 Encryption
Key Decryption using eq. (2)
Level 1 decryption
Convert rest of binary DNA to ASCII and Compute key for respective letter using eq. (1)
count of read cipher letter size equals key
Level 2 decryption
If read cipher letter count is less than k
Rearrange the collective n letter word
Cipher letter decryption using eq. (5)
Maintain frequency count and Limit Indicator (Initially set to 1) in dynamic linked list
Stop

Fig. 3: CDS Decryption Flow

Step 6: IF count of read cipher letter size equals n i.e key THEN

Step 7: For each word of size n from the binary strand, rearrange the letter in each word based on n being

EVEN/ODD.

ENDIF.

Step 8: Perform XAND operation between letter and decrypted key. Thus, the resultant is to be XANDed with Limit Indicator so, it would restore the original ASCII in bulky dna data; using eq. (5).

Step 9: For each distinct letter maintain an entry along with its frequency count and Limit Indicator (initially set to 1) in dynamic linked list.

Step 10: Repeat the steps 5,6,7,8 and 9 for entire cipher.

6. Flow of the Cryptosystem

6.1. Encryption Flow

The process involved of converting a plaintext into ciphertext in our algorithm is explained. The encryption flow for CDS explaining our 2-Step encryption is explained step by step below by figure 3.

6.2. Decryption Flow

The conversion of ciphertext to plaintext using our algorithm is explained. The decryption flow for CDS explaining our 2-Step decryption is explained step by step below by figure 4.

7. Algorithm Execution

7.1. Encryption Flow

The process involved of converting a plaintext into ciphertext in our algorithm is explained. The encryption flow for CDS explaining our 2-Step encryption is explained step by step below in table 2. via. example.

Input: Plain Text => Hello Hi

Data Structure: Dynamic Linked-List

Output: Cipher Text

Table 1: Encryption Flow

Data Structure Manipulation	Take Linked List, Node 1: (values) Plain Letter: H Current Frequency: 1 Limit Indicator: 1 Node 2: (values) Plain Letter: e Current Frequency: 1 Limit Indicator: 1 Node 3: (values) Plain Letter: l Current Frequency: 1 Limit Indicator: 1 Node 4: (values) Plain Letter: o Current Frequency: 1 Limit Indicator: 1
Step 1:	Random Number generated equals say, 3
Step 2,3:	say ASCII of H is 71 [XOR] Private Key = 72, i.e Key = { Random Number[1-5] + letter frequency } thus, key differs for each character 1111 1111 1111 [Plain Letter] 0001 0010 0100 [XOR] [Key] --------------- --------------- ------------- 1110 1101 1011 [Cipher Letter] Get ASCII 55 44 33 [Cipher ASCII]
Step 4,5:	Hello Hi=>XyzzwXg Word(s): |Xyz|zw |Xg|
Step 6:	Since random number is ODD Rearranged Word(s): |zyX| wz|Xg|
Step 7,8:	Convert each letter into 8-bit binary say, z=>ASCII=>10001000 then store it in nucleotide base, the sequence becomes TATA
Step 9:	Perform key encryption= 2's Complement {FF-HEX(3)} say, on computation for above example it gives encrypted key = 109
Step 10:	Store the key also in A,C,T,G so say 109 => ATCCGAGGGTCGGTG Final genome sequence, (key+TATA) ATCCGAGGGTCGGTATA

7.2. Decryption Flow

The conversion of ciphertext to plaintext using our algorithm is explained. The decryption flow for CDS explaining our 2-Step decryption is explained step by step below in table 2. via. example.

Input: Cipher Text

Data Structure: Dynamic Linked-List

Output: Plain Text => Hello Hi

Table 2: Decryption Flow

Step 1:	Receive the cipher text from strand ATCCGAGGGTCGGTATA
Step 2:	ConvertA,C,T ,G into binary and from binary to ASCII equivalent. A=>00, C=>01, T=>10, G=>11 00100101110011111110011111101110001000
Step 3:	Extract key for initial character from cipher text of 3-digit. i.e 1 digit 4 bases so 3 digit = initial 12 bases say for above example it gives ATCG AGGG TCGG 1 0 0
Step 4:	Key Decryption using eq. (4) DEC{FF - HEX(1's complement (Encrypted key in binary - 1))} say 100 in binary gives 11111000 - 1=> 111110111=>1's complement =>00001000=>FC FF-FC=03=>into DEC=>3
Step 8:	Perform XAND operation between letter and decrypted key. Thus, the resultant is to be XANDed with Limit Indicator so, it would restore the original ASCII in bulky dna data; using eq. (5). say above TATA gives z XAND 4 XAND => l (lower-case of L)
Step 9:	For each distinct letter maintain an entry along with its frequency count and Limit Indicator (initially set to 1) in dynamic linked list.
Data Structure Manipulation	Take Linked List, Node 1: (values) Plain Letter: l Current Frequency: 1 Limit Indicator: 1 Node 2: (values) Plain Letter: e Current Frequency: 1 Limit Indicator: 1 Node 3: (values) Plain Letter: H Current Frequency: 1 Limit Indicator: 1
Step 9:	Redo steps 5,6,7,8 and 9.

8. Algorithm Analysis

8.1. Time Complexity

For best case, the algorithm shall take O(log n) time complexity for encryption of plaintext. For average case, the algorithm shall take O(n) time complexity for encryption of plaintext. For worst case, the algorithm shall take O(n^n) time complexity for encryption of plaintext. When only one character is stored multiple times almost covering entire length. The complexity decreases as massive value of limit Indicator in linked list and in turn successive key computation.

8.2. Space Complexity

No extra characters are padded to make cipher text lengthier than plain text. The space occupied is minimal or rather trivial in comparison to size of DNA.

8.3. Performance Analysis

The performance of the algorithm increases with increase in number of characters in plaintext. The below graph 3, shows performance of algorithm measured with time against space/length of algorithm.

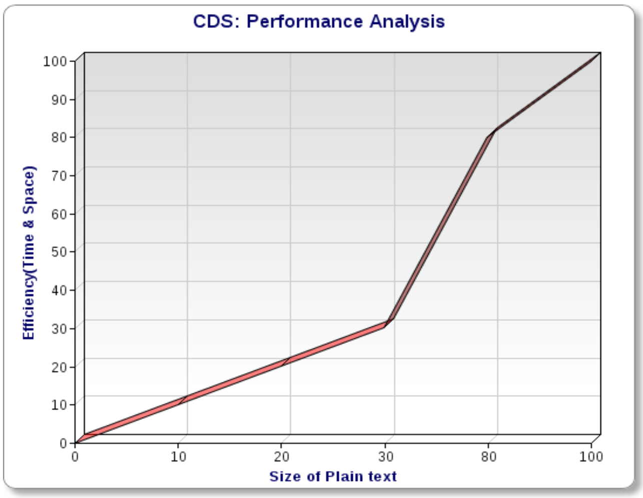

Fig. 4: CDS Performance Analysis

8.4. Dynamism of cipher

Even if the same plaintext is supplied to the algorithm, due to random number generation the cipher shall differ.

8.5. Pattern Analysis of DNA sequence

From the pattern analysis of DNA sequence it is difficult to guess the plaintext.

8.6. Key Strength

The sensitivity of the first letter in the plaintext is overcome by rearranging the cipher letter. In this sense, for first letter frequency shall be 1 and random number from 1-5 can be guessed. In special case, where plaintext length = 1, due to key padding. It is difficult to crack the cipher. Key generated initially for first character is small yet powerful.

9. Conclusion

DNA cryptography is a relatively new and very promising direction in cryptography research. It is currently an expensive technology. With lot of advancements in the future, it is hoped to be a highly cost effective and efficient cryptosystem. With its progressive growth, the IT market shall embrace DNA computing in near future for complex robotic solutions. It is often compared with quantum computing considering its scope and progress.

Although in its initial stage, DNA cryptography is very effective. Nevertheless, the use of the DNA as a means of cryptography has high implementation requirements and computational limitations, as well as the labor intensive extrapolation till date.

References

[1] Cui, G., Qin, L., Wang, Y. and Zhang, X. (2008, September). An encryption scheme using DNA technology. In *Bio-Inspired Computing: Theories and Applications, 2008. BICTA 2008. 3rd International Conference on* (pp. 37-42). IEEE.

[2] Gahlaut, A., Bharti, A., Dogra, Y. and Singh, P. (2017, May). DNA Based Cryptography. In *International Conference on Information, Communication and Computing Technology* (pp. 205-215). Springer, Singapore.

[3] Haque, R. and Saha, R. (2017). A novel Rolling based DNA Cryptography. *Journal of Bioinformatics and Genomics*, (1 (3)).

[4] Kalsi, S., Kaur, H., & Chang, V. (2018). DNA Cryptography and Deep Learning using Genetic Algorithm with NW algorithm for Key Generation. *Journal of medical systems*, *42*(1), 17.

[5] Kress, W. J. and Erickson, D. L. (2008). DNA barcodes: genes, genomics, and bioinformatics. *Proceedings of the National Academy of Sciences*, *105*(8), 2761-2762.

[6] Martin, T. (2011, February). Undecryptable symmetric encryption. In *GCC Conference and Exhibition (GCC), 2011 IEEE* (pp. 225-228). IEEE.

[7] Menezes, A. J., Van Oorschot, P. C. and Vanstone, S. A. (1996). *Handbook of applied cryptography*. CRC press.

[8] Pramanik, S., and Setua, S. K. (2012, December). DNA cryptography. In *Electrical & Computer Engineering (ICECE), 2012 7th International Conference on* (pp. 551-554). IEEE.

[9] Roy, B., Rakshit, G., Singha, P., Majumder, A. and Datta, D. (2011, February). An improved Symmetric key cryptography with DNA Based strong cipher. In *Devices and Communications (ICDeCom), 2011 International Conference on* (pp. 1-5). IEEE.

[10] Sadeg, S., Gougache, M., Mansouri, N. and Drias, H. (2010, October). An encryption algorithm inspired from DNA. In *Machine and Web Intelligence (ICMWI), 2010 International Conference on* (pp. 344-349). IEEE.

[11] Tornea, O. and Borda, M. E. (2013, January). Security and complexity of a DNA-based cipher. In *Roedunet International Conference (RoEduNet), 2013 11th* (pp. 1-5). IEEE.

[12] Wang, X. and Zhang, Q. (2009, October). DNA computing-based cryptography. In *Bio-Inspired Computing, 2009. BIC-TA'09. Fourth International Conference on* (pp. 1-3). IEEE.

[13] Wang, Z., Zhao, X., Wang, H. and Cui, G. (2013, May). Information hiding based on DNA steganography. In *Software Engineering and Service Science (ICSESS), 2013 4th IEEE International Conference on* (pp. 946-949). IEEE.

[14] Yunpeng, Z., Yu, Z., Zhong, W. and Sinnott, R. O. (2011, October). Index-based symmetric DNA encryption algorithm. In *Image and Signal Processing (CISP), 2011 4th International Congress on* (Vol. 5, pp. 2290-2294). IEEE.8