DECODING THE FUTURE USING A NOVEL DNA-BASED CRYPTOSYSTEM

Animesh Kairi[1*], Tapas Bhadra[2]

Abstract:
DNA cryptography is a particularly promising area of research in this field. The advantage of DNA cryptography is its potential to store vast amounts of data in a tiny volume. This high data density makes it well-suited for applications like secure archival storage, particularly for sensitive or classified information. Moreover, DNA cryptography's parallel processing capabilities, driven by DNA's multitasking nature can significantly accelerate encryption and decryption processes. The idea of utilizing DNA cryptography to encrypt and decode data is proposed in this article. The proposed approach performs well in computing, storage, and transmission and is especially resistant to different assaults, according to theoretical research and real world implementations. This offers a ground-breaking DNA cryptography technique based on the decimal bond algorithm. In this article, we develop the fundamental concepts underlying DNA cryptography, including the encoding of binary information into DNA sequences, encryption algorithms, and decoding procedures. We discuss various encoding strategies, their strengths, and their limitations, highlighting the trade-offs between data security and efficiency. Furthermore, we examine the challenges associated with DNA cryptography, such as the need for specialized laboratory equipment and the ongoing advancements required in biotechnology. This study also proposes a novel key generation scheme and an innovative cipher text generation approach. This article concludes with a comparison, demonstrating the effectiveness of the proposed approach against existing DNA cryptography methods.

Keywords: DNA Computing, DNA Cryptography, Decimal Bond Cryptography, Encoding, Decoding.

[1*]Institute of Engineering & Management, Kolkata, India, Email: ani.kairi@gmail.com
[2]Aliah University, Kolkata, India, Email: tapas.bhadra@aliah.ac.in

***Corresponding Author:** Animesh Kairi
***** Institute of Engineering & Management, Kolkata, India Email: ani.kairi@gmail.com

DOI: - 10.48047/ecb/2023.12.si10.00413

1. Introduction

The study of DNA cryptography is indeed a potential area for future research. DNA molecules offer an attractive option for safe data storage and transmission since, as was already noted, they are incredibly stable and have a huge capacity for information. The method is more secure because it is almost hard to decode the encoded data without knowing the precise coding scheme employed due to the complexity of DNA sequences [11]. The idea put forward in the paper is an illustration of using DNA in the process of decryption and encryption. The suggested method's distinctive cipher text generation and novel key generation processes are critical components since they considerably increase the encryption and decryption processes' security. DNA, as the carrier of genetic information, is an ideal medium for cryptography due to its stability and durability, which allows for long-term data storage. Encoding information into DNA sequences involves mapping binary data onto the four nucleotide bases (adenine, thymine, cytosine, and guanine) in a manner that ensures both data security and efficient decoding. Various encoding schemes and algorithms have been developed to facilitate this process, ensuring the integrity and confidentiality of the encoded data. Through a comprehensive review of existing research and practical implementations, we aim to provide insights into the current state of DNA cryptography and its prospects. We evaluate the limitations and ethical considerations associated with this technology and offer suggestions for further research and development. It should be highlighted, though, that DNA cryptography is still in its infancy and that there are issues that still need to be solved, such as the high cost of DNA synthesis and the requirement for specialized tools and knowledge to decode DNA sequences. Additionally, the viability of employing DNA cryptography for extensive data transmission and storage is also up for debate. However, DNA cryptography is a promising area of study and has the potential to make a significant contribution to the discipline of data and computer security. In addition to its storage and processing capabilities, DNA cryptography holds promise for secure communication and data transfer. DNA-based encryption methods can be used in various fields, including healthcare, finance, and national security, to protect sensitive information during transmission and storage. Furthermore, the biological nature of DNA ensures its resistance against conventional cyber-attacks, making it a potential solution for cyber security challenges in the digital age. This high data density makes it well-suited for applications like secure archival storage, particularly for sensitive or classified information. Moreover, DNA cryptography capability for parallel processing, driven by DNA's ability to perform multiple operations simultaneously, can significantly accelerate encryption and decryption processes.

The paper's primary contributions can be summed up as follows:

I. This study uses the Decimal Bond Cryptography methodology to provide a novel cryptography method.

II. To strengthen data security, a key generator generates both private and public keys for both the transmitter and recipient. To ensure that both parties are safely exchanging parameters, this is done.

III. This research presents experimental findings that demonstrate the suggested strategy is more effective than the current scheme.

The rest of the papers are organized as follows: Section 2 covers related works. Section 3 presents the Proposed Decimal-Bond DNA (DBD) Algorithm. Section 4 describes the experimental study of the suggested approach. Section 5 discusses security analysis and Section 6 concludes the paper.

2. Related Works

Although cryptography using DNA is a young field of study, there have been several important studies and uses in it. DNA-based authentication has been investigated by researchers in a variety of scenarios, including secured access to physical locations or online accounts [1]. Steganography, or the practice of concealing an expression within another communication or medium, is known as DNA steganography. Researchers have looked into techniques for embedding hidden messages into DNA sequences without changing the underlying genetic data, which is known as DNA steganography. Using a specifically created coding system, a 2019 study that appeared in the journal Scientific Reports disclosed a technique for encoding digital images into sequences of DNA [2]. DNA has been suggested as a potential answer to the problem of securely and long-term storing enormous volumes of digital data. This method involves encoding digital data into sequences of DNA, which are subsequently synthesized and kept in a secure location. In a 2017 study that was released in the journal Nature [3], an experimental device for DNA data storage that was able to achieve a data concentration equal to 215 petabytes per gram of DNA was exhibited. Researchers have investigated the application of natural cryptography as a means of ensuring the security of interparty communications. In this technology, messages are encoded into DNA sequences using a specific coding scheme and then transferred via methods like DNA microarrays or nanopore sequencing. Using DNA strands as "keys" to open encrypted communications, an architecture for DNA-based decryption and encryption was detailed in a 2021 study that appeared in the journal Nano Letters [4]. A novel technique for safe data storing using DNA molecules was proposed in this paper. The authors created a binary XOR-based DNA coding scheme that makes it possible to securely and effectively store data in sequences of DNA [5]. Overall, cryptography using DNA is a potential area with a wide range of possible uses in data storage, security, and privacy. The high expense and difficulty of DNA synthesis and DNA sequencing, as well as any potential ethical ramifications of using DNA for data storage and transmission, are just a few of the many issues that still need to be resolved.

Table 1 DNA Cryptography Algorithm and Techniques

Publication Year	Algorithm Used	Technique Used
2015	DNA encryption	PCR Amplification
2015	DNA encryption	DNA Steganography
2015	XOR, DNA Hybridization, Matrix computation	DNA Digital Coding
2015	Traditional encryption, DNA ECC	DNA Digital Coding
2015	DNA based Cryptography	DNA Digital Coding
2016	The DNA-based, MD5 algorithm	DNA Digital Coding
2016	XOR, OTP, DNA complementary rule	OTP based
2016	DNA encryption	DNA Digital Coding
2016	DNA encryption	DNA Digital Coding
2016	DNA encryption	DNA Digital Coding
2016	DNA-based Vigenere	DNA Digital Coding
2016	DNA based Cryptography	Pseudo DNA
2017	The traditional algorithm, DNA based	DNA Digital Coding
2017	DNA based Cryptography	Hamming code and a block cipher mechanism
2017	DNA based cryptography	DNA Digital Coding
2018	DNA based Cryptography	Binary DNA
2018	DNA-Based ECC for IoT Devices	DNA Elliptic curve cryptography
2019	Artificial DNA sequences based on Gaussian kernel function (GKF)	ECC, and Gaussian kernel function (GKF) cryptography
2020	DNA based cryptography	Data Security Using Flower Pollination Algorithm
2021	The machine is used for DNA coding	The mealy machine is used for DNA coding
2022	DNA based cryptography	Genetic algorithm

Table 1 presents the overall statistics of various DNA cryptography methods, aiding in the development of the proposed algorithm.

3. The Proposed Decimal-Bond DNA (DBD) Algorithm

The proposed encryption technique consists of four distinct stages – Mapping of Character with DNA encoding library, DNA to Binary Conversion, Bond table mapping and Encryption.

3.1 Encryption Algorithm

Step 1: Start and Take the Message from User
Step 2: Mapping of Character with DNA bits
Step 3: Addition of Padding Table
Step 4: Conversion into Binary bits
Step 5: Conversion Binary bits into Decimal bits
Step 6: End

Now, let's take an example to further understand the encryption method. It is as follows:

Step I: Consider the message M = ‘sunny (5 characters).

Step II: As a result of the mapping of the keystrokes "A," "G", "C" and "T" into a DNA encoding library, each keyboard character now has an individual combination based on these keywords.

For instance s’->’ TCCG, ‘u’->TCAG, ‘n’->’ TCCA’, and ‘y’->’ CGAT’.

A total of 4^4=256 letters can be imported in total. Given that 4 keywords can be used repeatedly, Therefore, the message in the example becomes M' = 'TCCGTCAGTCCACGAT'. (20 letters)

Step III: Now, to further conceal the message, we apply the padded library. To do this, use the padding table provided below (see Table 2).

Table 2 Padded Value Table Mapping with DNA Value

DNA VALUE	PADDED VALUE
A	AGCT
G	GCTA
C	CTAG
T	TAGS

Table 3 Binary Value Table Mapping with DNA Value

DNA VALUE	BINARY VALUE
A	00
G	01
C	10
T	11

Step IV: The following results from converting the above data into binary sequences using Table 3:

M'''='0110110010110001101100011100011001100000110111011000111000110000110111011000110110001110001101110110001101100011100011000110000110110110110010110001' (160 Characters)

The letters "A" and "C" create bonds with the letters "T" and "G," respectively, much as they would in a real DNA strand. As a result, this message will once again be padded with two of the keywords (all of which might be given to the host as a key). The message M will therefore be padded using these bonds, as shown in Table 4.

Table 4 Bond Table

BONDING COPULE	CODE
A-T	0
C-G	1

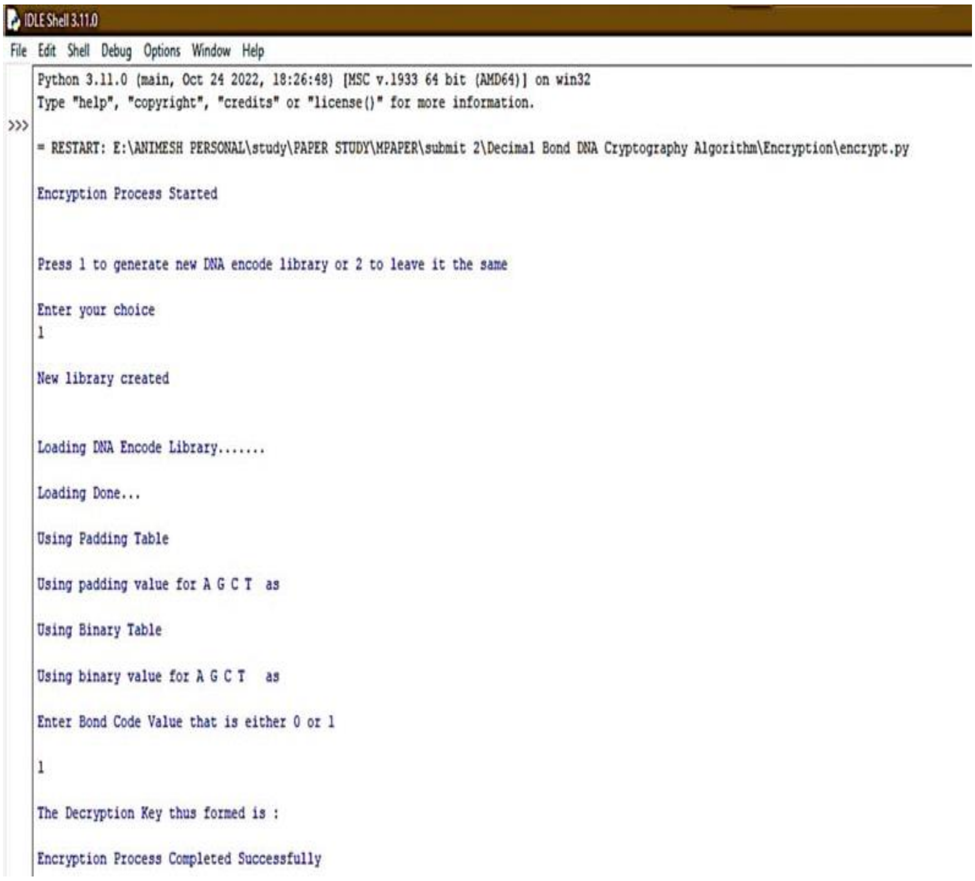
Step V: The first member of the "A-T" or "C-G" bonds, depending on which one is used, is padded at the start and end of the message, while the second member is added at the end.

For example, if bond zero, the 'A-T' bond, is chosen, the message becomes: M”” ='AM'''T' is:
M’’’’=‘00011011001011000110110001110001100110110000011011101100011100011000011011101100011011000111000110000110111011000110110001110001101100011000001101101101100101100011 1’

Total of 164 characters. The final encrypted message will be created by translating this binary message into its decimal version, two bits at a time.

“M(F)”='0123023012301301212300123230130120123130123012012312302301 3'

The last encoded message is this M(F). Figure 1 shows the output of the Encryption Code.



```
Python 3.11.0 (main, Oct 24 2022, 18:26:48) [MSC v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: E:\ANIMESH PERSONAL\study\PAPER STUDY\MPAPER\submit 2\Decimal Bond DNA Cryptography Algorithm\Encryption\encrypt.py

Encryption Process Started


Press 1 to generate new DNA encode library or 2 to leave it the same

Enter your choice
1

New library created


Loading DNA Encode Library.......

Loading Done...

Using Padding Table

Using padding value for A G C T  as

Using Binary Table

Using binary value for A G C T   as

Enter Bond Code Value that is either 0 or 1

1

The Decryption Key thus formed is :

Encryption Process Completed Successfully
```

Figure 1 Encryption Coding Output

3.2 Decryption Algorithm

The information will be transmitted together with the decryption key.

Step 1: Start
Step 2: Decryption key generation
Step 3: Bond code generation method
Step 4: Decimal to binary bit conversion
Step 5: Additional padding Removed
Step 6: Encoding library used for decryption
Step 7: Original message is retrieved
Step 8: End

Now, take an example of it for further understanding of the decryption method is consist of following steps:

Step I: Decryption key and Bond Code generation method Padded value A - Binary Value A; Padded value G - Binary Value G; Padded value C - Binary Value C; Padded value C - Binary Value C; Padded value T - Binary Value T

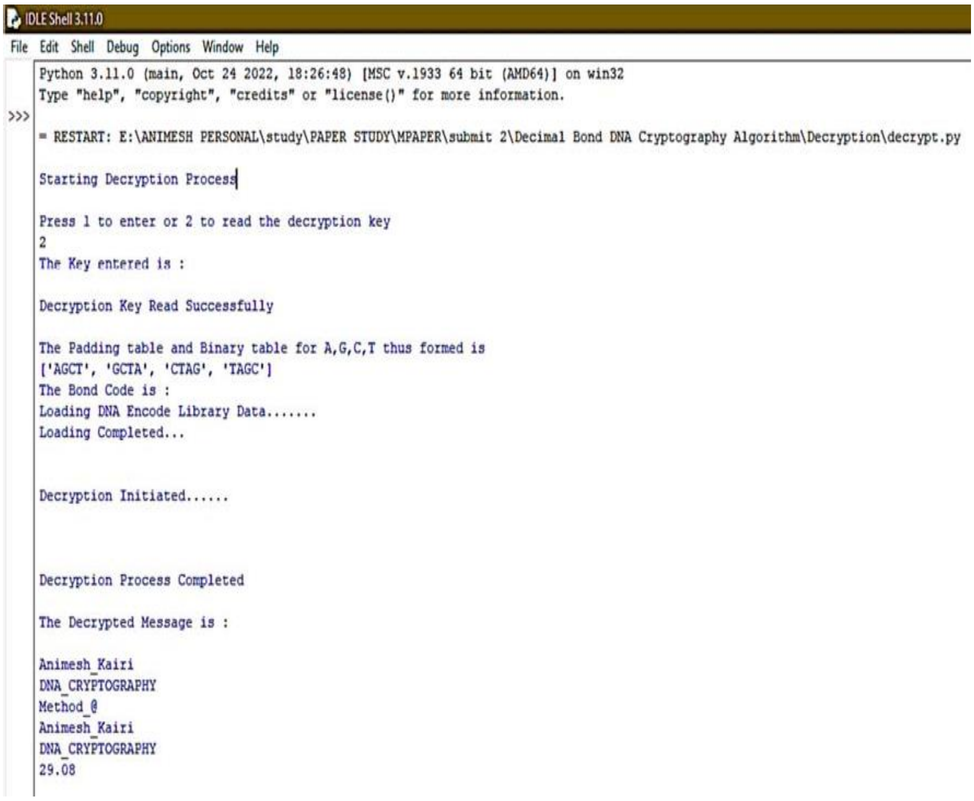
The private key is “0AGCT00GCTA01CTAG 10TAGC11” in this manner.

Step II: Each numeric in the message that is encrypted is converted to its binary equivalent as the first stage in the decryption process, producing the result M"".

Step III: The bonding pad will be eliminated by the coding value specified in the key utilized for decryption (M"').

Step IV: The encryption key will be used to decipher the message M" using a binary table code. Utilizing the key's pad information, the additional padding will be removed to create M'. Last but not least, the original message will be transmitted using the four-character encode library.

Step V: The key that's used for deciphering contains all additional information, thus the receiver only needs to keep the encoding library.



```
Python 3.11.0 (main, Oct 24 2022, 18:26:48) [MSC v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: E:\ANIMESH PERSONAL\study\PAPER STUDY\MPAPER\submit 2\Decimal Bond DNA Cryptography Algorithm\Decryption\decrypt.py

Starting Decryption Process

Press 1 to enter or 2 to read the decryption key
2
The Key entered is :

Decryption Key Read Successfully

The Padding table and Binary table for A,G,C,T thus formed is
['AGCT', 'GCTA', 'CTAG', 'TAGC']
The Bond Code is :
Loading DNA Encode Library Data.......
Loading Completed...


Decryption Initiated......



Decryption Process Completed

The Decrypted Message is :

Animesh_Kairi
DNA_CRYPTOGRAPHY
Method_@
Animesh_Kairi
DNA_CRYPTOGRAPHY
29.08
```

Figure 2 Decryption Coding Output

Step 6: Plaint text is retrieved. Figure 2 shows the output of the decryption code.

4. Experimental Analysis of the Proposed Decimal-Bond DNA (DBD) Algorithm

4.1 Experimental Setup

To develop the prototype of the proposed technique, we are using Python programming language for coding. It is developed under the hardware environment of Intel(R) Core(TM) i3-9100F. Detailed system Configuration is provided below

Processor Intel(R) Core(TM) i3-9100F CPU @ 3.60GHz 3.60 GHz

Installed RAM 8.00 GB

Device ID DDBEBA7F-55AF-4410-A60A-FC338149E86A

Product ID XXXXX-71397-24932-AAOEM

System type 64-bit operating system, x64 -based processor

OS Windows 10

4.1 The output of DNA Encoding Library

The stage of the DNA encoding library is shown in Table 5.

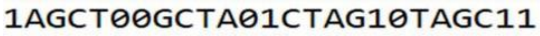
Table 5 DNA Encoding Library

AATC	ACGC	TTTG	ACAG	ACCG
ATCA	GCCT	CTCA	GTAA	ATTG
AGGT	TTAA	GTAT	ACCT	ACGG
CCAG	CCAC	GCCG	GCGC	TCTA
TTAC	TGCC	AAAT	CCGT	GGAG
CCAA	ACAC	TAAC	AGGC	CCGG
GTCA	TGGT	CAGA	ACTC	CACG
TTAT	CGCC	ACTT	CTAT	CTTA
TACA	AGAG	CGGA	GAAC	GCAG
ACAT	CTGT	ACTG	TAAA	TCAA
CGCT	CCTT	TGTT	GTCC	GACC
TCCA	GTAG	GTAC	GCGG	TGAG

TCTG	GGTG	GACA	CAAA	CGAA
GCTG	TGGG	TTGG	CGTC	TCAG
GTTG	AACG	TAGC	AAAC	TTCG
GGTA	AACC	GAAT	TCGC	GGGT
AGTC	TGTG	AGGA	GTGG	TTCA
GTCG	AGCA	TTTT	TGCG	CAAG
GTGA	CATA	CGGG	TTGT	CAGG

4.2 Output of Key

Both the public key which is the DNA encoded library (as per Table 1) in this proposed method and Private Key in the below figure 3 size is fixed at 25 bytes in the proposed method.

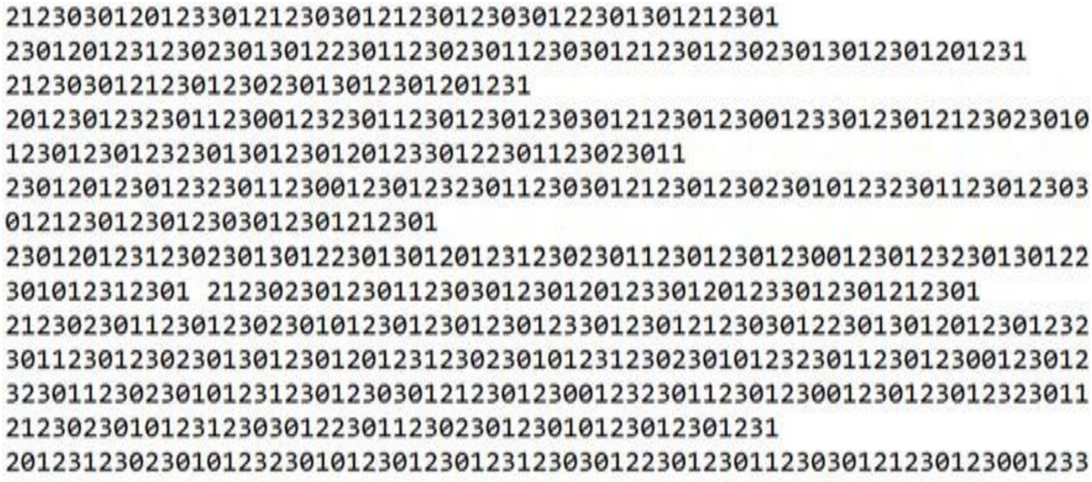


```
1AGCT00GCTA01CTAG10TAGC11
```

Figure 3 Private Key

4.3 The output of the Encryption Process

The encryption process output is cipher text. Here the plain text is converted into DNA using DNA encode library, DNA padding table, and bond table and converted into decimals of 0,1,2 and 3 as described earlier in this paper. The cipher text snap is shown below after using the proposed cryptography technique in Figure 4.



```
212303012012330121230301212301230301223013012l2301
23012012312302301301223011230230112303012123012302301301230l201231
21230301212301230230130123012O1231
20123012323011230012323011230123012303012123012300123301230121230230l0
1230123012323013012301201233012230112302301l
23012012301232301123001230123230112303012123012302301012323011230123O3
0121230123012303012301212301
230120123123023013012230130120123123023011230123012300123012323013O122
301012312301 21230230123011230301230120123301201233012301212301
21230230112301230230101230123012301233012301212303012230130120123O1232
3011230123023013012301201231230230101231230230101232301123012300123012
3230112302301012312301230301212301230012323011230123001230123012323011
21230230101231230301223011230230123010123012301231
20123123023010123230101230123012312303012230123011230301212301230O1233
```

Figure 4 Cipher Text of Proposed Decimal Bond Cryptography Technique

4.4 Comparative Analysis of Encryption and Decryption Times for the Proposed Technique Across Various Plain Text Sizes

In this section, we do a comparison study of various cryptography techniques [6]. Raw Text and Cipher Text size is compared in Table 6. Various cryptography techniques are compared with this method in terms of original text and cipher text.

Table 6 Comparison of Original Text Size and Cipher Text Size

Algorithms	**Original File in char**	**Original text in bytes**	**Cipher text in bytes**
DNA-One Time Password (OTP)	500 char	1070	4352
	1500 char	1685	6912
	2500 char	3070	12288
	3500 char	4834	19456
	4500 char	6056	24320
Time Password (OTP)	500 char	1070	1280
	1500 char	1685	1792
	2500 char	3070	3072
	3500 char	4834	4864
	4500 char	6056	6144
DNA- Rivest-Shamir-Adleman (RSA)	500 char	1070	113028
	1500 char	1685	177868
	2500 char	3070	324324
	3500 char	4834	510320
	4500 char	6056	651376
Rivest-Shamir-Adleman (RSA)	500 char	1070	32694
	1500 char	1685	39836
	2500 char	3070	54793
	3500 char	4834	86284
	4500 char	6056	103945
Compressed DNA- Rivest-Shamir-Adleman (RSA)	500 char	1070	40164
	1500 char	1685	57872
	2500 char	3070	91948
	3500 char	4834	110828
	4500 char	6056	106632
Decimal Bond DNA Cryptography (Proposed Algorithm	500 char	1070	7044
	1500 char	1685	14086
	2500 char	3070	28170
	3500 char	4834	42254
	4500 char	6056	56338

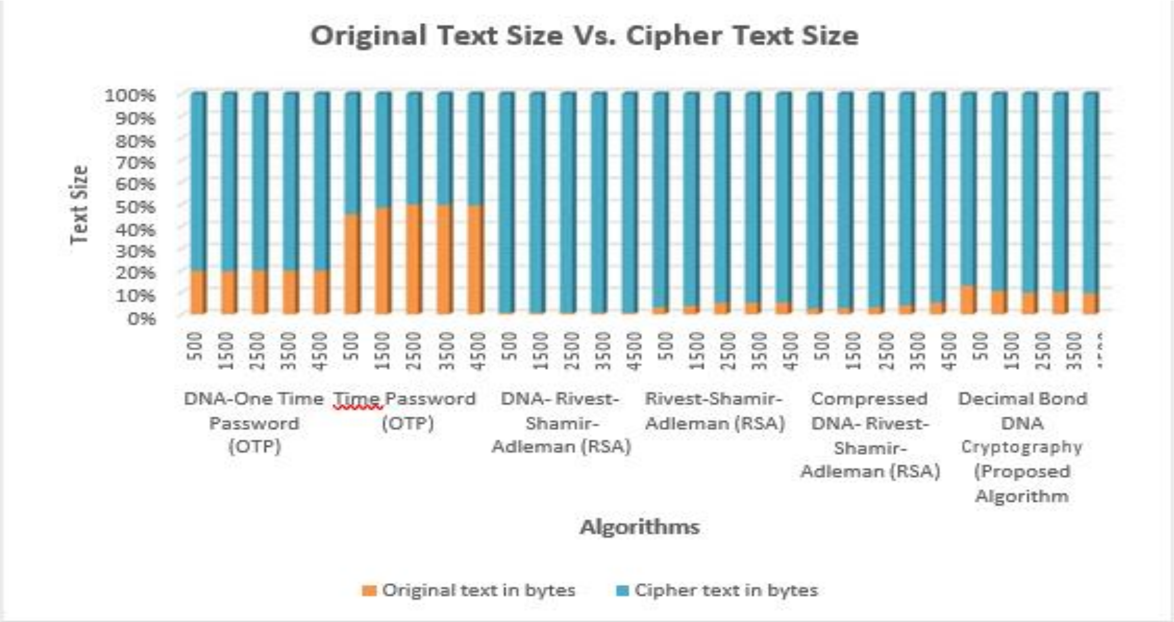


Figure 5 Original Text Size Vs. Cipher Text Size (in Bytes)

As we can see from Figure 5 that the size of the cipher text is quite large in comparison to other methods since cryptography always creates highly secure cipher text.

4.5 Proposed Technique Across Various Plain Text Sizes

We consider plain texts of various sizes similar to DNA cryptography algorithms. The results are presented in Table 7. It is evident that despite the increase in plaintext size, the encryption time remains notably lower compared to other DNA cryptography techniques. In subsequent sections of this paper, we conduct a comprehensive comparison between our proposed method and various existing DNA-based approaches.

Table 7 Comparison of Encryption and Decryption Times for the Proposed Technique Across Various Plain Text Sizes

Plaintext Size (bytes)	Encryption Time (ms)	Decryption Time (ms)
10	12.3513	11.0717
20	12.7075	10.7157
40	13.4275	10.471
80	13.7831	10.4592
100	13.9047	10.2435
500	14.6686	10.1259
1000	15.212	10.123
2000	16.541	10.121
4000	17.9325	10.11

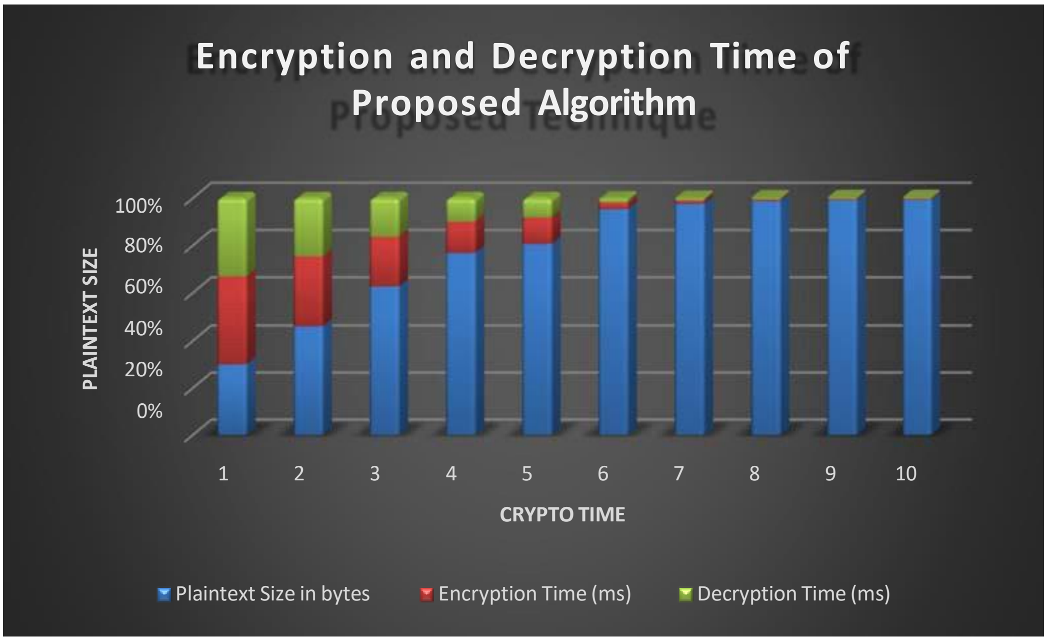


Figure 6 Encryption and Decryption Time of Proposed Algorithm

In this section also we compared the time of various cryptography techniques [6].

From Table 9, we made the comparison among various times taken for encryption and decryption techniques [14] and we observed that our proposed technique is a quite less time-consuming method in terms of crypto time as shown in Table 9. Similarly, the comparison of encryption times for various DNA cryptography algorithms is graphically presented in Figure 8, and the comparison of decryption times for various DNA cryptography algorithms is graphically shown in Figure 9.

Table 9 Comparison of Different DNA Crypto Time

Different Methods	Kaundal [12]		Pramanik [13]		Paul [14]		Pramod [15]		**Proposed Technique (DBD Cryptography)**	
Plain text length (char)	Encry-ption Time (ms)	Decry-ption Time (ms)	Encry-ption Time (ms)	Decry-ption Time (ms)	Encry-ption Time (ms)	Decry-ption Time (ms)	Encry-ption Time (ms)	Decry-ption Time (ms)	**Encry-ption Time (ms)**	**Decry-ption Time (ms)**
Series 1: 10	15	37	20	49	18	42	11.31	5.87	**12.351**	**11.0717**
Series 2: 20	27	44	35	68	28	57	16.42	9.57	**12.7075**	**10.7157**
Series 3: 40	38	110	49	159	42	123	31..1	24.75	**13.4275**	**10.4712**
Series 4: 80	70	270	96	414	84	310	62.27	49.11	**13.7831**	**10.4592**
Series 5: 100	96	365	105	530	112	415	85.45	61.25	**13.9047**	**10.2435**
Series 6: 500	420	1660	466	1768	486	1896	312.45	245.75	**14.6686**	**10.1259**

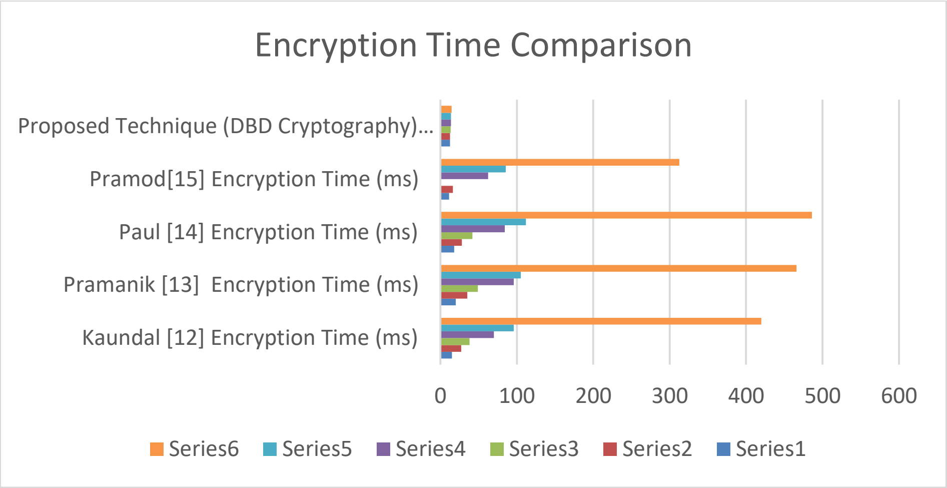


Figure 8 Encryption Time Comparison of Various DNA Cryptography Algorithm

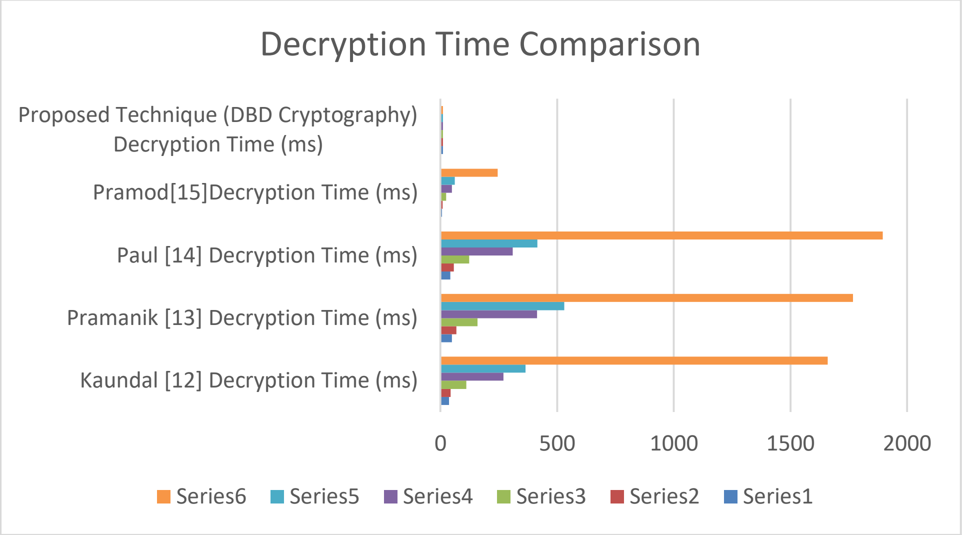


Figure 9 Decryption Time Comparison of Various DNA Cryptography Algorithm

5. Security Analysis

The security evaluation of the suggested technique is presented in this section.

5.1 Brute-Force Attack

A brute-force attack [7] is used by the attackers to try and discover the private key of the encryption. Brute force attacks in DNA cryptography are theoretically possible but extremely impractical due to the unique properties of DNA and the vast complexity of DNA-based encryption schemes. Here are several reasons why a brute-force attack in DNA cryptography is not feasible:

- **Massive Search Space:** DNA cryptography involves encoding data as sequences of nucleotide bases (A, T, C, and G). The search space for a brute force attack is exponentially large, as the length of the DNA sequence and the number of possible bases increase. This makes it practically impossible to exhaustively search all possible combinations.
- **High Information Density:** DNA has a very high information density, meaning a small amount of DNA can represent a vast amount of data. Even if you were to obtain a DNA sample, it would be challenging to determine if it contains encrypted data, let alone decipher the actual information.
- **Specialized Equipment:** DNA manipulation and sequencing require specialized laboratory equipment and expertise, which are not readily available to the average attacker. This significantly raises the bar for anyone attempting a brute-force attack.
- **Biological Constraints:** DNA cryptography often involves encoding data in a biologically meaningful way to ensure stability and durability. This means that the encrypted DNA sequences may not resemble random strings, further complicating brute force attempts.
- **Time and Resources:** Even with advanced

technology, the time and resources required to perform brute-force attacks on DNA-encoded data would be astronomical. It would likely take centuries or longer to decrypt even a moderately complex DNA-encoded message.

- **Ethical and Legal Barriers:** DNA is biological material with ethical and legal considerations regarding its handling and manipulation. Unauthorized access to DNA, especially for malicious purposes, is subject to strict regulations and laws.

The suggested approach is resistant to brute-force attacks as a result.

5.2 Known Plaintext Attack (KPA)

The attacker is given the ability to view both the plaintext and the cipher text that goes with it in a known plaintext attack (KPA). To lessen the likelihood of a Known Plaintext Attack, strong encryption measures should be used. These algorithms should possess strong security properties that prevent an attacker from easily deriving the encryption key or obtaining sensitive information from the known plaintext-cipher text pairs. Implementing strong encryption algorithms and protocols can help minimize the vulnerability to Known Plaintext Attacks. During a known plaintext attack, the attacker attempts to obtain both the cipher text and the matching plaintext [8]. The objective is to figure out the secret keys and create a method for decrypting any future messages. In this proposed method, any two comparable plaintexts can be encrypted to create two completely different cipher texts. As a result, the system can withstand this kind of assault.

5.3 Cipher-text Only Attack (COA)

A Cipher text-Only Attack (COA) is a type of cryptographic attack where the attacker only has access to the cipher text (encrypted data) and has no knowledge of the corresponding plaintext or any other additional information. In the context of DNA encryption, a Cipher text-Only Attack would involve an attacker attempting to analyze and decipher the encrypted DNA sequences without any knowledge of the original plaintext or the encryption key. The goal of the attacker is to gain insight into the plaintext or recover the encryption key. The effectiveness of a Cipher text-Only Attack largely depends on the strength of the encryption algorithm and the length and quality of the cipher text. Additionally, using longer encryption keys and ensuring a high degree of randomness in the cipher text can significantly enhance the security of the encryption scheme [9].

5.4 Differential Cryptanalysis Attack

A method for analyzing pairs of plaintext and matching cipher text to quickly determine the key is known as a differential cryptography attack [10]. The suggested solution is guarded against this attack thanks to a newly formed secret key that is generated randomly and other user variables. The proposed approach enhances security since during each transmission, the same plaintext is transformed into different cipher texts depending on Table 4.

6. Conclusions

In this paper, a unique cryptosystem is introduced for ensuring secure data transmission between sender and receiver. The proposed approach involves subjecting the plaintext to diverse cryptographic operations, resulting in its transformation into a DNA sequence. The suggested method offers robust protection against various security intrusions. Notably, the suggested approach outperforms existing methods in terms of system efficiency. Therefore, the proposed DNA-based encryption system stands out as a more efficient and safe effective and secure alternative to current techniques. For future DNA cryptography endeavors, it is advisable to explore the integration of DNA microchips, thereby adding an additional layer of security. However, a limitation of this study lies in the conversion of the result of the encryption method to actual molecular DNA. Overcoming this challenge necessitates collaborative input from experts across multiple disciplines, including biologists, chemists, and computer scientists, to achieve this objective.

Authors' contributions

Each author has made substantial contributions to the conception and design of the work. AK has analyzed and designed the work and created a new method presented in the work. TB has performed the data curation formal analysis and interpretation of the data, has utilized the software, and has attained manuscript review and editing. TB has substantively revised it. The authors read and approved the final manuscript.

Funding

The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Availability of data and materials

All data generated or analyzed during this study are included in the article (and in its supplementary materials).

Acknowledgements
Not applicable.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

References

1. Madhusudhan R and Shashidhara R 2020 A novel DNA-based password authentication system for global roaming in resource-limited mobile environments. *Multimedia Tools and Applications*. 79: 2185-2212.
2. Na D 2020 DNA steganography: hiding undetectable secret messages within the single nucleotide polymorphisms of a genome and detecting mutation-induced errors. *Microbial cell factories*. 19:128.
3. Doricchi A, Platnich C, Gimpel A, Horn F, Earle M, Lanzavecchia G et al. 2022 Emerging Approaches to DNA Data Storage: Challenges and Prospects. *ACS Publications*. Volume 16, no. 11: 17552-17571.
4. Matange K, Tuck J, Matange K 2021 DNA stability: a central design consideration for DNA data storage systems. *Nature Communications* 12, Article number: 1358.
5. Najaftorkaman M and Kazazi Sadat N. 2015 A method to encrypt information with DNA-based cryptography. *International Journal of Cyber-Security and Digital Forensics* 4.3: 417-426.
6. Hammad Tareq B, Sagheer Maki A, Ahmed Taha I, Jamil N. 2020 A comparative review on symmetric and asymmetric DNA-based cryptography. *Bulletin of Electrical Engineering and Informatics* 9.6:2484-2491.
7. Salamatian S, Huleihel W, Beirami A, Cohen A, Médard M. 2019 Why botnets work: Distributed brute-force attacks need no synchronization. *IEEE Transactions on Information Forensics and Security*. 14(9):2288–2299.
8. Ye Y and Mo Y. 2020 Security for cyber-physical systems: Secure control against known- plaintext attack Science. *China Technological Sciences*. 63.9: 1637-1646.
9. Chang X, Yan A and Zhang H. 2020 Cipher text-only attack on optical scanning cryptography. *Optics and Lasers in Engineering*. 126: 105901.
10. Zhao H. Han G, Wang L, Wang W. 2020 MILP-based differential cryptanalysis on round- reduced Midori64. *IEEE Access*. 8, 95888-95896.
11. Kairi A, Gagan S, Bera T and Chakraborty M. 2019 DNA Cryptography-Based Secured Weather Prediction Model in High-Performance Computing. *Proceedings of International Ethical Hacking Conference* 2018. Pages 103-114.
12. Kaundal Kumar A, Verma A. 2015 Extending Feistel structure to DNA Cryptography. *Journal of Discrete Mathematical Sciences and Cryptography*. 18.4: 349-362.
13. Pavithran P, Mathew S, Namasudra S, Lorenz P. 2021 A novel cryptosystem based on DNA cryptography and randomly generated mealy machine. *Computers & Security*. 104: 102160.
14. Paul S, Anwar T and Kumar A. 2016 An innovative DNA cryptography technique for secure data transmission. *International Journal of Bioinformatics Research and Applications*. 2016;12(3).
doi: 10.1504/IJBRA.2016.078235.
15. Pavithran P, Mathew S, Namasudra S and Srivastava G. 2022 A novel cryptosystem based on DNA cryptography, hyperchaotic systems and a randomly generated Moore machine for cyber-physical systems. *Computer Communications*. Volume 188: Pages 1-12.