

Al-Imam Mohammad Ibn Saud Islamic University

College of Computer and Information Sciences

Department of Computer Science

**Comparative Study of Three DNA-based information hiding methods**

**By:**

|  |  |  |
| --- | --- | --- |
| Maimounah Albrahim | 436001148 | Computer Science |
| Nisreen Terkawi | 436020767 | Computer Science |
| Wafaa Alsaffar | 431020643 | Computer Science |

**Supervised by:**

Dr. Amjad Ail Alamar

Project Submitted in Fulfillment for the CS493 Course requirements

Semester-2019



Al-Imam Mohammad Ibn Saud Islamic University

College of Computer and Information Sciences

Department of Computer Science

**Comparative Study of Three DNA-based information hiding methods**

**By:**

|  |  |
| --- | --- |
| Maimounah Abdurahman Albrahim | 436001148 |
| Nisreen Suliman Terkawi | 436020767 |
| Wafaa Saad Alsaffar | 431020643 |

We hereby certify that this project satisfies the project requirements:

|  |  |
| --- | --- |
| **Approved by** | Dr. Amjad Ali Alamar |
| **Date of Approval** | 2/12/2019 |
| **Signature** |  |

|  |  |
| --- | --- |
| **Approved by** | **[Vice Dean of Educational Affairs Name]** |
| **Date of Approval** |  |
| **Signature** |  |

**تعهد**

أتعهد بعدم المشاركة في الفعاليات أو المبادرات أو المسابقات ذات العلاقة دون أخذ موافقة خطية مسبقة من الكلية، و أقر بمعرفتي أنني إذا خالفت هذا التعهد ستتم محاسبتي وفق اللوائح و الأنظمة.

**Declaration**

We Maimounah, Nisreen, and Wafaa being members of final year project group number 9, declare that this report contains only work completed by members of our group except for information obtained in a legitimate way from literature, company or university sources. All information from these other sources has been duly referenced and acknowledged in accordance with the University Policy on Plagiarism.

Furthermore, we declare that in completing the project, the individual group members had the following responsibilities and contributed in the following proportions

to the final outcomes of the project:

|  |  |  |  |
| --- | --- | --- | --- |
| **Student ID** | **Responsibility** | **% Contributed** | **Signature** |
| 436001148 | Research, Documentation, Implementation, Analysis, Testing | 33.3% |  |
| 436020767 | Research, Documentation, Implementation, Analysis, Testing | 33.3% |  |
| 431020643 | Research, Documentation, Implementation, Analysis, Testing | 33.3% |  |

**Acknowledgment**

First and foremost, we would like to present my deepest gratitude to Almighty Allah for his bounties and blessings and for giving us the ability to finish this project.

We would like to express our deep appreciation and our sincere gratitude to our supervisor Dr. Amjad Alamar for her valuable advice guidance throughout this project.

Most importantly, none of this could have happened without our families and friends for continued encouragement and support during this project and along the years of study.

**Abstract**

With the increase of dependency on the internet for most of the current systems and applications. Some important information needs high security, so we must protect the information before sending it through the internet. The most common areas of security are cryptography, steganography, or a combination of them. Cryptography is the science of protecting information by transforming data into formats that cannot be recognized by unauthorized users. Steganography is the science of hiding information using different media such as image, audio, video, text, and DNA. The steganography based on Deoxyribonucleic acid (DNA) is a newly discovered information security technology characterized by high capacity, high randomization and low modification rate that lead to increased security. There are various methods to hide information based on DNA. the main algorithms are substitution, insertion, complementary pair [1][2]. In this project, we implement three algorithms to compare them in terms of capacity, cracking property, Bit Per Nucleate (BPN) and payload. The selected methods include some cryptography algorithms which is Playfair. It also includes stenography algorithms such as the insertion method and the substitution method. We compare methods based on measures to find differences between them to help future researchers design or improve methods.

**Keywords:**

Information security, Steganography, Cryptography, DNA, Cracking probability, Information hiding, Substitution, Insertion, complementary pair.

**Abstract (in Arabic)**

أصبحت العديد من الأنظمة والتطبيقات تعتمد على الانترنت في تناقل بياناتها، ونظرا لحساسية هذه البيانات والمعلومات إذ قد تكون خاصه ومكلفه فإن حمايتها بتوفير السرية والموثوقية لها اثناء انتقالها بين المرسل والمستقبل خلال شبكة الانترنت أصبح مطلب أساسي. ولهذا ازداد الاهتمام بعلم أمن المعلومات وتطبيق أكثر المجالات والاتجاهات شيوعا فيه كالتشفير والإخفاء، وذلك باستخدام خوارزمياتهم منفصلة او الجمع بينهما لتوفير حماية عالية. منهج التشفير يعتمد على العمليات الحسابية بينما الإخفاء على دمج الوسائط ببعضها سوآءا كان نص كتابي داخل صوره والعكس أيا كانت الوسائط.  في هذه الدراسة نحن سنقارن بين ثلاث طرق جمعت بين التشفير والإخفاء استنادًا إلى الحمض النووي لما فيه من مميزات وسعة عالية لتخزين البيانات وصعوبة في تخمين تسلسله. بعد توحيد البيانات المستخدمة في الدراسة ستتم المقارنة بين الطرق الثلاث من حيث السعة واحتمال كسر الخوارزمية وعدد البتات لكل النوكليوتيد و الطول المتبقي لتسلسل الحمض النووي بعد استخراج النوكليوتيدات الموجودة في الأصل، للوصول الى نتائج تساعد الباحثين مستقبلا في اختيار الطريقة الأفضل بنآءا على النتائج التي سوف تكون في هذه الدراسة. هذه الطرق الثلاث على النحو التالي: الأولى طريقة التبديل بالبت الأخير، الطريقة الثانية وهي الإضافة والطريقة الثالثة هي التبديل العام بالمكملات.

1. Table of Contents

[List of Abbreviations xi](#_Toc26177196)

[List of Figures xii](#_Toc26177197)

[List of Tables xiii](#_Toc26177198)

[Chapter One: Introduction 1](#_Toc26177199)

[1.1. Introduction 2](#_Toc26177200)

[1.2. Problem Definition 3](#_Toc26177201)

[1.3. Aim and Objectives 4](#_Toc26177202)

[1.4. Project Timeline 4](#_Toc26177203)

[1.5. Conclusion 4](#_Toc26177204)

[Chapter Two: Literature Review 5](#_Toc26177205)

[2.1. Introduction 6](#_Toc26177206)

[2.2. Background 6](#_Toc26177207)

[2.2.1 Steganography 6](#_Toc26177208)

[2.2.2 DNA sequence 8](#_Toc26177209)

[2.2.3 Encryption with steganographic 9](#_Toc26177210)

[2.3. Related Works 12](#_Toc26177211)

[2.4. Conclusion 22](#_Toc26177212)

[Chapter Three: Methodology 22](#_Toc26177213)

[3.1. Introduction 24](#_Toc26177214)

[3.2. Theoretical formalization and algorithms design 24](#_Toc26177215)

[3.3.1 Algorithm 1: based on the substitution method [6] 24](#_Toc26177216)

[3.3.2 Algorithm 2: based on the insertion method [10] 30](#_Toc26177217)

[3.3.3 Algorithm3: based on generic complementary substitution method [11] 32](#_Toc26177218)

[3.3. Experimental Design 35](#_Toc26177219)

[3.3.1 The algorithm based on the substitution method [6] 35](#_Toc26177220)

[3.3.2 The algorithm based on the insertion method [10] 36](#_Toc26177221)

[3.3.3 The algorithm based on the substitution method 37](#_Toc26177222)

[3.3.4 comparison measurements 40](#_Toc26177223)

[3.4. Conclusion 40](#_Toc26177224)

[Chapter Four: Results 42](#_Toc26177225)

[4.1. Introduction 43](#_Toc26177226)

[4.2. Project Results 43](#_Toc26177227)

[4.3. Experimental Results 45](#_Toc26177228)

[4.3.1 Hiding Message: 45](#_Toc26177229)

[4.3.2 Extracting Message 46](#_Toc26177230)

[4.4. Conclusion 49](#_Toc26177231)

[Chapter Five: Discussion 50](#_Toc26177232)

[5.1. Introduction 51](#_Toc26177233)

[5.2. Discussion of the Results 51](#_Toc26177234)

[5.2.1. Discussion of capacity measure 51](#_Toc26177235)

[5.2.2. Discussion of payload measure 51](#_Toc26177236)

[5.2.3. Discussion of BPN measure 52](#_Toc26177237)

[5.2.4. Discussion of cracking properties measure 52](#_Toc26177238)

[5.2.5. Difficulties 53](#_Toc26177239)

[5.3. Conclusion 53](#_Toc26177240)

[Chapter Six: Conclusion and Future Work 54](#_Toc26177241)

[6.1. Introduction 55](#_Toc26177242)

[6.2. Chapter Body 55](#_Toc26177243)

[6.2.1. Contributions 55](#_Toc26177244)

[6.2.2. Future Work 55](#_Toc26177245)

[6.3. Conclusion 55](#_Toc26177246)

# List of Abbreviations

|  |  |  |
| --- | --- | --- |
| AES | **:** | Advanced Encryption Standard |
| ASCII | **:** | American Standard Code for Information Interchange |
| BPN | **:** | Bit Per Nucleate |
| BCR |  | Binary coding rule |
| DNA | **:** | Deoxyribonucleic Acid |
| HECC | **:** | Hyperelliptic Curve Cryptography |
| LSB | **:** | Least significant bit |
| NIST | **:** | National Institute of Standards and Technology |
| PCR | **:** | polymerase chain reaction |
| RSA | **:** | Rivest-Shamir-Adleman |
| SHA-2 | **:** | Secure Hash Algorithm 2 |
| TLSM | **:** | Table Lookup Substitution Method |

# List of Figures

[Figure 1 Project Timeline 4](#_Toc26169488)

[Figure 2 DNA Structure [13]. 8](#_Toc26169489)

[Figure 3 DNA Nucleotide Bases [13]. 9](#_Toc26169490)

[Figure 4 Vigenère Table [15]. 11](#_Toc26169491)

[Figure 5 Stages of hybrid steganography and cryptography. 12](#_Toc26169492)

[Figure 6 sender side flowchart 22](#_Toc26169493)

[Figure 7 Data Encryption Flowchart [6] 23](#_Toc26169494)

[Figure 8 Steganography steps [6] 25](#_Toc26169495)

[Figure 9 receiver side flowchart 25](#_Toc26169496)

[Figure 10 Steganography steps [6] 26](#_Toc26169497)

[Figure 11 Steganography steps [6] 27](#_Toc26169498)

[Figure 12 Alogrithm insersion processing 29](file:///C:\Users\user\Downloads\semester%209\updates%20in%20our%20paper\endd.docx#_Toc26169499)

[Figure 13 The proposed scheme followed 30](#_Toc26169500)

[Figure 14 A digital coding of DNA bases 31](#_Toc26169501)

[Figure 15 Complementary rules for the nucleotide bases of DNA 31](#_Toc26169502)

[Figure 16 message base and after steganography base 31](#_Toc26169503)

[Figure 17 Example hiding message for Algorithm 1 based on LSB substitution 42](#_Toc26169504)

[Figure 18 Example hiding message for Algorithm 2 based on insertion 42](#_Toc26169505)

[Figure 19 Example hiding message for Algorithm 3 componentry generic substitution 43](file:///C:\Users\user\Downloads\semester%209\updates%20in%20our%20paper\endd.docx#_Toc26169506)

[Figure 20 Example extract message for Algorithm 1 based on LSB substitution 44](#_Toc26169507)

[Figure 21 Example extract message for Algorithm 2 based on insertion 44](#_Toc26169508)

[Figure 22 Example extract message for Algorithm 3 complementary generic substitution 45](file:///C:\Users\user\Downloads\semester%209\updates%20in%20our%20paper\endd.docx#_Toc26169509)

# List of Tables

[Table 1 Lookup table substitution rule 8](#_Toc26170125)

[Table 2 2-Bits Binary Coding Rule [5 12](#_Toc26170126)

[Table 3 Comparison of algorithms related to our work 19](#_Toc26170127)

[Table 4 4-bit BCR 23](#_Toc26170128)

[Table 5 Distribution for codons on English alphabet [31] 24](#_Toc26170129)

[Table 6 5 Ambiguity numbers (0 to3) to binary 25](#_Toc26170130)

[Table 7 laws of our algorithms 37](#_Toc26170131)

[Table 8 Results LSB substitution of hiding message into different DNA sequences 40](#_Toc26170132)

[Table 9 Results insertion of hiding message into different DNA sequences 41](#_Toc26170133)

[Table 10 Results complementary generic insertion of hiding message into different DNA sequences 41](#_Toc26170134)

[Table 11 Cracking property comparison between our algorithms 41](#_Toc26170135)

# Chapter One: Introduction



## Introduction

Data transfer through the Internet has become necessary and important but the transmission of sensitive information through the Internet is at high risk, its unreliable and unsafe. Therefore, exchanging messages between the sender and receiver is required to be in a confidential manner in order to avoid being hacked or susceptible to threats through the internet. For example, in 2015, the second-largest health insurance in the United States was exposed to a covert attack that extends for weeks. Their users' contact information and social security numbers were stolen and maybe their medical information also. Now anthem is considering encrypting all its data to protect it from attackers.  Given the need for robust data protection, information security has become an important part of protecting valuable and sensitive information for organizations and users. Protection can be investigated using steganography or cryptography. Both are to protect sensitive information from unauthorized access, therefore provide data security and confidentiality [1][2].

Encryption is a technique for preventing third parties from reading a secret message. This method converts any type of data into an encrypted format meaning incomprehensible format. Some of the techniques in the encryption are Playfair, Rivest Shamir Adleman (RSA), Advanced Encryption Standard (AES), etc. [3][4][5]

Steganography is a technique for hiding a secret message within a cover media such as text, image, audio, video, or DNA without changing its quality. This method does not change the message information just makes the message unclear to any illegal read. We will focus on steganography based on DNA because there are advantages, which are it has a huge data storage capacity, and that the original DNA and the fake DNA with secret data cannot be distinguished because the DNA sequence is too long [6][7][8]. The main DNA-based steganography methods are the insertion method, the complementary pair method and the substitution method [5][9].

In general, steganography is better than cryptography because it is less vulnerable to attack, hackers cannot disclose confidential message because the secret message does not attract attention to itself [4]. However, there are some approaches that combine the steganography with one or more encryption ways in which the message has been encrypted first then hidden in one of the selected media. Steganography and cryptography combined to provide a double layer of security to prevent the detection of the original message. They are notable for achieving confidentiality, low cracking, and complexity [5].

In recent, many algorithms have been extracted in the field of steganography based on DNA to prevent unauthorized access and increase data security. In this work, we compare three DNA-based steganography algorithms on several properties: capacity, payload, BPN, and the cracking probability [6][10][11]. Also, we will clarify the following characteristic for the selected algorithms: the type of data, size of data, the type of encryption algorithm used, the type of DNA-based steganography algorithm used, bind or not bind. Our comparison will be helpful for future researchers' design or to improve methods being more secure.

The rest of the chapter is organized as follows: Section 1.2 problem definition, Section 1.3 aims, and objectives, Project timeline in section 1.4. Finally, the chapter is concluded in the last section.

## Problem Definition

In our real-world sometimes, we may have critical information to exchange with others. But may have a high probability to see and modified it any message exchanged from the others. So, we need to contact others secretly to protect the content of our communication from the outer people. The security field defined as a solution to this problem, but it is a huge field for use. When we want to use one of the security methods we must know which one is better to use. In this project, we did a comparison between three steganography algorithms based-DNA sequence by three measurements are capacity, payload, and PBN to encrypt the message inside the sequence to prevent the outer people from knowing the message content. This report answers the related research questions:

* What is the steganography field?
* What are the features of our project?
* Why our work is helpful?
* How we measured the algorithm performance?
* What is the best algorithm to use between the three algorithms?
* What is the result of measuring algorithms?

## Aim and Objectives

The main aim of this project is to compare three DNA-based steganography algorithms based on the following features: capacity, cracking probability, BPN, and also modification rates.

Our aim will be achieved by applying the following objectives:

1. Survey existing DNA-based steganography algorithms.
2. Implement selected algorithms.
3. Compare these algorithms based on specified comparison measurements.
4. Analyze the comparison results.
5. Evaluate these results.

## Project Timeline

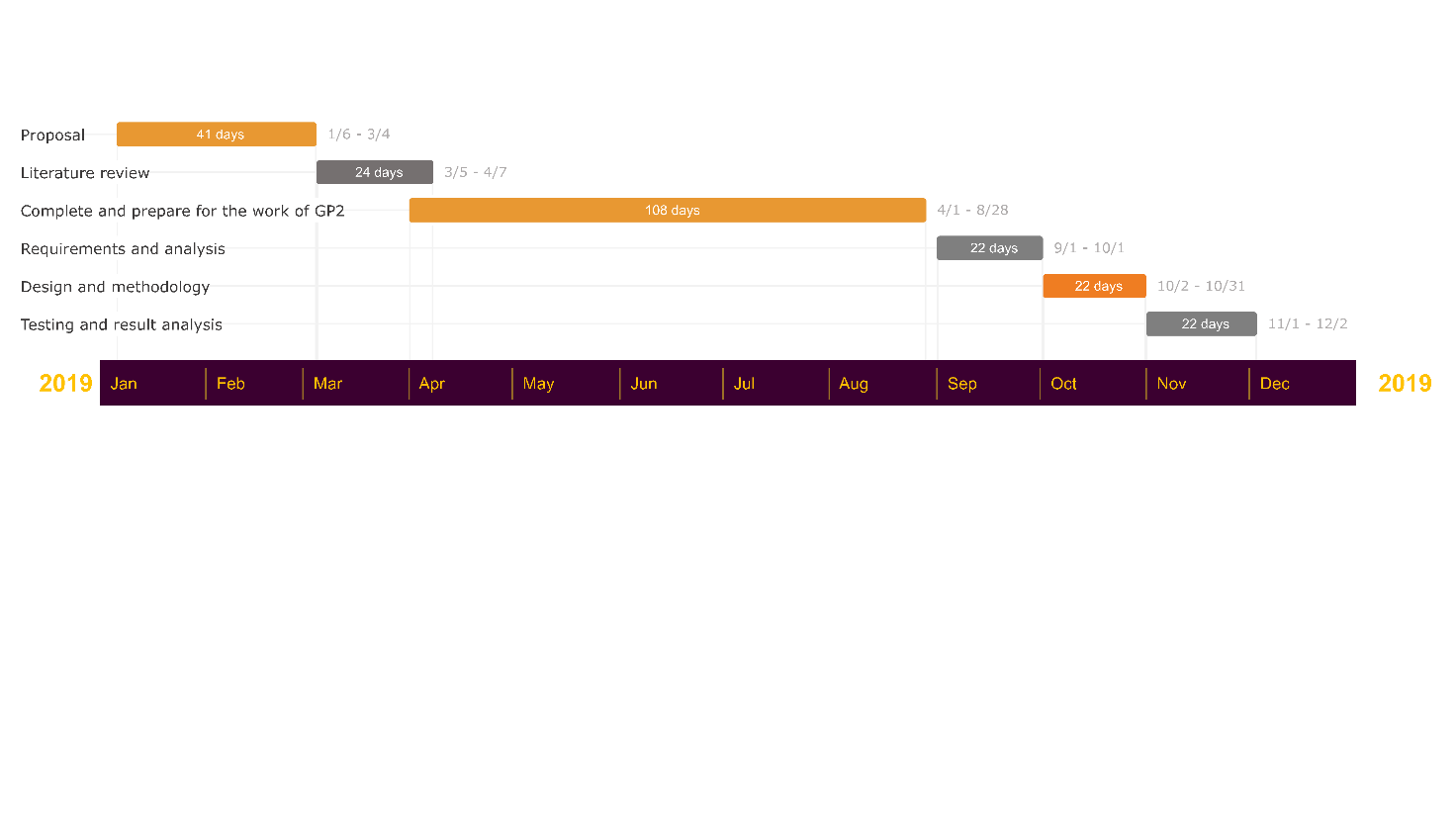
****

Figure 1 Project Timeline

## Conclusion

In this chapter, we introduced a problem that we try to solve it. We work to measure the features of three algorithms of hiding data with DNA sequence technology. These comparison features used to evaluate them in terms of capacity, cracking property, BPN, and payload. Then determine which algorithm is better to use for different kinds of projects depend on their requirements and it helps researchers to develop the methods to make them perform better. We discuss the background and related work in the next chapter.

# Chapter Two: Literature Review

## Introduction

In recent years, the need for electronic communications has become essential and has led to an increase in the number of hackers, of electronic eavesdropping and electronic fraud. So, security becomes an urgent necessity, particularly when using sensitive information such as in medical, or covert applications. As a result, various types of data protection mechanisms were proposed. Steganography and cryptography are two science of security that support these purposes [1][2][3][4][5].

This chapter helps the reader to understand the scientific background related to our study. In addition, an overview of the related work. This chapter is introducing the necessary background of our study and the related work of the recent DNA based steganography algorithms, and the last section is the conclusion.

## Background

In the following sections, we provide a brief biological background and overview of the steganography and cryptography techniques.

### Steganography

Steganography is a technique of hiding confidential information in an ordinary -not secret- cover carrier without changing the quality of cover carrier to prevent any unauthorized party from detecting or discovering the existing of a confidential message within it. It includes a secret message and cover carrier. A secret message is a message you want to send in a secret way. This message could be text, audio, image or video. Cover carriers contain the hidden information embedded in it such as text, image, video, and DNA. Text-based steganography which is used the number of tabs, white spaces, capital letters and every nth letter of a word for the hiding process. Image-based steganography using pixel intensity as the hiding process. Audio-based steganography in which the digital audio signal- like WAV, AU, and MP3- is used for the hiding process. Video-based steganography which is a collection of images and sounds- using the video signal for the hiding process. Whereas, the DNA-based steganography used DNA sequence to hide data [3][4].

In general, there are three basic DNA-based steganography methods which are: insertion algorithms, substitution algorithms, and complementary rule algorithms.

1. **The insertion system steganography-based DNA:**

The secret message and DNA reference sequence are translated into a binary system according to the binary coding rule (BCR). After that, the DNA sequence and the secret message are divides into equal-sized segments in order to insert each part of the secret message after each part of the DNA reference and the last step converts it into the DNA sequence. There is a drawback to this technique is the increase in redundancy, also the length of faked DNA is higher than the length of the original DNA, this shows the intruder that there is a secret message within it [5][9].

1. **The complementary rules steganography-based DNA:**

Secret data is hidden in the DNA reference sequence using a complementary rule. First, you must select a DNA sequence that contains all the permutations of the nucleotide pairs of secret messages. Then hiding it in a DNA reference sequence by searching for the indicators for each complementary DNA nucleotides segment of M in the DNA reference sequence. that is increasing as a message size increase [5][9].

1. **The substitution system steganography-based DNA:**

The positions specified in the DNA reference are selected randomly and the receiver must know these positions. Then at least one n base to replace each letter in the message with DNA contents in specific positions. This technique is used either by using a lookup substitution table or least significant bit (LSB) substitution mechanism or any other substitution rule. Advantages of this technique preserving the length of DNA after the hidden of the message. This technique is more efficient than other technologies because it provides more complexity and performance better [5][9].

Using a lookup substitution table to replace each 2-bits binary message into a DNA nucleotide based on conversion function. For example, lookup substitution table as shown in table 1, the first row is two bits in the reference DNA sequence, a msg column refers to two bits of the confidential message, the sub-column refers to the bits It will replace it. According to the equation is replaced [5][9][12].

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 00 | | 01 | | 10 | | 11 | |
| Msg | Subs | Msg | Subs | Msg | Subs | Msg | Subs |
| 00 | 01 | 00 | 00 | 00 | 11 | 00 | 01 |
| 01 | 10 | 01 | 11 | 01 | 00 | 01 | 00 |
| 10 | 11 | 10 | 10 | 10 | 01 | 10 | 11 |
| 11 | 00 | 11 | 01 | 11 | 10 | 11 | 10 |

Table 1 Lookup table substitution rule

Using LSB substitution to replace the least significant bit of each codon with the bits of the message to be hidden either into purine base (A & G) or pyrimidine base (T & C) [5][9].

### DNA sequence

DNA sequencing is the process of determining the [nucleic acid sequence](https://en.wikipedia.org/wiki/Nucleic_acid_sequence) as shown in figure 2. For each nucleotide is made up of three parts: a deoxyribose (5-carbon sugar) in figure 2 labeled as S, a phosphate group in figure 2 labeled as P, and one of four nitrogenous bases Adenine (A), Thymine (T), Guanine (G), or Cytosine (C). The nitrogenous bases have main two groups are purines and pyrimidines the first two types (A) and (G) are purines and the other (T) and (C) are pyrimidines [13].

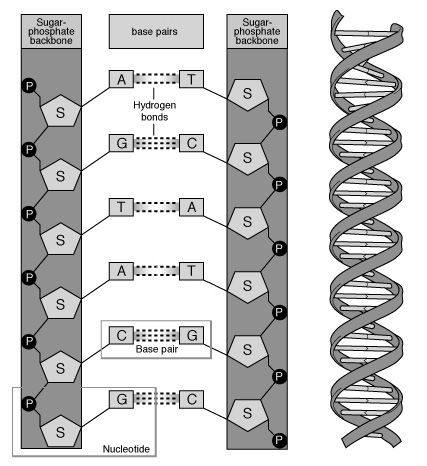


Figure 2 DNA Structure [13].

The nitrogenous bases of the two separate polynucleotides are bound together, Purine Adenine (A) always pairs with the pyrimidine thymine (T) are linked by two bonds, and Pyrimidine Cytosine (C) always pairs with the purine Guanine (G) are linked by three bonds as shown in figure 3 [13].

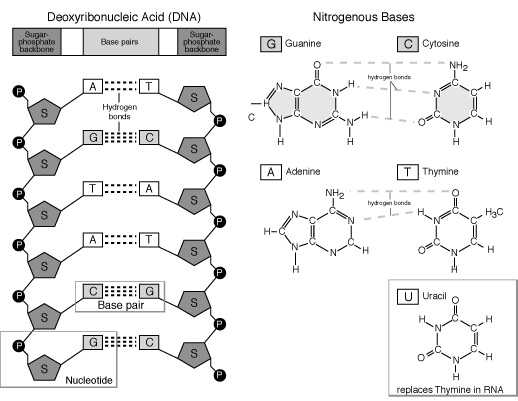


Figure 3 DNA Nucleotide Bases [13].

A codon is a sequence of 3-letter combinations DNA nucleotides that used either to a specific amino acid or stop signal during protein synthesis, Number of possibilities for codon number 64 possible codon. For example, the codon CAG represents the amino acid glutamine, and TAA is a stop codon [13].

The main advantages of DNA molecules make it effective to be a cover carrier. Because the advantage's DNA is better than any other media for hiding data. Features are high storage capacity, simple data conversion to DNA sequence, and also the DNA sequence complexity and randomness provides a great uncertainty [14].

### Encryption with steganographic

However, steganography suffers from some drawbacks in terms of computation and security. The security issues can be enhanced by applying encryption to the original message before hiding it. Encryption is any type of data transformation from a readable form to an encrypted version that can only be decrypted by an authorized party [1][2]. In general, there are three dimensions of encryption, first, the operations used to convert plaintext to ciphertext its type Substitution or Transposition, secondly the number of keys used symmetric or Asymmetric encryption, and thirdly the way in which plain text is processed as block cipher or stream cipher processes. Substitution is mapping each element in the plaintext to another element in the ciphertext. It includes two methods: monoalphabetic and polyalphabetic cipher. The most widely used polyalphabetic cipher is Playfair cipher. Transposition is only a rearrangement of the elements in the plaintext. Symmetric encryption is both the sender and the receiver use the same key the most widely used symmetric-key cipher is the AES. Asymmetric encryption the sender and receiver each use a different key the RSA encryption algorithm is the most widely used public-key algorithm. Block cipher is processing the input one block of elements at a time, producing an output block for each input block example of algorithms there use it RSA and AES. A stream cipher is processing the input elements continuously, producing output one element at a time. The Encryption methods we will focus on here are AES, RSA, and Playfair [1][2].

The AES algorithm is a symmetric [block cipher](https://searchsecurity.techtarget.com/definition/block-cipher). It comprises of a sequence of rounds operations, with (substitutions) replacing inputs by specific outputs and (permutations) involve shuffling bits around, controlled by the key. The general structure of the AES process consists of a block of plaintext, a key, number of rounds in the process. The cipher takes a plaintext block size of 128 bits or 16 bytes. The key size is 16, 24, or 32 bytes (128, 192, or 256 bits). Depending on the length of the key, the algorithm is called AES-128, AES-192, or AES-256. The round number in the process consists of 10, 12 or 14 rounds, each full round consists of four separate functions: byte substitution, shift rows, mix columns, and add round key [1][2].

The RSA algorithm is the first public-key algorithm developed in 1977. It is a block cipher in which the plaintext and ciphertext are integers between 0 and n-1 for some n which is known for both parties along with e. There are two different [keys](https://simple.wikipedia.org/wiki/Key_(cryptography)): public key PU = {e,n} can be given to anyone and private key PR = {d,n} must be kept in secret. They are an integer which is relatively prime to the ϕ(n) and d is a multiplicative inverse of it. The encryption and decryption process will be done using the following form, where M and C refer to the plaintext and ciphertext consequently:

C = Me mod n

M = Cd mod n = (Me)d mod n = Med mod n

The strength of the RSA algorithms depends on the complexity of exponent and inverse computation which is difficult. In addition, it is based on the mathematical difficulty of finding the prime factors of a composite number which almost infeasible [1][2].

The Playfair algorithm is [symmetric](https://en.wikipedia.org/wiki/Symmetric_key_algorithm) [encryption](https://en.wikipedia.org/wiki/Encryption) and substitution. The technique encrypts pairs of letters (bigrams or diagrams) by processing digraphs in the plaintext as units rather than as single letters. The Playfair algorithm is the use of a 5 × 5 matrices of letters constructed using a keyword, which the sender and receiver must agree on a keyword. The cipher replaces each pair of letters in the plaintext with another pair of letters [1][2]. However, the Playfair algorithm used in DNA-based algorithms, the matrix size, and its internal contents could be changed depending on the DNA.

The Vigenere algorithm is [symmetric](https://en.wikipedia.org/wiki/Symmetric_key_algorithm) [encryption](https://en.wikipedia.org/wiki/Encryption) and substitution. The Vigenere uses a 26×26 table with A to Z as the row heading and column heading as shown in figure 4 [15].  The top row of this table has the 26 English letters which are plaintext alphabet and the left column is the key alphabet. Starting process with the second row, each row has the letters shifted to the left one position in a cyclic way [15].

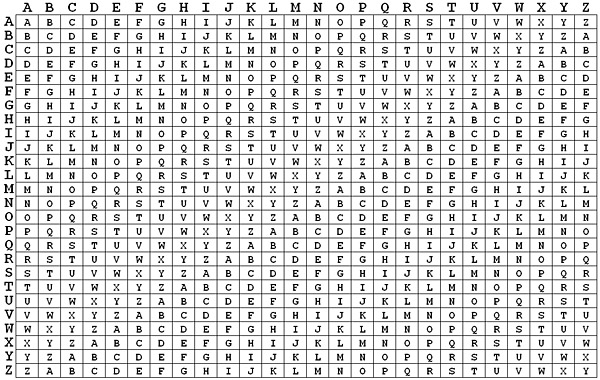


Figure 4 Vigenère Table [15].

In general, the hybrid approach of steganography that combining steganography with cryptography is passed through the following stages as illustrated in figure 5.

Figure 5 Stages of hybrid steganography and cryptography.

In order to convert from binary format of the message to DNA format, have to use a binary coding rule. The general rule is N = 2n. Each DNA has four bases: A, C, G, and T. It must select an equivalent binary form for each base. This method makes the randomness properties which is efficient for both computing and cryptography. Therefore, this makes N! permutations. As shown in Table 2 example for BCR that is each two bits assign for one nucleotide [5][9].

|  |  |
| --- | --- |
| DNA Nucleotides | Binary Representation |
| A | 00 |
| C | 01 |
| G | 10 |
| T | 11 |

Table 2 2-Bits Binary Coding Rule [5]

## Related Works

G. Hamid et al. in [14] proposed a technique that combining encryption and steganography. For encryption, they use the Playfair algorithm to encrypt the message then hidden the encrypted message within DNA using the LSB method. The generic N-bits binary coding rule is used to convert the message to DNA. They evaluate their algorithm from three-point of view security, capacity, and payload. The security of the proposed algorithm is measured by the calculation of the probability of cracking and they found it very low. From the capacity point of view, they found that whenever DNA sequence length increases it leads to increase capacity. And they found that the payload is equal zero.

P.Vijayakumar et al. in [16] used stenographic hiding methods using LSB techniques, RSA algorithm, DNA cryptography, and DNA steganography to have less embedding capacity, high MSR and PSNR value compared with other stenographic methods. This DNA steganography based Hyperelliptic Curve Cryptography (HECC). The results of increased processing time are 30% for the encryption process and 42 % for the decryption process.

M.Sajisha KS and Dr.S.Mathew in [8] propose multilayer security combining cryptography and steganography by using the LSB method for steganography AES encryption algorithm. First, they convert the massage to the binary form based on the DNA sequence. The second step converts the DNA form into ASCII value. The third step converts ASCII value to amino acid-based on it is the table. Forth step perform swapping to obtain a new form of DNA. Then apply the AES algorithm to encrypt the data. Finally, convert the DNA cipher to binary digits.

A. Khalifa et al. [17] use a novel method to prevent exploring data and providing a double layer of security that relies on cryptography and steganography means. This work accomplished with two phases of hybrid technique data hidden by using LSB. In this work, they apply binary coding to increase building a strong algorithm, the coding that leads to using zero to avoid attracting attention to fake sequences. The algorithm proposed in this paper still blind when data extracts from DNA sequence also, the real DNA reference couldn’t be able to expand.

G.Hamed et al. in [6] talked about introducing the LSB method as a novel method of DNA steganography to manage key exchanging in the hybrid system. They rely on represent thy key as binary code after encryption to hide it in DNA sequence with last significant bases which preserve its functionality and with LSB`s redundancy utilize, consequently through the network without getting attention from the attacker. In addition, the LSB technique they used to provide both capacity and carrier blindly which the main advantage of recovering without referencing the original DNA sequence.

H. I. Hussein et al. in [12] concentrated on expanding the security of the original Table Lookup Substitution Method (TLSM). The original TLSM was modified by including the letter transformation in the message to a binary organization based on an 8-bit binary base, and a base table was made to supplant every two bits in the binary reference DNA sequence with two indistinguishable bit letter entries. The outcomes in this paper demonstrate that this strategy is superior to the original in cracking probability and capacity.

G.Hamed et al. in [7] proposed a method which used steganography based on DNA and encryption, encryption confidential message then hiding based on DNA, also discussed how to extract and decrypt the confidential message. convert message to binary by using N-bits then convert to DNA using BCR then encryption using the DNA Playfair cipher size matrix is 8\*8 filled with 64 combinations of the codon in a random way. The strategy of hiding using the DNA conservative mutations allows the DNA base to be substituted by another DNA base to be able to hide two bits. As a result, a high capacity of hiding of data within the DNA and, low cracking probability.

S.Marwan et al. in [18] proposed an adjustment to the technique is the DNA-based data encryption and hiding using Playfair 4×4 cipher grid which makes largest hiding capacity. They also presented an analysis of comparisons between DNA-based Playfair, DNA-based Vigenere, DNA-based RSA and DNA-based AES ciphers. have been compared based on key and data size, security, speed, and hiding capacity.

Malathi et al in[10] in their study they use insertion algorithm based-DNA are modified to decrease the cracking probability of data. Also, they experiment with their algorithm on three measurements: Capacity, Payload, and BPN. After they applied their method showed it is better than the Substitution method, the Complementary method, and the Insertion method.

A. Khalifa et al. in [19] proposed a solution for an issue that was beforehand in another paper. the issue is the need to secretly send steganography -DNA and the reference sequence between the sender and the receiver and how to solve the problem. The method of hiding is applied in two steps: the first step is the hidden process in the reference DNA sequence by the substitution method. The second to embed the modified DNA sequence in the original reference sequence is the insertion technique used by randomly inserted.

P.Johri et al. in [4] make a survey on several data hiding methods Like audio, video, image, and text, to study increase data protection over the network. This paper compares cryptography and steganography depends on these forth features; security mechanism, the security of data, robustness, and attack. It also talks about digital watermarking; this method uses the signal to hide data. In the end, they show a decrease difference between the original cover-file and steganography file.

A. Ali et al. [20] survey many methods that combine cryptography and steganography in the same system, to support more reliable protection from hackers. The combination depends on the advantages of cryptography and steganography together. It shows a table of differences between them. Also explain steganography kinds like text, image, audio, and video. The benefits of this combination are latent in high memory space, security, and strength for data. It is the proposed method AES to the encrypted messages, then use Secure Hash Algorithm 2 (SHA-2) to hash the key. After that, use the LSB technique to embed the encrypted information. The result in video files cover better than other proposed kinds based on their high capacity.

K.Santoso et al. in [21] this paper exhibit sector-based DNA steganography method. They chose this method to achieve a high level of security, more data capacity and an increase error-handling rate. Their embedding process divided into two levels: coding region and non-coding region. The DNA properties summarized as Sector-based-embedding are arranging the number of sectors then embed it across coding sequence, False- start-codon search about neighboring watermarked nucleotide bases. Protein preservation to protect protein-coding regions, High Capacity can control embedding space in the sector, Security used two keys; message encryption key and embedding key, Error detection, and correction detect the error by special strategy of detection. The result of this experimentation is capacity higher than 1.0 BPN and increasing according to the increasing length of the sector. Finally, for message extraction, no need for the original host DNA just need for the encryption key and embedding key.

K.Kordov and B.Stoyanov in [22] are use LSB based on a Hitzl-Zele chaotic function. It depends on many steps the method works. And they use three software test programs to measure the manner of the output binary streams. First, one NIST includes 15 statistical tests, DIEHARD includes 19 statistical tests and ENT includes 6 statistical tests. Also, by using the pseudorandom bit generation algorithm scheme. The Proposed Pseudorandom Bit Generation Algorithm uses the key for space measuring and sensitivity key for measure data-sensitive from different secret keys. In conclusion, they find their algorithm has adequate key size, highly sensitive of the key and good performance.

R. Tank and Prof. V. Agrawal in [23] to enhance the data security they are use method of three levels. One level of encryption and two levels of steganography. The first level is encryption use DNA based RSA algorithm converts the insides of the secret file into binary form then encrypts it into a DNA sequence, after that store the DNA sequence in a text file. The second level is steganography use a randomized LSB substitution technique to hides the encrypted message inside the DNA sequence. The third level is also based on steganography hides the embedded DNA sequence inside the audio file using LSB encoding. Finally, the secret file has passed through the encryption stage and two stages of steganography to protect data from a third party.

Mrs. A. Sharma in. [24] They developed a message decomposing method based on DNA steganography to hiding message, and then to avoid the deficiency, solving problems of DNA steganography. They produce some procedures to accomplish it via, first step:  encrypt the massage by Vigenere cryptography algorithm then, the second step:  decompose the messages into two parts that will send publicly as microdot if it is not contaminated or assumed from an adversary, if it is not the composing method will be keeping repeat a previous step. This paper concluded that DNA molecule has potential in the multidisciplinary areas especially with the information science field both with another, also such as polymerase chain reaction (PCR) and DNA chip (microarray) requires researchers from different areas to work together in the new cooperative path.

E. Kusuma. et al in. [25] A Combination: they provide a method with high capacity, combining two techniques of security steganography and cryptography.  This method relay on two algorithms of encryption both together such as Arnold’s transformation (ACM) to shuffle the image’s pixels and RSA scheme to increase the complexity therapy the process it takes. Furthermore, the method applies the LSB technique to investigate steganography by planned each two-bit to one bit. In this study, the first step an image is encrypted by the RSA algorithm after that, apply Arnold’s transformation then, using 2-bit LSB steganography on the cipher pixel. The paper concluded that after applying PSNR and entropy measures to evaluate this method, the high degree of PSNR and best of 7.9948 entropy.

S. H. Lee in [26] Based on multilevel histogram shifting they produce a reversible hiding method to hide the information in DNA sequence which provides blind facilitate for recovery and detection from the original DNA sequence, prevent the mutations, keep the sequence length, increase the watermark capacity. This work starts with encoding the four-character nucleotide bases into integer values. Then they apply multilevel histogram shifting of both types noncircular (NHS), and circular (CHS). After that, by avoiding generate start/stop codon this study concluded exploit the potential of high capacity in DNA sequence storage with using (NHS) and (CHS) methods hide data to achieve the high capacity, the high degree of BPN more than the conventional method.

S. Marwan. et al in. [27] steps of the method, first the preprocessing algorithm converts the message into binary and then encrypts by any of the encryption techniques such as Vigenere or Playfair after that conversion to the DNA sequence using BCR. second, hide data by substitution algorithm at specific positions. The final output is the fake DNA sequence that is hidden in the secret message inside. Compare the method suggested in this paper with other methods and the result is that it is able to store twice the amount of data than the other and provide a high level of security.

A. Taha et al in [28], concern on information hiding in Arabic text. In their opinion, the Arabic language richness at its characteristics makes it a strong candidate for use it in hiding information. By improves the length of the secret message to prevent impact on their quality as much as possible. In conclusion they notice the suggested algorithm works better and gives higher capacity than peer algorithms.

S. Aljawarneh et al in [29], they advanced and designed a resource-efficient encryption algorithm system. This system defines a multi-level encryption model to enhance system efficiency and performance. This algorithm combined three processes: scrambling the mutation and crossover operations of Genetic Algorithm (GA). At the end the system results were very promising and very encouraging with the performance and the security.

S. Zaghbani and M. Bouhlel in [30] for data hiding, they proposed the frequency domain benefits schema and the YCbCr color space. They used inserting manipulates in the discrete cosine transform domain because it provides a good-quality image. After they use this algorithm clarified it was invisibility and effective in robustness rather than the insertion capability. This algorithm they use it had a problem in block gathering. For that, they decide will change the incorporation of these blocks to benefit from all. Finally, the result of the insertion capacity is larger and high value of invisibility property.

I​n this table 3 we summarized some of the reviewed DNA-based steganography. First column, the secret text type. The second column is the type of BCR used to convert binary message to DNA sequences. The third column is the encryption algorithm they use it. The fourth column is DNA based steganography algorithms. The last column is blind or not blind that is whether the embedded confidential data can be retrieved without having to original DNA sequence reference at the extraction stage.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Paper**  **Author** | **Secret Text Type** | **Used Binary Coding Rule** | **Encryption Algorithm** | **Data Hiding Methodology** | **Blind/**  **Not Blind** |
| G. Hamid et al | Any type of data. | N-bits binary coding rule. | Encryption used DNA and amino acids based Playfair cipher, 5\*5 matrix. | LSB substitution but after shuffling them. | Blind. |
| G.Hamed et al | Any type of data. | 4-bits binary coding rule. | Encryption used DNA and amino acids based Playfair cipher, 5\*5 matrix. | LSB substitution. | Blind |
| G.Hamed et al | Any type of data. | N-bits binary coding rule. | Encryption use DNA Playfair cipher, 8\*8 matrix. | LSB substitution but after shuffling them. | Blind, |
| S.Marwn et al | Any type of data. | N-bits binary coding rule. | Encryption used DNA Playfair cipher, 4\*4 matrix. | Substitution method. | Not blind. |
| A. Khalifa et al | Any type of data. | N-bits binary coding rule. | Encryption used DNA and amino acids based Playfair cipher. | Combine Substitution method and  Insertion method. | Blind. |
| S. Marwan. et al | Any type of data. | 2-bit binary coding rule. | Any of the cryptography  (the Vigenere or the Playfair). | Substitution method. | Not blind. |
| Mrs. A. Sharma | Text. | - | The Vigener cryptography. | Decomposing method. | Not blind |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Paper**  **Author** | **Secret Text Type** | **Used Binary Coding Rule** | **Encryption Algorithm** | **Data Hiding Methodology** | **Blind/**  **Not Blind** |
| M.Sajisha KS and Dr.S.Mathew | Any type of data. | 4-bits binary coding rule. | Encryption used DNA and swapping of amino acids-based AES. | LSB substitution and the hiding is done on the adjacent bases. | Blind. |
| A. Khalifa et al | Image. | - | Hyperelliptic Curve Cryptography. | Hide an image in another image by converting it into DNA | Not blind. |
| R. Tank and Prof. V. Agrawal | Text files. | 2-bit binary coding rule. | Encryption used DNA based RSA. | LSB substitution | Blind. |
| E. Kusuma. et al | Image | - | ACM and RSA Encryption. | LSB insertion | - |
| K.Santo et al | Any type of data | - | Use XOR cipher | sector-based DNA steganography using the non-coding region of DNA sequence | blind |
| Malathi et al | Any type of data. | 2-bit binary coding rule. | Use XOR cipher | Insertion method. | Blind. |
| H. I. Hussein et al |  | 8-bits binary coding rule. | No Encryption. | Table Lookup Substitution. | Not blind. |
| K.Kordv and B.Stoynov | Image. | - | No Encryption. | LSB by using random properties of chaotic pseudorandom bit generator. | Blind. |
| S. H. Lee | Image | N-bit binary coding rule. | No Encryption. | Revisable multilevel histogram shifting method | blind |

Table 3 Comparison of algorithms related to our work

## Conclusion

In this chapter, we presented the background and related work that is attached to our study. In our project, we compare three DNA-based steganography [6][10][11] in terms of capacity, payload, BPN, and cracking property. We select the algorithms that based DNA combined with encryption. The selected algorithms the substitution method and the insertion method and complementary method. In the next chapter, we will discuss more the algorithms we used in detail to be accomplishing this work.

# Chapter Three: Methodology

## Introduction

The previous chapter explained the related security concepts such as stenography (DNA based) and cryptography as well as outlined the related studies in the DNA based steganography algorithms.

This chapter presents an overview of the algorithms used. Moreover, it explains the solution formalizations and algorithms design. Also, it gives descriptions of the measurement units used to compare the selected algorithms.

The Chapter is organized as follows: Section 3.2 focuses on to formulization the problem and the algorithm approach precisely. Section 3.3 explains the solution design in clear steps of these algorithms and the measures are illustrated. The conclusion is included in Section 3.4.

## Theoretical formalization and algorithms design

In recent years there are many methods of DNA-based data hiding, we compare three of these methods to show the difference between them based on the specified measurements. In this section, for each method, we introduce the formalization and algorithms that use it.

All these methods apply steganography in two phases. The first phase is the encryption based on DNA where the sender and receiver agreed on some schemes such as n-bit coding, which transforms ASCII code into binary code and vice versa. Another scheme is BCR, which transforms DNA nucleotides A, C, G, and T into binary codes and vice versa. These schemes should be kept secret. Then they use one of the encryption methods such as Playfair to encrypt the message. The second phase including one of the steganography methods like insertion and substation. to hide the encrypted message on the DNA sequences.

### Algorithm 1: based on the substitution method [6]

In this paper, they use the Playfair method for encryption and the substitution method for steganography. Later in this section, we explained in detailed the algorithm on both sides (sender and receiver).

**Sender side:**

On the sender side as already indicated the encryption technique is applied before steganography as shown in Figure 6.

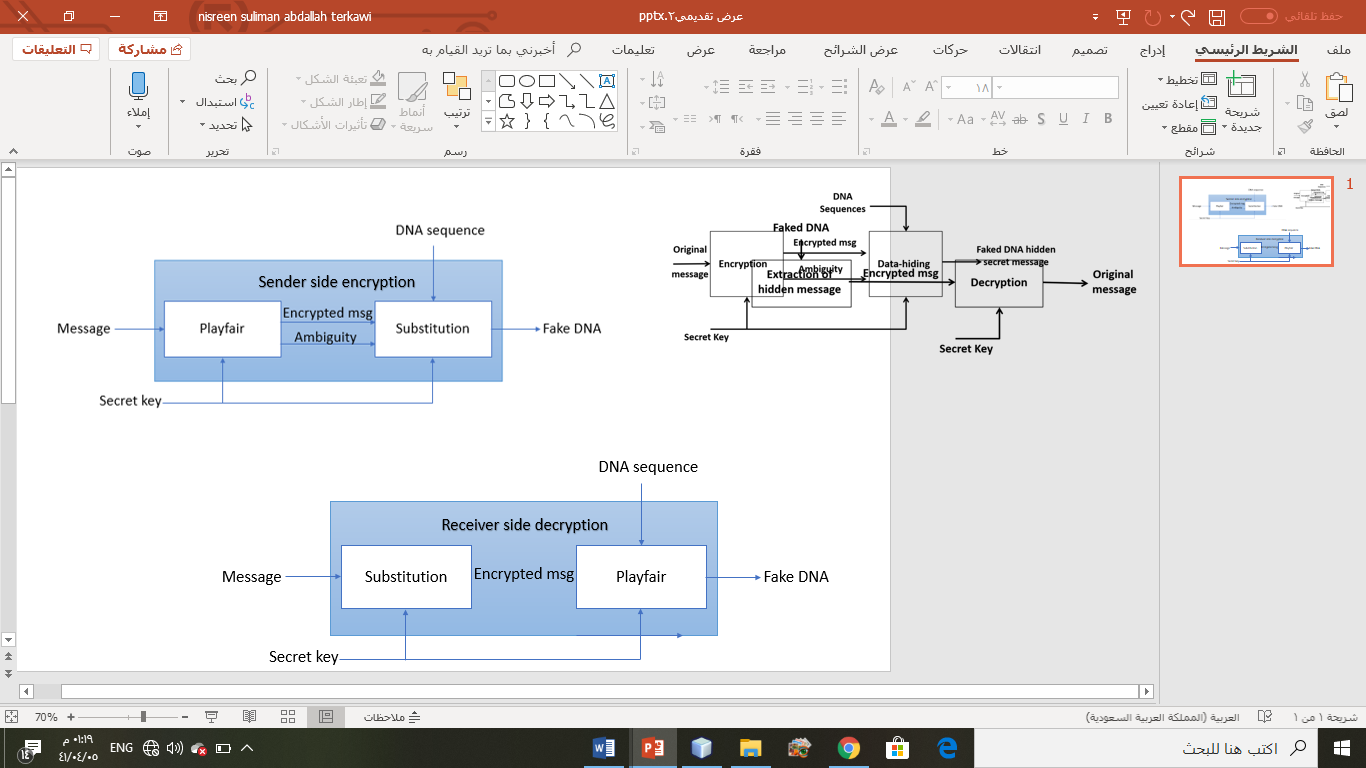


Figure 6 sender side flowchart

In the proposed algorithm, initially they take the secret message and secret key and processing them as follows: the message transforms to corresponding ASCII then to binary using 8-bits coding. Then the binary of secret message converts to DNA nucleotide using 4-bits BCR according to Table 3 which is maps each 4 bits binary message to 2-bits DNA nucleotides. Then the DNA of secret message is converted to amino acids according to Table 4 which is the standard universal table of the distribution of the alphabet with their corresponding codons [13]. While conversion keeping the ambiguity to retrieve the message at the decryption. Now apply Playfair using the secret key to encrypt the form of the amino acid of the secret message. After that, the cipher message is converted back to DNA by selecting the first codon corresponding to each amino acid. By this step, the encryption phase finished figure 7 shows the process flow of it.

صورة تحتوي على لقطة شاشة

تم إنشاء الوصف تلقائياً

Figure 7 Data Encryption Flowchart [6]

|  |  |  |  |
| --- | --- | --- | --- |
| **DNA Nucleotides** | **Binary Representation** | **DNA Nucleotides** | **Binary Representation** |
| AA | 0000 | GG | 1000 |
| AC | 0001 | GA | 1001 |
| AG | 0010 | GC | 1010 |
| AT | 0011 | GT | 1011 |
| CC | 0100 | TT | 1100 |
| CA | 0101 | TA | 1101 |
| CG | 0110 | TC | 1110 |
| CT | 0111 | TG | 1111 |

Table 4 4-bit BCR

|  |  |  |  |
| --- | --- | --- | --- |
| UUA, UUG | O | GCU, GCC, GCA, GCG | A |
| CCU, CCC, CCA, CCG | P | UAA, UGA, UAG | B |
| CAA, CAG | Q | UGU, UGC | C |
| CGU, CGC, CGA, CGG | R | GAU, GAC | D |
| UCU, UCC, UCA, UCG | S | GAA, GAG | E |
| ACU, ACC, ACA, ACG | T | UUU, UUC | F |
| AGA, AGG | U | GGU, GGC, GGA, GGG | G |
| GUU, GUC, GUA, GUG | V | GAU, GAC | H |
| UGG | W | AUU, AUC, AUA | I |
| AGU, AGC | X | AAA, AAG | K |
| UAU | Y | CUU, CUC, CUA, CUG | L |
| UAC | Z | AUG | M |
|  |  | AAU, AAC | N |

Table 5 Distribution for codons on English alphabet [31]

After encrypting the message, the steganography process starts using LSB substitution. The LSB method hides the secret message by substituted the least significant bit of each codon in the reference DNA sequences depends on the bit of the secret message. They take a reference DNA sequence S, ciphered DNA message, and ambiguity and processing them as follows: the ciphered DNA message converts back to binary representation using 4-bits BCR. Also, the ambiguity converts to binary. Since the maximum number of codons corresponding to amino acid is 4. So, it can represent in 2 bits as shown in Table 6. Now hide the binary ciphered DNA message bits and the binary ambiguity bits. Reference sequences are purine base (A&G) or pyrimidine base (C&T). To hiding data check if LSB codon is a purine base, it is substituted by (G) to encode 1 of the secret messages or (A) to encode 0. If the LSB codon is a pyrimidine base, it is substituted by (C) to encode 1 of the secret messages or (U) to encode 0. The innovation idea 3:1 ratio used to hide 3 bits of binary cipher message followed by 1 bit of binary ambiguity. This because the ratio of the length of the binary cipher message to the length of the binary ambiguity is 3:1. Figure 7 shows the process of steganography.

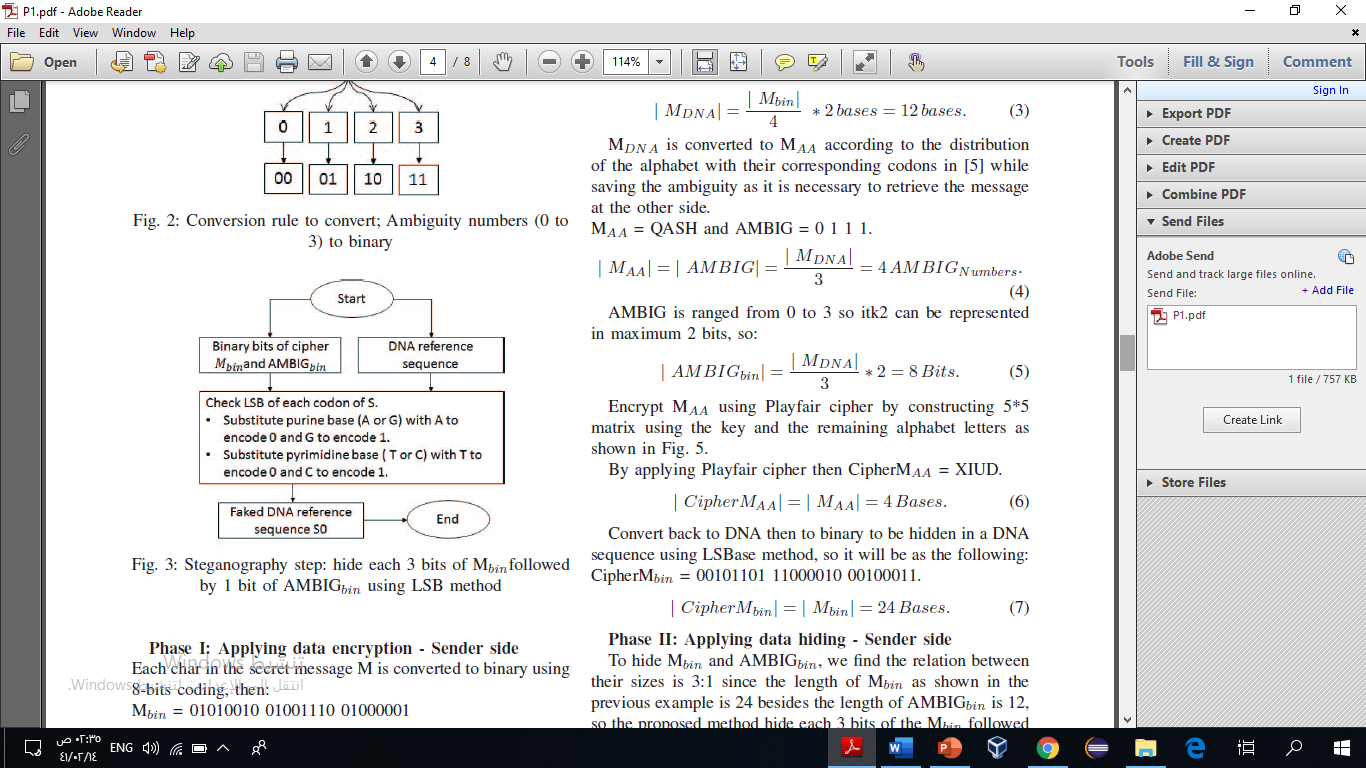


Figure 8 Steganography steps 6

|  |  |
| --- | --- |
| **Decimal** | **Binary representation** |
| 0 | 00 |
| 1 | 01 |
| 2 | 10 |
| 3 | 11 |

Table 6 5-Ambiguity numbers (0 to3) to binary

**Receiver side**

On the receiver side, they extract the ciphered message from the DNA sequence then decrypt the ciphered message as shown in Fig 9.

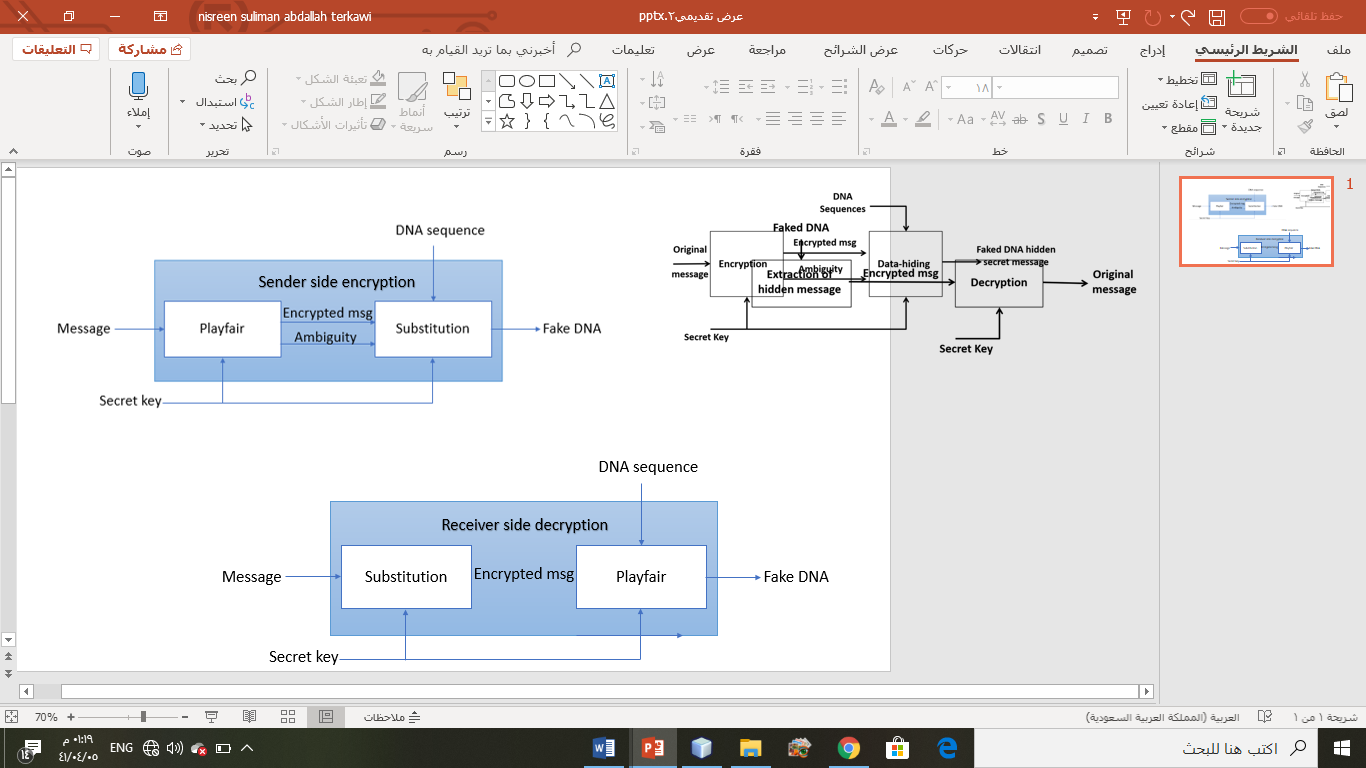


Figure 9 receiver side flowchart

The sender sends the faked sequence to the receiver. Then, the receiver extracts the data from the faked sequence. Then applies the Playfair to decrypt the message using the secret key.

In the proposed algorithm, the faked DNA reference sequence was taken, and the data extraction applied to extracts the data. Data extraction is inverse of the hiding algorithm. It is processed as follows: to retrieve the hidden bits of the secret message checks if the LSB of each codon is (A or T), then the hidden bit is '0'. if the LSB is (C or G), then the hidden bit was '1'. For every four bits extracted from LSB of codons, the first three bits added to the secret message and the last bit added to the ambiguity of the secret message till the secret message and ambiguity are completely extracted from the faked DNA reference sequence.

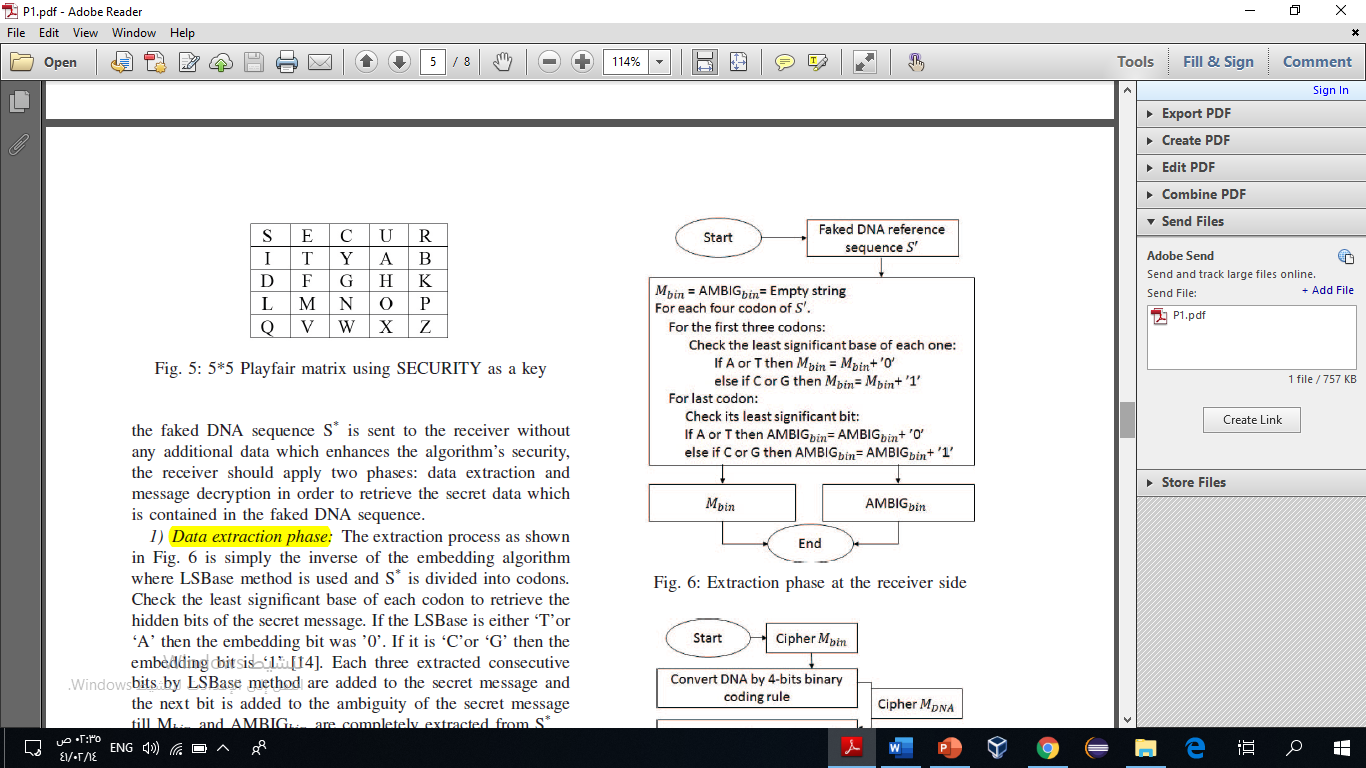


Figure 10 Steganography steps [6]

Then after extracting the data, the decryption is used to retrieve the original message. For this purpose, they take the binary cipher DNA message and binary ambiguity and process them as follows: the binary cipher DNA message converts to DNA using 4-bits BCR. Then the ciphered DNA converts to amino acids. Now apply Playfair using the secret key to decrypt the ciphered amino acids. Convert binary ambiguity to decimal as shown in Table BB. The output of Playfair form amino acids of the secret message converts to corresponding DNA using the associated ambiguity. Finally, the retrieved DNA of the secret message convert to binary using 4-bits BCR then converts to ASCII to get the corresponding original secret message.

صورة تحتوي على لقطة شاشة

تم إنشاء الوصف تلقائياً

Figure 11 Steganography steps [6]

### Algorithm 2: based on the insertion method [10]

In this paper, they use the XOR operation for encryption and the insertion method for steganography. Later in this section, we explained in detailed the algorithm on both sides (sender and receiver).

**Sender side:**

On the sender side, as we already indicated the encryption technique is applied before steganography as shown in Figure 12.

In this algorithm, initially they take a message and they use two keys processed as follows: the secret message is converted into ASCII value, then it is converted into a binary sequence. After that, the binary sequence is split into 8-bit binary segments. The first key-value generates randomly for used to XOR the message. The first key value is converted into an 8-bits binary sequence then it is XORed with the first 8 bits of the binary value of the message. The result value of the XOR is again XORed with the next 8 bits of the message and so on until the end of the message. The final XOR result concatenated to form a single binary sequence of the cipher. The DNA sequence taken from the data set is divided into segments using the randomly generated key value (the second key). The value of the second key is kept having the minimum length of the DNA sequence to hide the secret message. Using the value of the second key the binary converted DNA is split into segments and the binary bits of ciphers are inserted one by one at the beginning of each segment. Using the dictionary rule the resulting binary sequence is converted into DNA sequence according to table 2. Use the insertion algorithm for the encoding of cipher inside the DNA to obtain a fake DNA sequence.

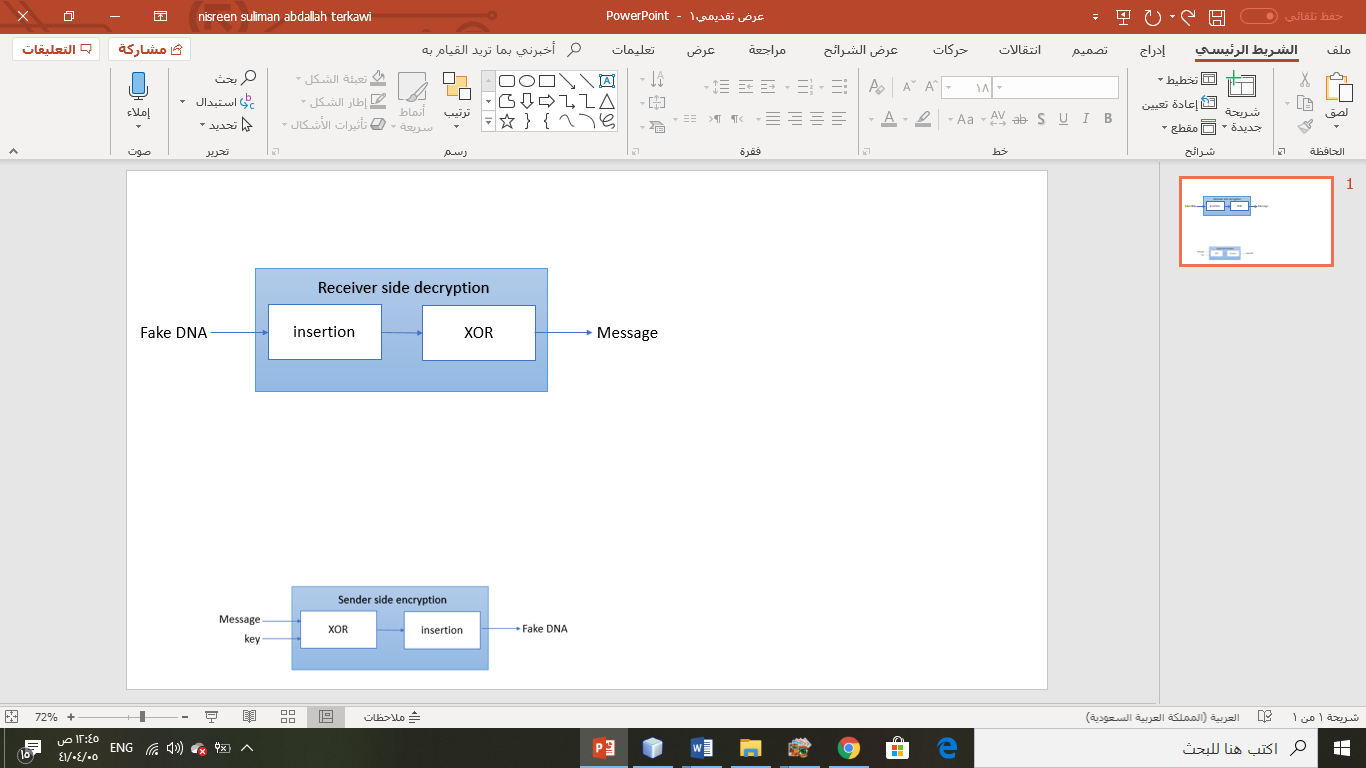
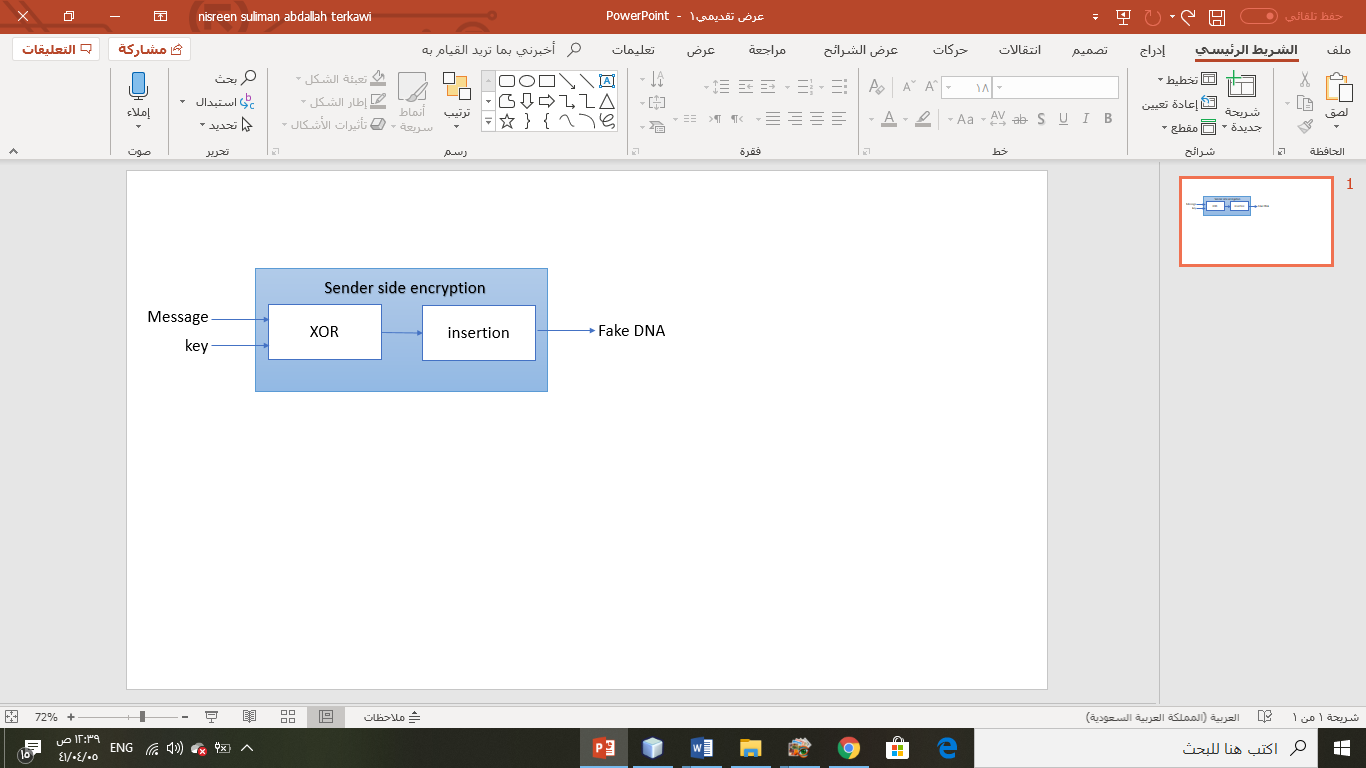
**Receiver side**

On the receiver side, they extract the ciphered message from the DNA sequence then decrypt the ciphered message as shown in Fig 12.

By complementing the sender side steps of encoding, the decoding of the fake DNA is done. In the end, get the string message by transformed this binary value obtained into its corresponding ASCII values.

In the decoding part, encoding DNA sequence to binary form, then divide it into multiple parts of the same length to get a new DNA sequence. From a previous division operation get the binary value of the message, then Convert the binary message and the binary sequence to DNA sequence by coding rule according to Table 5. After that, apply the insertion method on the previous binary string to get a new binary string, then divide it into 8-bit pieces. The next step is to apply the XOR operation to get the message in binary form, then convert it to its ASCII equivalent value to obtain the message.

Figure 12 Alogrithm insersion processing



### Algorithm 3: based on generic complementary substitution method [11]

In this paper they rely on two layers of techniques to provide more higher security. The two's techniques they followed are encryption layer by Playfair algorithm and steganography layer by complementary Substitution technique. They applied these two's layers for both ends communication sender and receiver sides when sending a message happen. They accomplish their work with two scenarios in both sides illustrated below.

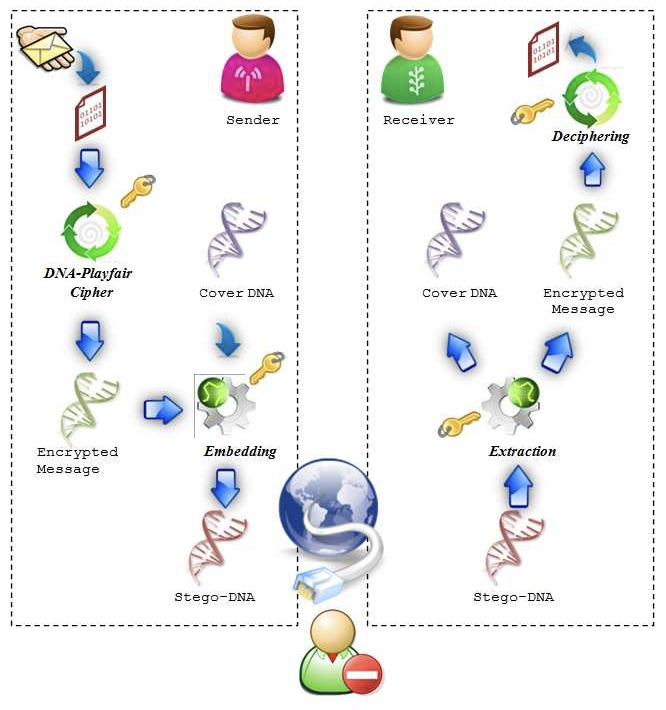


Figure 13 The proposed scheme followed [11]

In general, as it is clear in figure 13 they divided their work into two main procedures, first one is embedding procedure at the sender side and the second one is extraction procedure at the receiver side.

**Sender side:**

In this side the algorithm proposed should has a message, secret key of string and sequence of DNA reference cover S. The scheme starts with encrypt secret message before hide it as in figure 13. They accomplish the hole processes in this side by start making the secret message that we will send it to the receiver adapted and that done by apply the first step which is convert secret message to the ASCII code. Then complete the process depend on the first step which is message converted to ASCII form. The algorithm in this paper establish by transform message in ASCII code to binary in 8 bits. Now the algorithm converts the sequence of binary form to the sequence of DNA nucleotides by grouping of 2bits from binary sequence as shown in figure 14, then construct the Amino acids sequence by chain sequence of codons which is grouped three bases from DNA sequence according in figure 15, also find the ambiguity sequence that belong to each codon position to avoid codon redundancy in decryption phase.

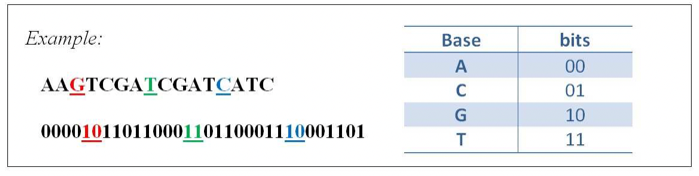


Figure 14 A digital coding of DNA bases [11]

Now by using secret key of string, apply Playfair ciphering algorithm to encrypt the sequence of Amino acids. After encryption convert back the sequence of encrypted Amino acids to the DNA sequence by using first codon for each amino acid character and append it with the ambiguity character for each one.

Now start the second layer which is hide the encrypted DNA sequence by embedding to the sequence of DNA reference cover. The hide phase will be done by finding palindrome word from DNA reference then added it to the end of DNA message with T base for both side. After this step, start apply substitution phase to generate DNA sequence S'. This phase done by replace some base in cover DNA with their generic base paring figure 15 according to how many times complement it depend on the rule shown in figure 16.

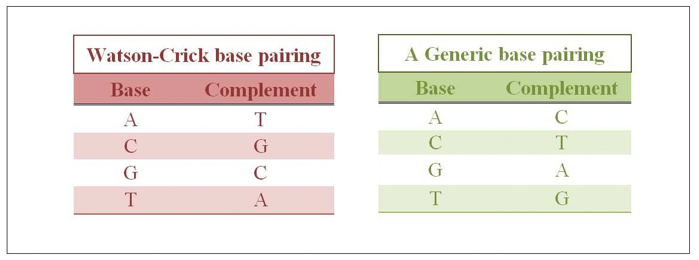


Figure 15 Complementary rules for the nucleotide bases of DNA [11]

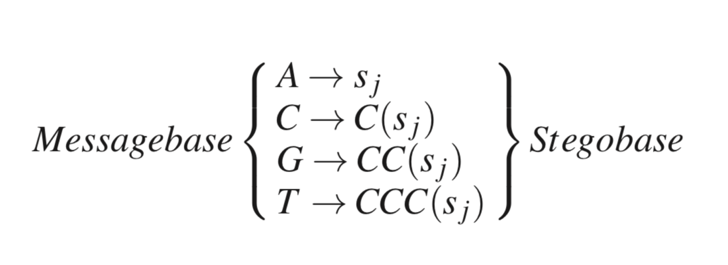


Figure 16 message base and after steganography base [11]

After applying generic complementary substitution method, the last phase was insertion to get the fake DNA sequence S''. the fake DNA sequence S'' come from padding an arrangement segments of DNA reference cover S and DNA sequence S' depend on a secret key.

**Receiver side:**

In this side they get the original message by revise the steps that done in sender side as shown in figre13. The receiver side get a fake DNA sequence S'' from sender side. This side considers dealing with the second main procedure in this scheme which is called an extraction procedure. To make this process done it will start by adapt the fake DNA sequence S'' to decipher step for find the original message. The algorithm proposed to adapt the DNA sequence S'' by extract the two sequences of it : DNA reference cover S and DNA sequence S' by divided S'' into two segments according to constant different tall generated from key as a seed the DNA sequence S' appeared . After that by apply reveres replacing some base in DNA sequence S' with their generic base paring figure 15 according to how many times complement it depend on the rule shown in figure 16. After this step the DNA message sequence appeared. Then this time for parse this sequence to separated it to two DNA sequence: DNA ambiguity sequence and DNA message sequence. now convert DNA message sequence to Amino acids sequence to apply the DNA Playfair deciphering algorithm. After that convert the decrypted sequence to the DNA sequence and append it with ambiguity sequence. Then after concatenating these two sequences together, convert them into binary form then by grouping of 8 bits to convert them to ASCII code then to real original message.

## Experimental Design

### The algorithm based on the substitution method [6]

|  |  |
| --- | --- |
| Algorithm 1-1 Sender side algorithm | |
| Input: | Secret message M, secret key K, and DNA reference sequence S. |
| Output: | Faked DNA reference sequence S` hidden the secret message and ambiguity AMBIG. |
| Step 1. | Transform M to binary Mbin using 8-bit |
| Step 2. | Transform Mbin to DNA MDNA using 4-bit BCR |
| Step 3. | Transform MDNA to amino acid MAA and keep AMBIG |
| Step 4. | Apply Playfair to encrypt MAA using K Cipher MAA |
| Step 5. | Transform Cipher MAA to DNA Cipher MDNA |
| Step 6. | Transform Cipher MDNA to binary Cipher Mbin |
| Step 7. | Transform AMBIG to binary AMBIGbin |
| Step 8. | Initialize S` to be a copy of S |
| Step 9. | For each codon of S' |
|  | For the first three codons check LSB of each one. |
|  | Substitute purine base (A or G) with A to encode 0 and G to encode 1. |
|  | Substitute pyrimidine base (T or C) with T to encode 0 and C to encode 1. |
|  | For the last codon check LSB. |
|  | Substitute purine base (A or G) with A to encode 0 and G to encode 1. |
|  | Substitute pyrimidine base (T or C) with T to encode 0 and C to encode 1. |

|  |  |
| --- | --- |
| Algorithm 1-2 Receiver side algorithm | |
| Input: | Faked DNA sequence S`. |
| Output: | The secret message M. |
| Step 1. | Cipher Mbin= AMBIGbin=Empty |
| Step 2. | For each codon of S' |
|  | For the first three codons check LSB of each one. |
|  | If (A or T) add to Cipher Mbin bit '0'. |
|  | If (G or C) add to Cipher Mbin bit '1'. |
|  | For the last codon check LSB. |
|  | If (A or T) add to AMBIGbin bit '0'. |
|  | If (G or C) add to AMBIGbin bit '1'. |
| Step 3. | Transform Cipher Mbin to DNA Cipher MDNA using 4-bit BCR. |
| Step 4. | Transform Cipher MDNA to Amino acid Cipher MAA uses zeros AMBIG. |
| Step 5. | Decrypt Cipher MAA Using K MAA. |
| Step 6. | Transform AMBIGbin to decimal AMBIG. |
| Step 7. | Transform MAA to DNA MDNA use AMBIG. |
| Step 8. | Transform MDNA to binary Mbin using 8-bit. |
| Step 9. | Transform Mbin to ASCII then to plaintext M. |

### The algorithm based on the insertion method [10]

|  |  |
| --- | --- |
| Algorithm 2-1 Sender side of algorithm 2 | |
| Input: | Secret message M, sample DNA sequence S. |
| Output: | Faked DNA reference sequence S` hidden the secret message. |
| Step 1. | By random generate k1 8-bit binary string - it can be in the range of 0 to 255- which is to be used as a cipher ‘s’. |
| Step 2. | Split the message ‘M’ onto characters M= {m1, m2, m3…mn} and each element is converted to its 8-bit binary equivalent based upon ASCII standards. |
| Step 3. | The last character of set M is XORed with s. |
| Step 4. | The result of the XORed with the element preceding the last one in the set M and repeated until all the elements are converted and stored in ‘A’. are converted and stored in ‘A’. |
| Step 5. | The binary sequence in A is converted to the protein (Ribonucleic acid) sequence. |
| Step 6. | A sample DNA sequence ‘S’ is taken and along with a randomly generated number n (k2) less than the size of M. |
| Step 7. | Break S down into n divisions such that size of each division sums up to the size of S. |
| Step 8. | Similarly, break down the generated DNA sequence A and insert each division of A onto the divisions of S using the insertion algorithm. |
| Step 9. | Generate the binary form of the sample DNA sequence S by using the coding methods. |
| Step 10. | Generate a random number p so that we get to know the number of splits of the message M and sequence S. |
| Step 11. | A series of random numbers (s1, s2, s3, ..., sP) and (r1, r2, r3, ..., rP) would be generated such that the sum of all the random numbers would be equal to the size of the sequence and the message respectively for their series. |
| Step 12. | Now combine the corresponding splits that are s1 and r1 and then s2 and r2 and so on. Now you would get a new binary string. |
| Step 13. | Now convert the binary string back to DNA form by the coding rules to get the new DNA sequence S` with the message encrypted with it. |

|  |  |
| --- | --- |
| Algorithm 2-2 Receiver side algorithm of method 2 | |
| Input: | Faked DNA sequence S`. |
| Output: | The secret message M. |
| Step 1. | Convert the encoded DNA sequence S` into binary form. |
| Step 2. | Divide S’ sequence into parts of length s1+r1, s2+r2… sP+rP. |
| Step 3. | Now from all the small strings obtained from step 2 extract the first r(i) bits. |
| Step 4. | Combine all the remaining s(i) strings to get the DNA sequence. |
| Step 5. | Join all the extracted bits in step 3 to form the message in binary. |
| Step 6. | Convert the binary message and sequence to ATGC form by using the coding rule. |
| Step 7. | Get the binary equivalent from the insertion decryption output and store in M |
| Step 8. | Divide M into pieces of 8 bits each {m1, m2, m3….mn-2, mn-1, mn} |
| Step 9. | For i=n: 2  A= (XOR mn-1 with mn) +A |
| Step 10. | A= (XOR m1 with key) +A |
| Step 11. | This binary is converted to its ASCII equivalent to obtain the message. |

### The algorithm based on the complementary generic substitution method

|  |  |
| --- | --- |
| Algorithm 1 sender side | |
| Input: | S: A reference DNA sequence, as a cover media |
| Output: | Msg: a secret binary message  Key: a secret Key word  S'': A faked DNA sequence embedded with the secret message |
| Step 1. | By using binary code, convert Msg into DNA sequence S msg |
| Step 2. | Concert the sequence from step 1 to Amino acids sequence S Amino, find Ambiguity sequence S ambig |
| Step 3. | Encrypt S Amino into Senc by using key with DNA Playfair Ciphering |
| Step 4. | Concatenate Senc and Sambig into Smsg |
| Step 5. | Start the substitution phase |
| Step 5.1. | Find the palindrome W in S |
| Step 5.2. | Pad it with base T for both side of Wand add it to end of S msg |
| Step 5.3. | Declare n = tall of S and m = tall of S msg |
| Step 5.4.  Step 5.5. | Generate permutation of n by using the key as a seed ( p1, p2, …pn)  Create S' be copy of S |
| Step 5.6. | Create i to 1 |
| Step 5.7. | For j =1 to m to do the change with complement on S' bases depend on the  Complement rule |
| Step 5.8.  Step 6.  Step 6.1.    Step 6.2.  Step 6.3. | Return S' after changing  Start the insertion phase  Declare r , k derived values of key  Generate permutation of r,k ( r1, r2, ……), (k1, k2,….. )by using r, k as a seed  Defined t which the smallest integer number  divide S' into segments ( s'1, s'2, .. s't-1) with length permutation of r  divide S into segments ( s1, s2, .. st-1) with length permutation of k  initialize S'' to be an empty sequence  for i-1 to t-1 to appends the corresponding segment from S and S' into S''  return S" |

|  |  |
| --- | --- |
| Algorithm 2 receiver side | |
| Input: | S'' : A faked DNA sequence, embedded with secret message |
| Output: | Key: a secret Key word  Msg: the secret message embedded |
| Step 1. | The recovery phase |
| Step 1.1  Step 1.2  Step 1.3  Step 1.4    Step 1.5  Step 1.6  Step 1.7  Step 1.8    Step2.  Step2.1  Step2.2  Step2.3  Step2.4  Step2.5  Step2.6  Step3.  Step3.1.  Step3.2.  Step3.3.  Step3.4.  Step3.5.  Step4. | Declare r , k derived values of key  Generate permutation of r,k ( r1, r2, ……), (k1, k2,….. )by using r, k as a seed  Defined t which the smallest integer number  Divided S'' into 2t segments (s''1, s''2, s''3,….., s''t-1) with length of the pervious  generation of r( r1, r2, ……),and k (k1, k2,…..)  Initialize S' to be an empty sequence  Initialize S to be an empty sequence  for i=1 to t-1  Append the segment from S'' ri to S'  Append the segment from S'' ki to S  The Message recovery phase  Declare n = tall of S  Find the palindrome W in S  Generate permutation of n by using the key as a seed ( p1, p2, …pn)  Initialize Smsg to be an empty sequence  for i=1 to n  by reverse to the complementary rule append the bases from S' to Try to find this pattern TWT exists in Smsg then exit the loop  Deciphering phase  Parse the Smsg to find the Sambig and Senc  Convert the Senc into SAmino  By using the key decrypt the SAmino into Sdec with DNA Playfair deciphering  By using Sambig convert Sdec back to DNA sequence  Convert DNA sequence to binary representation to get the Msg  Return **Msg** |

### Comparison measurements

The algorithms are evaluated by measuring four parameters which are: capacity, payload, BPN, and cracking property.

These parameters are computed for checking the security and integrity of the concealed message within the reference DNA sequence. The results are experiments on different DNA reference sequence.

Cracking probability is the most important measurement that indicates the security degree of an algorithm. In terms of security, to crack an algorithm, we define some fundamental factors for each algorithm. The effective factors of the LSB substitution algorithm [6] are the DNA reference sequence, Binary coding rule, and the least significant base substitution rule. And the effective factors of insertion algorithm [10] are DNA reference sequence, Binary coding rule, the size of the message and prefix DNA, The message and DNA are segmented using the randomly generated key values, the probability of guessing segmentation of DNA and the XOR operation is performed for encoding the data inside the DNA sequence and the probability of XOR combination. And the effective factors of the complimentary generic substitution algorithm [11] are the random number generator and the two seeds used in the insertion phase, the complementary rule, the binary coding rule, and the Playfair technique.

The calculation of these measurements for each algorithm done as bellow in table 6:

In the table 10, there are shortcuts which are S: DNA reference sequence, M: message, and A: ambiguity.

|  |  |  |  |
| --- | --- | --- | --- |
| Measures | Algorithm based on substitution [6] | Algorithm based on insertion [10] | Algorithm based on complementary substitution [11] |
| Capacity |  |  |  |
| Payload |  |  |  |
| BPN |  |  |  |

Table 7 laws of our algorithms

## Conclusion

In this chapter, we explained the formalization and discussed the algorithms for all of three papers algorithm [6] [10] [11] and the measurements to evaluate the performance. Finally, we described the general design of the project. In the next chapter, we present the experiments of the project and the results discussion as well as the implementation details.

# Chapter Four: Results

## Introduction

The previous chapter discussed some of the topics related to solution formalization, algorithms design. Also, it presented an overview of the measurements to evaluate three algorithms.

This Chapter presents the result of all algorithms [6][10][11] based on selected measurements. Also, illustrate the experiments of the project.

The result of the project is presented in Section 4.2 the project experiments are presented in Section 4.3. Section 4.4 presents the conclusion of this Chapter.

## Project Results

The experiment of all three algorithms has been done on the following input: a secret message M of size 20-kilo bytes containing letters, numbers and special characters. And the secret key is ’SECURITY’. Also, we use eight different DNA reference sequence S from the NCBI database to measure each of capacity, payload, and BPN. The results illustrated in table 7, table 8, and table 9.

The result of the algorithm [6] based on substitution Shown in Table 7.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DNA reference | Number of nucleotides | Capacity | Payload | BPN  (M+A)/C |
| AC153526 | 200117 | 61197 | 0 | 0.03331 |
| AC166252 | 149814 | 59576 | 0 | 0.04449 |
| AC167221 | 204841 | 62642 | 0 | 0.03254 |
| AC168901 | 191136 | 58451 | 0 | 0.03487 |
| CS236146 | 118777 | 36323 | 0 | 0.05612 |
| JX978467 | 9270 | 2834 | 0 | 0.71916 |
| NC\_021114 | 8870 | 2712 | 0 | 0.75159 |
| NC\_021116 | 8958 | 2739 | 0 | 0.74421 |

Table 8 Results LSB substitution of hiding message into different DNA sequences

The result of the algorithm [10] based on insertion Shown in Table 8.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DNA reference | Number of nucleotides | Capacity | Payload | BPN |
| AC153526 | 200117 | 204626.0 | 500.0 | 0.00488 |
| AC166252 | 149814 | 154384.0 | 500.0 | 0.00647 |
| AC167221 | 204841 | 209349.0 | 500.0 | 0.00477 |
| AC168901 | 191136 | 195706.0 | 500.0 | 0.00510 |
| CS236146 | 118777 | 123278.0 | 500.0 | 0.00811 |
| JX978467 | 9270 | 13770.0 | 500.0 | 0.07262 |
| NC\_021114 | 8870 | 13372.0 | 500.0 | 0.07478 |
| NC\_021116 | 8958 | 13459.0 | 500.0 | 0.07429 |

Table 9 Results insertion of hiding message into different DNA sequences

The result of the algorithm [11] based on complementary generic insertion Shown in Table 9.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DNA reference | Number of nucleotides | Capacity | Payload | BPN |
| AC153526 | 200117 | 300175.5 | 1.5 | 0.22209 |
| AC166252 | 149814 | 224721 | 1.5 | 0.029666 |
| AC167221 | 204841 | 307261.5 | 1.5 | 0.021697 |
| AC168901 | 191136 | 286704 | 1.5 | 0.023252 |
| CS236146 | 118777 | 178165.5 | 1.5 | 0.0374183 |
| JX978467 | 9270 | 13905 | 1.5 | 0.479443 |
| NC\_021114 | 8870 | 13305 | 1.5 | 0.501064 |
| NC\_021116 | 8958 | 13437 | 1.5 | 0.4961424 |

Table 10 Results complementary generic insertion of hiding message into different DNA sequences

The result of cracking property for each algorithm is shown in Table 10

|  |  |
| --- | --- |
| Algorithms | Cracking property |
| Based on substitution [6] |  |
| Based on insertion [10] |  |
| Based on complementary generic substitution [11] |  |

Table 11 Cracking property comparison between our algorithms

## Experimental Results

### Hiding Message:

In this section, we view an example of a hiding message for all of the selected algorithms. The first input is the secret message = RNA. The second input is Keyword = SECURITY. These two inputs are constant for all algorithms.

The fig 13 shows the result in detail of the hiding message for the first algorithm which based on LSB substitution.

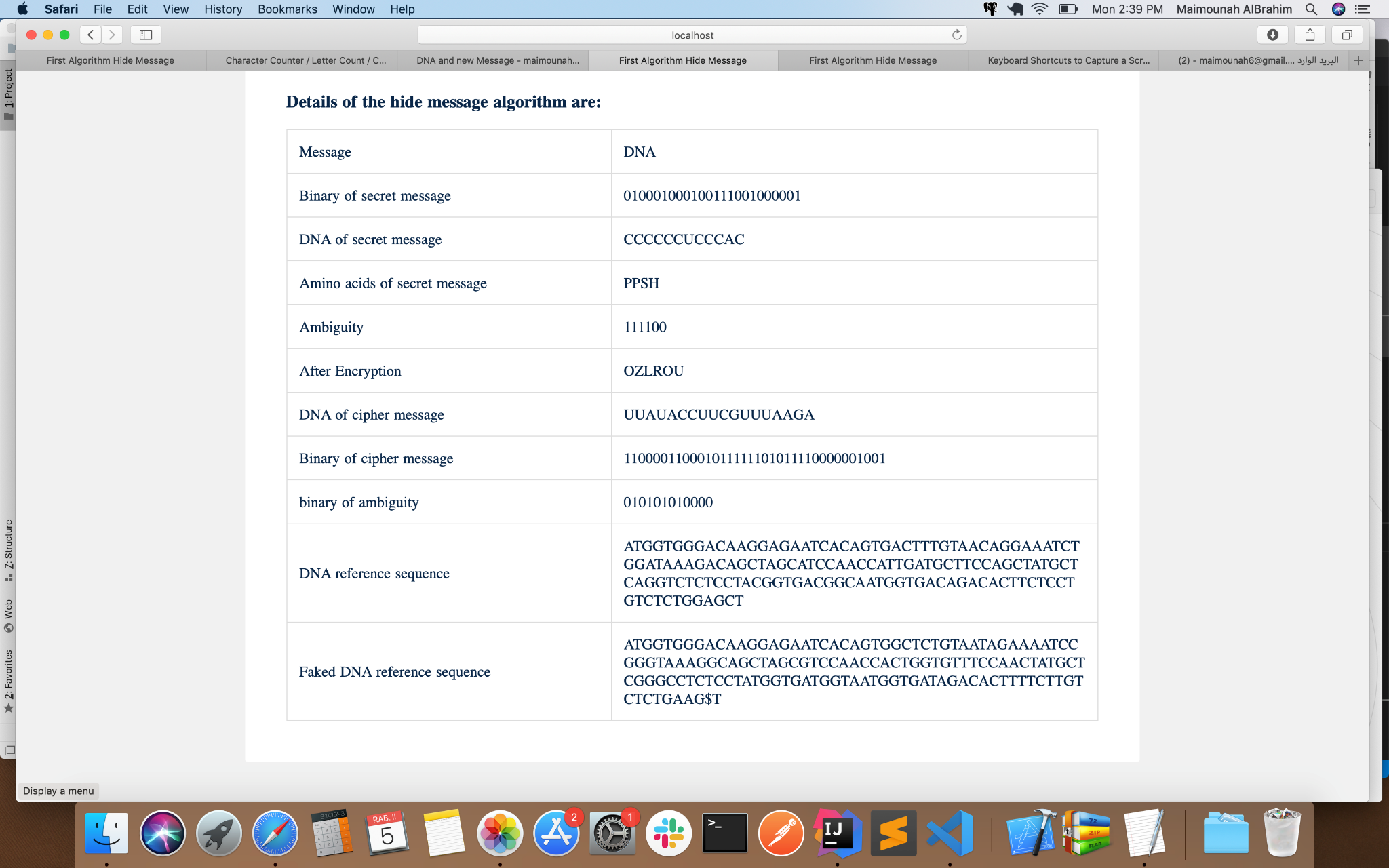


Figure 17 Example hiding message for Algorithm 1 based on LSB substitution

The fig 14 shows the result in detail of the hiding message for the second algorithm which based on insertion.



Figure 18 Example hiding message for Algorithm 2 based on insertion

Finally, the fig 15 shows the result in detail of hiding message for the third algorithm which based on componentry generic substitution

Figure 19 Example hiding message for Algorithm 3 componentry generic substitution

### Extracting Message

Here, we view an example of extracting the original message from the faked DNA already produced in the hiding message process. Also, we use the same keyword which is (SECURITY). Shows the result in detail.

The figure 16 shows the result in detail of the extracting message for the first algorithm which based on LSB substitution.

صورة تحتوي على لقطة شاشة

تم إنشاء الوصف تلقائياً

Figure 20 Example extract message for Algorithm 1 based on LSB substitution

The figure 16 shows the result in detail of the extracting message for the second algorithm which based on insertion.



Figure 21 Example extract message for Algorithm 2 based on insertion

Finally, the figure 17 shows the result in detail of extracting message for the third algorithm which based on complementary generic substitution**.**



Figure 22 Example extract message for Algorithm 3 complementary generic substitution

## Conclusion

In this chapter, we presented experimental results based on capacity, payload, BPN, and cracking property for all selected algorithms. In addition, we view examples of the outputs of these algorithms.

The next chapter the discussion of the results of each measurement.

# Chapter Five: Discussion

## Introduction

The previous chapter presented project experiments results measured by capacity, payload, BPN, and cracking property.

This chapter discusses the results of the measurements for each algorithm. And summarize and conclude the final evaluation of them.

## Discussion of the Results

The subsections discuss all measurements (capacity, Payload, PBN and cracking property) To compare the performance of the algorithms

### Discussion of capacity measure

All our algorithm is affected by the length of the DNA sequence used. As the DNA sequences length increases, its hiding capacity increases and visa verse as shown in three tables [] [] [].

The lowest capacity is the LSB substitution algorithm [6] because it is used only to hide within it the least significant base of all three bases. This algorithm can hide four bits per base. As shown in Table 8 the capacity of this algorithm ranges from [62642 – 2712].

Then the insertion method [10] divides the DNS sequence by using a random number to insert one bit from the message inside each division. Therefore, it has the second-lowest capacity. As shown in Table 9 the capacity of this algorithm ranges from [209349 – 13372]

Finally, the complimentary generic substitution algorithm [11] can hide one message base per base which is the message base is the equal two bits message for that it has the highest capacity. As shown in Table 10 the capacity of this algorithm ranges from [307261 – 13305]

### Discussion of payload measure

The less Payload, the better it was to avoid drawing attention to it. Therefore, the best algorithm is LSB substitution [6] because Payload is zero, which means that the length of the fake DNA reference sequence is not expanded after the message bits in it are hidden. This is achieved as a result of hiding the secret data by substituting the nucleotides.

Regarding the complimentary generic substitution algorithm [11] can hide one message base within one base of the DNA sequence. The message base is equal to two bits in a binary message. The message is an actual message and ambiguity. According to the algorithm, only 3/4 of these bases represent the actual message bits, and the remaining 1/4 should be reserved for ambiguity rules. Therefore, as shown in Table 10 Payload is equal to 1.5.

Finally, the insertion algorithm [10], the Payload of this algorithm depends on the message length, so its Payload is variable. The Payload equals the message divided by two, which attracts attention to it. As shown in Table 9 its equal 500

### Discussion of BPN measure

LSB substitution Algorithm [6], BPN is within [0.03254 – 0.74421] as shown in Table 8. This algorithm is acceptable embedding capacity which is distributed on both the message and ambiguity bits which results in increasing the total number of nucleotides required for hiding the message bits only. In complementary generic substitution [11], BPN is within [0.021697-0.49614] as shown in Table 10. Because this algorithm can be hidden within capacity both the message and ambiguity bits.

In the insertion method, the sequence divides by a specific random number then the message inserts across each division one by one bit, as shown in Table 9 the BPN between [0.00477 – 0.07478]

The more bits are hidden within a base rule, the better because it contains more messages. As we mentioned the highest is the LSB substitution algorithm.

### Discussion of cracking properties measure

The possibility for the intruder to crack the fake DNA to extract the hidden secret depends on the factors. For that, the insertion is the lowest and the complex than others. Then the complementary generic substitution is the second lowest to crack the fake DNA. Finally, is highest is LSB substitution.

We have concluded from a comparison of algorithms that the LSB substitution algorithm has the disadvantages of being less capacity. The advantages of this algorithm are Payload, so it does not attract attention and it's the largest amount of BPN.

The complementary generic substitution has disadvantages that are the Payload which is worst compared to the LSB substitution algorithm and BPN is not high. The advantage of this algorithm is its highest capacity.

The insertion algorithm contains disadvantages that the Payload depends on the message. Therefore, the longer the message, the increase the Payload. The other drawback is BPN is the lowest. The advantage is high capacity.

### Difficulties

In the implementation of the algorithms we encounter the following problems:

* In [6][11] when converting from DNA to amino acids, every 3 nucleotides are converted to one-letter amino acid. If the message is not a multiple of the three, it may lose the last letter of the original message.
* When using the Playfair method, it is possible to increase the number of characters from the actual message, as this can cause damage. Especially in LSB substitution paper, Method 3: 1, this method requires hiding three from the actual letter and one from the ambiguity. This problem is solved by increasing the zeros in ambiguity with the same increase in letters.
* In the implementation of the second method [10] at step 10 as illustrated in section 3.3.2 the boundary of the p random generation value is not specified, for that, we assumed the boundary between 1 and 5 to be suitable to the long and short message length.

## Conclusion

In this chapter, we discuss and summarize the results of the measurements in detail. Also, we present some of the difficulties we face while implementation.

The next chapter summarizes the overall work done on this project and discusses the possible future works.

# 

# Chapter Six: Conclusion and Future Work

## 6.1. Introduction

In the previous chapter, we show the discussion of the results of the algorithms used in this project and their measurements. In this chapter, we clarified the chapter body in section 6.2 and conclusion in section 6.3.

## 6.2. Chapter Body

### 6.2.1. Contributions

This project contributed to help who want to protect their information by knowing the features of each algorithm to choose the suitable for their requirements. Also, it helps the researcher to build on our study to get more secure properties. Because in our project we provide suitable information as:

* Measuring the basic features for the DNA-based steganography.
* Implementing three sample algorithms for each main steganography method.
* Compare the algorithms in terms of the selected features.

### 6.2.2. Future Work

In the future, we are thinking about extending this project by creating a new steganography method better than and stronger than others, by combining more than one algorithm, to obtain more secure data at exchanging it. Or, we can improve and overcome some of the limitations we found during our implementation of the selected algorithm. Also, we can examine the selected algorithm with a different encryption algorithm and study the effect of that on the new proposed algorithm. In addition, we are thinking about studying more steganography algorithms and make comparisons among them to clarify their advantages and their disadvantages then combine different algorithms to propose a new better algorithm that inherent the strength of each algorithm.

## 6.3. Conclusion

Nowadays the security field is trending, over time it becomes increasingly important because also the attackers and the intruders over time becomes increasingly dangerous. For that in our project, we focused on the steganography and we study a comparison of three ways of them. For that in our project, we focused on the steganography methods and we made a comparison of three ways of them by measuring the capacity, payload, and BPN. The three algorithms are by using substitution, by using insertion, and the last by using complementary generic substitution. As a result, it turns out that the algorithms we noticed that each algorithm has advantages and disadvantages based on measurements.

**References**



[1] W. Stallings and L. Brown, *Computer security: principles and practice*, Third edition. Boston: Pearson, 2015.

[2] C. Christensen, ‘Review of Cryptography and Network Security: Principles and Practice, Fifth Edition’, *Cryptologia*, vol. 35, no. 1, pp. 97–99, 2010.

[3] S. N. Kishor, G. N. K. Ramaiah, and S. A. K. Jilani, “A review on steganography through multimedia,” 2016 International Conference on Research Advances in Integrated Navigation Systems (RAINS), 2016.

[4] A. Kumar, *Survey on steganography methods (text, image, audio, video, protocol and network steganography)*. 2017.

[5] A.-E. Hassanien, C. Grosan, and M. Fahmy Tolba, Eds., *Applications of Intelligent Optimization in Biology and Medicine*, vol. 96. Cham: Springer International Publishing, 2016.

[6] G. Hamed, M. Marey, S. A. El-Sayed, and M. F. Tolba, “Hybrid technique for steganography-based on DNA with n-bits binary coding rule,” 2015 7th International Conference of Soft Computing and Pattern Recognition (SoCPaR), 2015.

[7] G. Hamed, M. Marey, S. E.-S. Amin, and M. F. Tolba, ‘Hybrid, randomized and high capacity conservative mutations DNA-based steganography for large sized data’, *BioSystems*, vol. 167, pp. 47–61, May 2018.

[8] K. S. Sajisha and S. Mathew, ‘An encryption based on DNA cryptography and steganography’, in *2017 International conference of Electronics, Communication and Aerospace Technology (ICECA)*, 2017, vol. 2, pp. 162–167.

[9] D. Zebari, H. Haron, and S. Zeebaree, *Security Issues in DNA Based on Data Hiding: A Review*, vol. 12. 2017.

[10] M. P, M. M, M. R, V. Raghavan, and V. R.e., ‘Highly Improved DNA Based Steganography’, *Procedia Comput. Sci.*, vol. 115, pp. 651–659, Jan. 2017.

[11] A. Khalifa, A. Elhadad, and S. Hamad, “Secure Blind Data Hiding into Pseudo DNA Sequences Using Playfair Ciphering and Generic Complementary Substitution,” Applied Mathematics & Information Sciences, vol. 10, no. 4, pp. 1483–1492, Jan. 2016.

[12] H. Hussein and W. M. Abduallah, *A Modified Table Lookup Substitution Method for Hiding Data in DNA*. 2018.

[13] The Structure of DNA. [Online]. Available: http://ircamera.as.arizona.edu/Astr2016/text/nucleicacid1.htm. [Accessed: 30-Nov-2019].

[14] G. Hamed, M. Marey, E.-S. Safaa, and M. Tolba, *Hybrid Randomized and Biological Preserved DNA-Based Crypt-Steganography Using Generic N-Bits Binary Coding Rule*. 2016.

[15] “Secret Code Breaker,” Google. [Online]. Available: https://books.google.com.sa/books?id=Ew0yhEGzvwUC&printsec=frontcover&hl=ar#v=onepage&q&f=false. [Accessed: 30-Nov-2019].

[16] P. . Vijayakumar Vijayrgcet@gmail. co., V. Vijayalakshmi vvijizai@pec. ed., and G. . Zayaraz gzayaraz@pec. ed., ‘An Improved Level of Security for DNA Steganography Using Hyperelliptic Curve Cryptography’, *Wirel. Pers. Commun.*, vol. 89, no. 4, pp. 1221–1242, Aug. 2016.

[17] A. Khalifa, “LSBase: A key encapsulation scheme to improve hybrid crypto-systems using DNA steganography,” 2013 8th International Conference on Computer Engineering & Systems (ICCES), 2013.

[18] S. Marwan, A. Shawish, and K. Nagaty, ‘DNA-based cryptographic methods for data hiding in DNA media’, *BioSystems*, vol. 150, pp. 110–118, Dec. 2016.

[19] A. Khalifa, A. Elhadad, and S. Hamad, *Secure Blind Data Hiding into Pseudo DNA Sequences Using Playfair Ciphering and Generic Complementary Substitution*, vol. 10. 2016.

[20] A. A. AL-Shaaby and T. AlKharobi, ‘Cryptography and Steganography: New Approach’, *Trans. Netw. Commun.*, vol. 5, no. 6, p. 25, Dec. 2017.

[21] K. Santoso, S.-H. Lee, W.-J. Hwang, and K.-R. Kwon, ‘Sector-based DNA information hiding method’, *Secur. Commun. Netw.*, vol. 9, no. 17, pp. 4210–4226, Nov. 2016.

[22] K. Kordov and B. Stoyanov, ‘Least Significant Bit Steganography using Hitzl-Zele Chaotic Map’, *Int. J. Electron. Telecommun.*, vol. 63, no. 4, pp. 417–422, Nov. 2017.

[23] “Enhanced lsbase: a dna based security scheme to augment audio steganography,” International Journal of Latest Trends in Engineering and Technology, vol. 7, no. 4, 2016.

[24] M. A. Sharma, ‘Security and Information Hiding based on DNA Steganography’, p. 6, 2016.

[25] E. J. Kusuma edi. jaya. kusuma@gmail. co., C. A. Sari, E. H. Rachmawanto, and D. R. I. Moses Setiadi, ‘A Combination of Inverted LSB, RSA, and Arnold Transformation to get Secure and Imperceptible Image Steganography’, *J. ICT Res. Appl.*, vol. 12, no. 2, pp. 103–122, Apr. 2018.

[26] S.-H. Lee, ‘Reversible Data Hiding for DNA Sequence Using Multilevel Histogram Shifting’, *Secur. Commun. Netw.*, pp. 1–13, Mar. 2018.

[27] S. Marwan samiha. abdelrahman@bue. edu. e., A. Shawish ahmed. gawish@bue. edu. e., and K. Nagaty khaled. nagaty@bue. edu. e., ‘Utilizing DNA Strands for Secured Data-Hiding with High Capacity’, *Int. J. Interact. Mob. Technol.*, vol. 11, no. 2, pp. 88–98, Apr. 2017.

[28] A. Taha, A. S. Hammad, and M. M. Selim, “A high capacity algorithm for information hiding in Arabic text,” Journal of King Saud University - Computer and Information Sciences, 2018.

[29] S. Aljawarneh, M. B. Yassein, and W. A. A. Talafha, “A multithreaded programming approach for multimedia big data: encryption system,” Multimedia Tools and Applications, vol. 77, no. 9, pp. 10997–11016, 2017.

[30] S. Zaghbani, N. Boujnah, and M. S. Bouhlel, “High Capacity Data Hiding Scheme for DCT Image Using the YCbCr Color Space and Chaotic Map,” Journal of Image and Graphics, vol. 5, no. 1, pp. 10–15, 2017.

[31] M. Sabry, M. Hashem, T.Nazmy and M. E. Khalifa, “A DNA and Amino Acids-based Implementation of Playfair Cipher,” International Journal of Computer Science and Information Security, ISSN: 1947–5500, vol. 8, Issue No. 3, pp. 129–136, 2010.

**Appendix**

A.I