Republic of Iraq
Ministry of Higher Education and Scientific Research
University of Babylon
College of Sciences for Women
Department of Computer Sciences

A Message Hiding Method Based on Single Nucleotide Polymorphisms

A Thesis

Submitted to the Council of College of Science for Women, the University of Babylon in a Partial Fulfilment of the Requirements for the Degree of Master in Science\ Computer Science

By

Defaf Shiker Kadhum

Supervised By

Assit.Prof. Dr Sahar Adil Kadhum

2023 A. D. **1444 A. H.**

سُورَةُ العَلَقِ

الآيات (1-5)

Supervisor Certification
I certify that this thesis entitled

" A Message Hiding Method Based on Single Nucleotide Polymrphisms"

written by

"Defaf Shiker Kadhum"

was prepared under my supervision at College of Sciences for Women as a partial fulfillment of the requirements for the degree of a Master's in Computer Science.

Signature:

Name: Prof. Dr. Sahar Adil Kadhum

Date: / / 2023

Address: University of Babylon/College of Sciences for Women

Head of the Department Certification

In view of the available recommendations, I forward the thesis entitled " A Message Hiding Method Based on Single Nucleotide Polymorphisms "

" for debate by the examining committee.

Signature:

Name: Asst. Prof. Dr.Saif Alalak

Date: / / 2023

Address: University of Babylon/College of Science for Women

Acknowledgements

First of all, I thank Almighty God who inspired me with patience and strength to complete this study.

It is not easy except for what God makes easy.

I would like to express my sincere thanks and appreciation to my Supervisor

"Prof. Dr. Sahar Adil Kadum"

To guide and follow up with me and provide important tips and suggestions to improve this study. Without her, I would not have completed this thesis. I learned a lot from her on this journey of research. I am really grateful to her.

Dedications

To my great creator, Almighty Allah

To the first teacher of mankind, the prophet Mohammed , may God bless him and his family and grant them peace.

To my intercession with God in this world and the hereafter, the pure imams, peace be upon them.

To the example of dedication and devotion .... my beloved father.To whom offered me happiness and comfort over her happiness ... My honorable mother.

To those who supported me and encouraged me with all love and patience.....my husband and daughters

To those who wish happiness and success for me from the bottom of their hearts without any compensation ... my dear brothers and sisters.

To all my dear loyal friends

I offer you that humble work .....

Defaf

Abstract

DNA sequences have attracted much interest as pieces of quaternary digit information that can be used to store information, solve problems, encrypt and hide messages.

Modern research papers lean towards in exploiting DNA features in the security aspect. They used DNA features as a new idea in embedding the transferred data instead of using DNA as a hidden carrier only as in old methods. These researches use DNA features called single nucleotide polymorphisms (SNPs) as a one of a modern DNA-Steganography method. But, there are several limitations in using these research methods that highlight some weaknesses points, such as sequential storage for the message segment that can easily revealed, using the same lookup table for all entered messages, so compromising this table is inevitable, another limitation like ambiguity or mutations will significantly decrease hiding capacity, Therefore, this thesis tend to solve these limitations. In this thesis, a developed DNA steganography methodology was proposed to hide a message in a variable regions (SNPs) exploiting the DNA features. Through this method, a message was encrypted to a DNA form in order to be hidden randomly in a SNPs genome using dynamic lookup table for each message. The proposed DNA steganography methodology in this thesis prove its advantage in protecting the engineering of cells as clarified from the obtained results. Additionally, Use SNPs in DNA as hiding locations provide good noise, Zero payload, high capacity, low modification rate, high information entropy, low cracking probability, blind algorithm and preserve functionality.

Table of Content

Table of Figures

List of Tables

List of Algorithms

List of Abbreviations

Abbreviations	Meaning
BER	Bit Error Rate
CNV	Copy number variants (CNV)
DNA	Deoxyribonucleic Acid
EK	Embedding key
IT	Information Technology
LK	Location Key
Msg	Message
NCBI	National Center For Biotechnology Information
NIST	National Institute of Standard Technology
SNP	Single Nucleotide Polymorphisms

Appendixes

Chapter One

General Introduction

CHAPTER ONE
GENERAL INTRODUCTION

1.1 Introduction

With the rapid advancement of communication technology and the internet, sharing of information over the internet and mobile networks has become a common form of communication [1]. it is very essential that the communication is made in an extremely secure manner, with the primary concern being how to transmit the information securely and prevent the data from hacking, unauthorized access, or modification; As a result, information security became crucial to facilitating the confidential exchange of information between any sender and receiver. Preserving this information, a security techniques invented such as data hiding and cryptography that used the most frequently in the sectors of communication and computer security [2][3].

Cryptography is the science of using mathematics to encrypt and decrypt data to keep messages secured by transforming intelligible data form (plaintext) into an unintelligible form (cipher text) [4]. On other hand, steganography used to disguise data so that thoroughly others who are not intended recipients do not recognize the stego-graphic medium that carrying secret information. Steganography is a method of concealing private messages in digital media (images, audio, video, and text) so that no one suspects their existence. Steganography varies from cryptography in that cryptography is concerned with keeping the contents of messages secret. While steganography is concerned with concealing the presence of the message. Both techniques are effective at keeping data from unauthorized access, but neither technique is perfect and can be exploited. Hybridizing steganography with cryptography increases the security strength [5][6].

Nowadays, new discipline has been merged with both techniques to improve their performance and security such as bioinformatics. DNA computing is one of bioinformatics methods served as the inspiration for DNA cryptography, a relatively new cryptographic technique that uses DNA as an information transport. In fact, DNA can be employed for computation information storage and transmission [7]. Due to its benefits Deoxyribonucleic Acid (DNA) including its ultra-high storage density, ultra-low energy consumption and potential for ultra-large-scale encryption for data concealing. The DNA sequence is intriguing. A public database of DNA sequences contains over 163 million of them. Therefore, using DNA sequence as a medium considerably lowers the likelihood of the system being cracked and strengthens the hiding system [8] [9].

Using DNA-based steganography is to increase the hiding capacity than other digital media because of its huge data hiding capacity and its high redundancy and randomness, is almost used now in most steganography applications. It has many characteristics which make it an excellent steganography medium. A mentioned earlier, DNA-based data hiding makes use of DNA as a cover media for secure data transmission [10] [11].

In modern DNA steganography, a new trend is used such as exploiting DNA engineering cells features. DNA is not used merely as a carrier, modern techniques tend to exploit DNA engineering cells features to offer better solutions and performance compared to the old generation methods [21]. One of these interesting features is the single nucleotide polymorphism (SNP) called SNPs (pronounced “snips”), are the most common type of genetic variation among people or Organisms. Each SNP represents a difference in a single DNA building block, called a nucleotide. So, the SNPs are considered a genetic change in a single base in the DNA strand [12].

All the above mentioned schemes inspire this thesis to develop a new steganography algorithms based on DNA SNPs for hiding a secret message in variable regions of DNA strand.

1.2 Related works

This section illustrates the works that have already been done to hide information using DNA steganography. This section has been divided into two sub sections: related works using a traditional DNA- steganography, and related works of modern DNA- steganography.

1.2.1 Traditional DNA Steganography

Different techniques are exploited for hiding information through communication channels using a traditional DNA- steganography techniques are presented in the following lines. Table (1.1) shows the summary of related work using traditional DNA- steganography:

In 2020, Marghny et al, proposed a new method to embed secret data in DNA files using genetic algorithm to enhance the performance of steganography performance measures. By tackling the problems of substitution method, using genetic algorithm to choose the best positions in the DNA file, to embed secret data, make the modification rate equal 0 in most cases. The cracking probability of the algorithm is very low and improvement in modification rate of the carrier. For more security, the secret data are encrypted with RSA algorithm before embedding [13].

In 2021, Amal, proposed a hiding method that combined both encryption and Steganography to build a secure communication between related parties. The proposed method uses DNA sequence data as a cover to hide the secret message. The hiding process start with message encryption using XOR cipher based on DNA digital representation. The hiding process

starts with complementary substitution operation followed by a random insertion process. Furthermore, a fixed-size header is embedded right before the message itself to facilitate the blind extraction process. [14].

In 2021, Shah, Parsa , ..etal, proposed a method for hiding a message into a cover DNA sequence to generate a Stego-DNA sequence to transmitted in an unsecured channel. The DNA sequence is chosen randomly from a database The data can be converted to binary form and substituted by corresponding nucleotide bases by a substitution lookup table. The substitution lookup table is a relational dataset where the rows represent 2-bit binary information and columns represent the corresponding nucleotide bases. This table is inversed at receiver site, the row number represents nucleotide bases of the Stego-DNA sequence, and column number represents the nucleotide of the selected DNA sequence [15].

In 2021, Amal: proposed a method for hiding starts by encrypting the message using a bio-inspired 8x8 play-fair ciphering algorithm. The secret sequence is then randomly spliced and merged into the cover sequence replacing its non-coding regions [16].

In 2021, Amany, et al , proposed a method based on substitution method for data hiding in DNA sequences. The data is encoded using a binary coding rule to be hidden in a DNA sequence. The DNA Amino acids can be organized into groups where each DNA codon in one of the groups can be used to encode two bits of the hidden message. The proposed method is blind, preserves the DNA original biological structure in the fake DNA sequence and provides no expansion in the DNA sequence. The proposed method is evaluated using a public DNA sequences dataset named BALIBASE. The evaluation results showed that the proposed method achieved about 50%

increase in the data hiding capacity. Moreover, the results showed that the proposed method resulted in significant decrease in the cracking probability [17] .

1.2.2 Modren DNA Steganography

Related works to a modern DNA- steganography**.** Table (1.2) shows the summary of related work using modren DNA- steganography:

In 2019, Partha, Lubna, et al, proposed framework using DNA steganography that provides a higher payload capacity, using balanced tree data structures for message encoding where the leaf node contains the intended message. This unique process of message encoding and decoding guarantees a payload capacity of greater than or equal to 0.50 [18].

In 2019, Marghny, Botheina, and Ahmed: proposed an algorithm called self-adaptive DNABS (DNA-based steganography) . This algorithm is applied for data hiding without changing the function or the type of the original DNA protein. It is implemented using a DNA-based steganography and a Neural Network (backpropagation) algorithm to achieve a lower cracking probability than other techniques [19].

In 2020, Zeena and Melad: proposed a secure data hiding method using DNA sequences. The method begins with DNA coding uses the eight rules of DNA code based on a sequence of characters in text. To increase the security of encryption text, a hyper-chaotic system is used for obtaining the color image position that used to hide on it. The proposed steganography method hides a character of the encrypted text in each pixel of the cover [20].

In 2020 Na, Dokyun: proposed a DNA steganography methodology to hide messages in variable regions using single nucleotide polymorphisms (SNP). This approach is expected to be useful for tracking cells and protecting biological asset [21]. the view of the related work that used a modern DNA-Steganography trends, the proposed method of this thesis has Related to work[21].

-In 2023 , Bambang Harjito et al : proposed a combined method between the Playfair cipher cryptographic technique and the DNA substitution steganography algorithm for data security. Comparative analysis was conducted on techniques that are being tested by using similar methods. Four tests are conducted to measure the performance of the proposed method and another method was tested for comparative purposes [22].

-In 2023 , Raj Kumar Ettiyan et al: proposed a hybrid chaotic DNA and AES encryption methodology that combines the powerful features of 3D logistic chaotic maps and employs DNA operations in an AES system to provide more security against IoT attacks. Extensive experimentation is carried out using National Institute of Standard Technology (NIST) tests [23].

Table (1.1): Summarey of Related work using DNA Steganography

Ref/year	Author	Filed Type	Methodology	Measurement
[13]	Marghny et al	Biology	Hiding data using enhancement substitution method DNA file and genetic algorithm encrypted with RSA algorithm.	modification rate equal = 0, The probability is very low and improvement in modification rate of the carrier
[14] 2021	Amal	Biology	Hiding data in DNA cover using complimentary substitution method followed by insertion method, the cipher using	High Capacity, cracking probability =0, and algorithm is blined

			XOR based on DNA data representation	
[15] 2021	Shah, ..etal	Biology	The data converted to binary and substituted by corresponding nucleotide bases by a substitution lookup table. The substitution lookup table is a relational dataset where the rows represent 2-bit binary information and columns represent the corresponding nucleotide bases.	Capacity high Zero payload. Low cracking probability BPN =2
[16] 2021	Amal	Biology	encrypting the message using a bio-inspired 8x8 play-fair ciphering algorithm. The secret sequence is randomly spliced and merged into the cover sequence replacing it non-coding regions	high capacity blind extraction
[17] 2021	Amany, et al	Biology	Using group os amino acid to code msg. bits and substitute in DNA cover	50% increase in capacity, Payload =0,preserve functionality, low cracking

Table (1.2): Summary of Related Work Using Modern DNA Steganography

Ref/Year	Author	Field type	Methodology	Measurement
[18] 2019	Saha etal.	Data structure	Use one nucleotide in leafs nodes for hiding capacity. Use a tree structure	Pattern dependent technique. Do not Preserve biological features. High modification rate. Over 50% hiding capacity. Scattered hiding locations. Blind. Zero payload.
[19] 2019	Mohammed etal.	Artificial intelligence	Develop a robust DNA steganography algorithm using ANN.	Preserve biological features. Zero payload. Low cracking probability Not blind. High execution time. Sequential hiding pattern.
[20] 2020	Zeena,and Melad	Chaos system & image processing	Use dynamic DNA coding, Use the Liu system to generate four pseudo-random sequences, Hide byte of encrypted data EBS1 in each pixel	strength and difficulty in breaking and analyzing the code, loss of data BER=0
[21] 2020	Na, Dokyun	Biology Networking	Use SNPs in DNA as hiding locations. Developing a mutation detection method.	SNPs provide good noise cover. Single mutation detection. Zero payload. Blind algorithm. Preserve functionality. Low hiding capacity.

1.3 The Problem Statement

Bioinformatics is one such directions. Many research papers tend to exploit bioinformatics techniques in the security aspect, such as work [21] used DNA features to find a new idea to hide the data instead of the old methods that used DNA as a carrier in embedding data. Work [21] uses DNA features called SNPs. Which is considered one of the modern methods in DNA-Steganography. But, there is some indications in the method highlight some weaknesses, such as sequential storage for the message segment that can easily revealed, using the same lookup table for all entered messages, the compromising of this table is inevitable, another indication is exploiting biological characteristics like ambiguity or mutations will significantly decrease hiding capacity, …etc. Therefore, this thesis tend to solve these indicators.

1.4 The Aim of Thesis

The Aim of this thesis is to solve the above problems through achieving the following objectives:

adopt and Implement a method for Hiding information that is more secure, with

- High capacity
- Coding messages using
    - Dynamic lookup table
        - Random permutation
        - Integer coding
- Scattering hiding method depends on
    - Special keys generated randomly

- Ensuring the integrity using
  - Dedicated DNA sequence , and
  - SNPs complementary base
- Develop a method to detect mutation (obstacles) in the SNPs.

1.5 Thesis Layout

Additionally, to chapter one, four chapters will be introduced:

- **Chapter Two:** presents the basic principles of the theoretical background such as: Deoxyribonucleic Acid computing (DNA computing), Nucleic Acid, DNA based data protection, single nucleotide polymorphisms (SNPs), SNPs database.
- **Chapter Three:** Presents the theoretical algorithms of the proposal system.
- **Chapter Four:** Presents the implementation and results of the proposal system.
- **Chapter Five:** summarizes the conclusions and future research suggestions.

Chapter Two

Theoretical Background

CHAPTER TWO
THEORETICAL BACKGROUND

2.1 Introduction

This chapter explains the theoretical background of the subjects the thesis related to such as Deoxyribonucleic Acid Computing (DNA computing), Nucleic Acid, DNA Cryptography,DNA Steganography ,Single Nucleotide Polymorphisms (SNPs), database of National Center Biotechnology Information (NCBI).

2.2 Deoxyribonucleic Acid Computing (DNA Computing)

The process of DNA computing involves bimolecular computation that uses biological methods in order to perform massive sequential computations. Nowadays, DNA computing is an area of natural computing [24]. This area based on the notion of the ability to perform both logic and arithmetic operations using molecular biology processes performed on the biological structure of DNA to encode the information. As a result, the biological technology is used as implementation tool and DNA is used as information carrier [25]. DNA computing is an inspiration drawn from biomedical stream. The structure exactly maps to the human genome. Thus, it can be either single stranded or double stranded. It not only has huge amount of information storage but also it provides parallel processing. Due to both capabilities it is expected to a become super-computer in near future [24] . The most recent biological method used in a variety of applications is DNA. This is because a variety of computational problems with exponentially increasing calculation times [26] [27].

2.2.1 Advantages of DNA Computing

- **Minimal Power Requirements**: While the computation is taking place, DNA computing is not required power. Because the chemical bonds that are the building blocks of DNA, happen without any outside power source. There is no contrast to the required power in traditional computers [28] [29].
- **Minimal Storage Requirements:** About 1 bit per cubic nanometer is the density of DNA molecular, where 1012 cubic nanometers of traditional storage media needs to store only 1 bit [30].
- **Speed:** Traditional computers can execute around 100 million of instruction per second. Combining DNA strands as demonstrated by Adleman made computations corresponding to 109, i.e. over 100 times faster than the fastest computer [19] [27].

2.3 Nucleic Acid

Nucleic acids are a cluster of biomolecules which are being part of the cell nucleus. These nucleic acids are long polymers made up of monomeric elements (units) known as nucleotides: Adenine, Cytosine, Guanine ,Thymine and Uracil [28]. There are two types of nucleic acids present in the cell nucleus: DNA and Ribonucleic Acid (RNA) [31].

2.3.1 Deoxyribonucleic Acid (DNA)

The Deoxyribose Nucleic Acid (DNA) is the biological molecule that possesses all the genetic information of the cell and it is responsible for transfer genetics from the parents to their offspring [24].

These molecules are bound as a two long strands are twisted around like a ladder as in Figure (2.1). Each strand is made up of units of

nucleotides which consists of three basic molecules: sugar (S), a phosphate (P) group, and one of four nitrogen bases. The nitrogenous bases are Purines (A and G) and Pyrimidines (T and C). Every DNA can be viewed as a sequence of bases (AAGTCGATCGATCATCGATCATACGT). Every three adjacent bases constitute a unit known as the codon which corresponds to a specific amino acid [32].

DNA molecules are inbuilt having exceptional energy efficiency, huge parallelism and immense information density. These characteristics will add on security like authentication, encryption and many more . Further. DNA is used as a molecular tool for exploring theories because of its properties and the DNA operations are used as basic operations in the IT (Information Technology) field [24][27].

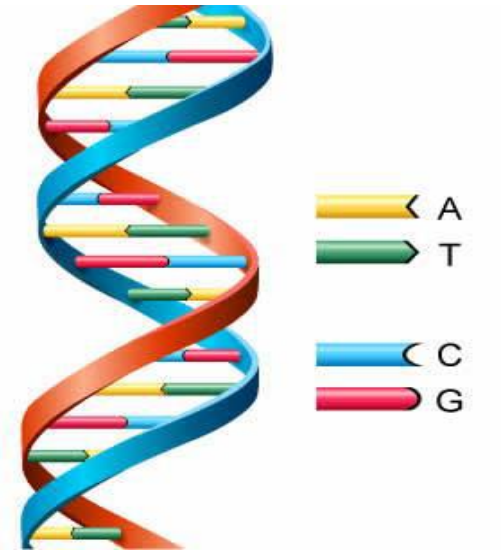

Figure (2.1): DNA Molecule Structure [21]

2.4 DNA-Based Data Protection

This section discusses the role of DNA with cryptography, and information hiding.

2.4.1 Scope of DNA Based Cryptography

The cryptanalyst can easily cryptanalyze the advanced systems and today the whole globe is waiting for different ways to provide network security to get the entire information in order to secure the data it translates. Among the existing techniques is DNA cryptography has been found as a new promising trend in cryptography. Improving the security and reliability of data can be adopted based on the nature of DNA. DNA cryptography techniques have high security level, storage capacity, and more time for hackers to break the crypto system to decrypt the original message from cipher. The main reason for using bimolecular computation along with cryptography is to provide the technology having unbreakable algorithms [25]. This cryptography could be an advanced cryptanalytic model from newly rising bimolecular computation as this process can verify upcoming computations [33] [34] [35].

In practical, there is no direct connection between cryptography and biological genetic molecule, but there is a combination between these two aspects that could make wonders in the data security field [26].

Cryptography in DNA can be achieved in utilizing biological, arithmetic operations separately or combining the both [24] [35]. Figure (2.2) summarize various biological and arithmetic DNA operations. Data protection by using DNA based on Arithmetic operations [36].

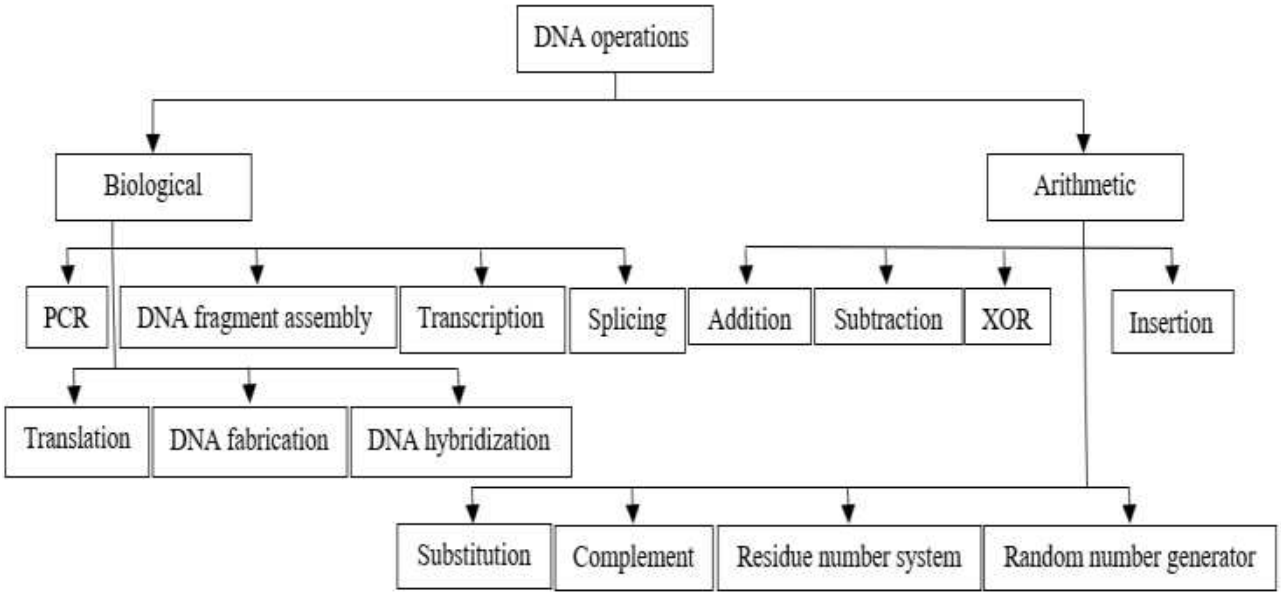

Figure (2.2): DNA Biological and Arithmetic Operations [21]

2.4.2 Scope of DNA Sequence Based Data Hiding

Data hiding based on the DNA sequence has been attracting much attention due to its potential storage capacity. The data hiding techniques based DNA include traditional hiding techniques, cryptography, steganography and watermark [37].

A- DNA-Based Steganography

In the field of steganography, mediums like text, image, audio, and video are used by researchers as containers to hide a message inside them. All the covering mediums mentioned above struggle to cope with the increasing size of information as well as meet the demanded security measures [38]. An urgent need has arisen for a concealing medium capable of holding a large amount of data without corrupting or degrading the quality of this medium. Consequently, deoxyribonucleic acid (DNA) is proposed as the ultimate concealing medium that avoids or mitigates the drawbacks of other mediums. DNA's most important

feature is the huge capacity it has around 215 petabytes of data can be stored in one gram of DNA. Another useful feature is the randomness of the building blocks forming the DNA. Besides that, low power is required when dealing with DNA computing which leads to fast execution [26]. For all the reasons mentioned above, many DNA steganography algorithms have been suggested since the beginning of the twenty-first century [2]. Figure (2.3) illustrate the DNA steganography technique. Three components are needed in DNA steganography that combined to get the fake DNA sequence. These are the covering medium, the message, and the secret key as shown in Figure (2.4). DNA-based steganography that relies on three techniques; insertion technique, substitution technique, and a complementary technique [37].

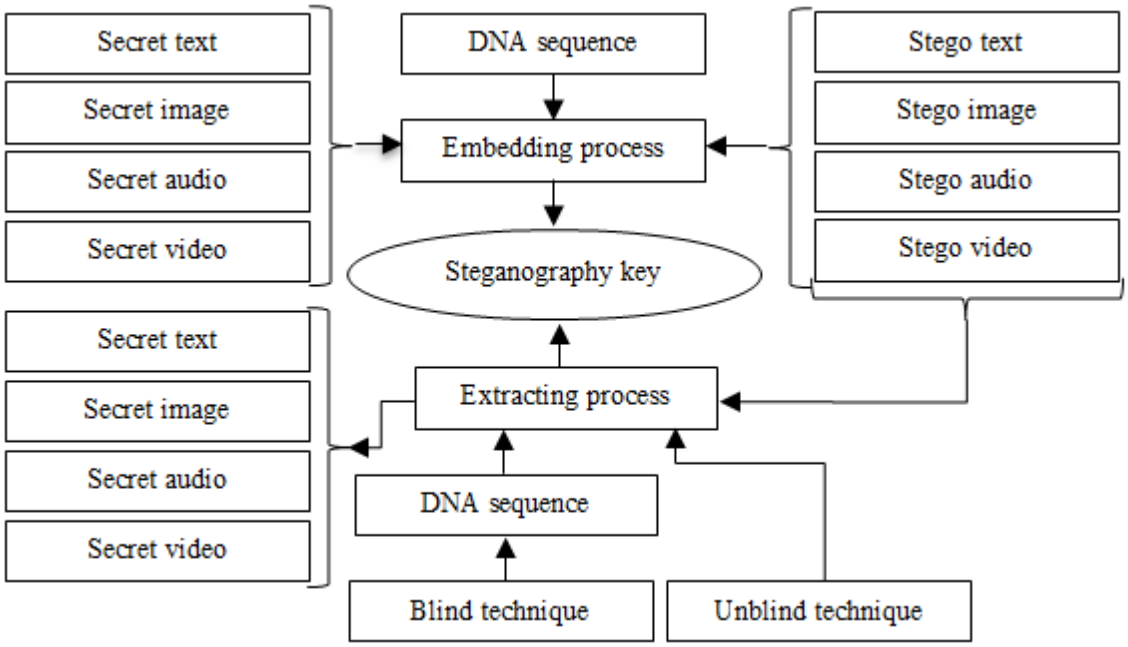

Figure (2.3): A General Block Diagram of DNA Steganography process[2]

Figure (2.4): DNA Steganography Components [35]

a) Insertion Technique

The Insertion-based technique is one of the methods to embed a message in a DNA sequence. It is performed by inserting the data of the embedded message in one or more different locations of the DNA reference sequence.

The benefits of this technique are high capacity, easy to implement, and Low cracking rate, Furthermore, less modification rate is considered as a great feature of this technique because it depends on inserting secret data in the DNA reference not replacing the contents of DNA reference. The main drawbacks of this technique are the increase in DNA reference sequence size, the length of faked DNA which is higher than the length of the original DNA, increase in redundancy during the process, Payload not equal zero, and not blind [40].

b) Substitution Technique

Regarding this technique, there is no merge between reference DNA sequence and the secret information. In this scheme, specific positions in the DNA reference are selected randomly as determined by the algorithm. After that, at least one complementary rule should be selected to replace each letter of the message with the DNA contents in particular locations. Depending on the contents of the message, the process will be carried out to obtain the stego DNA. Hence, the DNA

length is maintained following the embedding of the message only the replacement has done between secret message and DNA reference. This, in turn, means that in an effort to conceal the secret data, the resulting stego DNA is highly modified. As a result, this technique is considered as a more efficient technique than the previous techniques because it provides more complexity and better performance The main advantage of this technique is preserving the length of the DNA sequence after hiding the secret message. The main drawbacks of this technique is changing of bases in DNA sequences [37] [40].

c) Complementary Rule-Based Technique

In this technique, the procedure begins with the selection of a DNA sequence in which the longest existing complementary pair is contained. This is followed by the random generation of two complementary string pairs whose length is one more than that existing in the sequence, after which these pairs are padded with a 'T' at the posterior and anterior. Afterwards, they are inserted one at a time into *S* while ensuring that there is no overlapping. The message is then divided into segments, each containing an even number of bits after which the data is coded back into nucleotides using the binary coding rule. For each pair of a complementary substring in the converted sequence, a message bit is inserted before T*aj*T, where *aj* represents the pair of longest complementary substrings. A resultant sequence containing message *S*' is then obtained. This scheme results in a significantly alters the length of the DNA sequence which rouses the suspicion of a hacker to the existence of an embedded message [39][40]. Legally, there are six main complementary rules as it is shown in Figure (2.5).

AT	TC	CG	GA
AT	TG	GC	CA
AC	CT	TG	GA
AC	CG	GT	TA
AG	GT	TC	CA
AG	GC	CT	TA

Figure (2.5): The Six Complementary Rules [39]

In the attempt of improving the performance of currently used DNA steganography algorithms, a new trend has evolved. This trend relies on the concept of utilizing one of the DNA biological features and/or merging a technique existing in one of the fields of computer science with a DNA-based steganography algorithm. Doing so showed promising results in terms of overcoming or at least alleviating the issues and gaps that existed in the field of DNA steganography. But with the advent of new solutions, new issues have also arisen [41].

2.5 Modern Methods of DNA Steganography

The new trend aims either to exploit the biological attributes of DNA or borrow a suitable technique from another field of computer science. The reason behind that is offering better solutions and performance compared to the old generation methods. These trends are [42]:

- Exploiting biological features
- The field of networking
- The field of data structure
- The field of artificial intelligence

Exploiting biological features trend is what this thesis interest**.**

2.5.1 Exploiting Biological Features

One of the most interesting features of DNA is the condition called single nucleotide polymorphism (SNP). where, a particular nucleotide in the genome sequence differs between members of the same species and the regions including SNPs have the potential to hide a secret message. Attackers cannot distinguish between changes that occur because of data hiding and changes that occur because of SNPs [43].

Single nucleotide polymorphisms, frequently called SNPs (pronounced "snips"), are the most common type of genetic variation among people. Each SNP represents a difference in a single DNA building block, called a nucleotide. For example, a SNP may replace the nucleotide cytosine (C) with the nucleotide thymine (T) in a certain stretch of DNA as shown in Figure (2.6) [21][42].

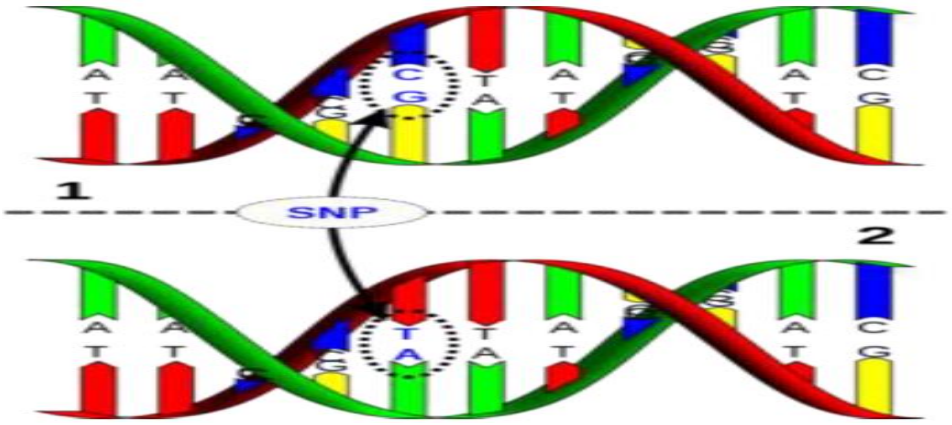

Figure (2.6): Single Nucleotide Polymorphism [20]

SNPs occur normally throughout a person's DNA. They occur almost once in every 1, 000 nucleotides on average, which means there are roughly 4 to 5 million SNPs in a person's genome. These variations occur in many individuals; to be classified as a SNP, a variant is found in at least 1 percent of the population.

Scientists have found more than 600 million SNPs in populations around the world [44].

SNPs differ from substitution variants, which replace one DNA building block (nucleotide) with another. Substitution variants usually cause disease and are generally not found in 1 percent of any population. Additionally, SNPs differ from copy number variants (CNVs), which occur when a whole gene (or other large section of DNA) is duplicated or deleted. Most commonly, SNPs are found in the DNA between genes. They can act as biological markers, helping scientists locate genes that are associated with disease [21] [44]. While the SNPs are naturally polymorphic, so it becomes difficult to the hacker to determine whether a nucleotide is a SNP or a part of an encrypted message, this characteristic provides a good security level in protect transferred data with high capacity. Moreover, in this method, the mutational errors induced information changes in network data transmission are detected and easily fixed [45] [46].

2.5.2 Finding SNPs in Human Genome

Scientists approach the problem of identifying, cataloging, and characterizing SNPs in two main ways:

A-Genomic Approach

This approach is used by scientists who want to see the big picture. Several large-scale projects have combined the efforts of many institutions to identify and catalog all of the SNPs in the 3-billion-base pair human genome. Each project involves hundreds of scientists, who compare the genomes of numerous individuals to identify the differences. These comparisons require a lot of computer-powered data analysis. As they work, scientists sort and catalog their results in databases that are available to anyone over the Internet [43].

B-Functional Approach

This approach is used by scientists who are interested in a particular disease or drug response. The biological processes involved in diseases and drug responses are controlled by the activities of many genes. Scientists interested in a particular process select genes known to be involved in the process and examine them in people who have a response or disease, as well as those who do not. By comparing people's DNA sequences, scientists can identify SNPs that correspond with a particular function or response. Figure (2.7) illustrate this comparison.

Each SNP location in the genome can have up to four versions: one for each nucleotide, A, C, G, and T. A SNP and its distribution in a population might look like the images below and to the left.

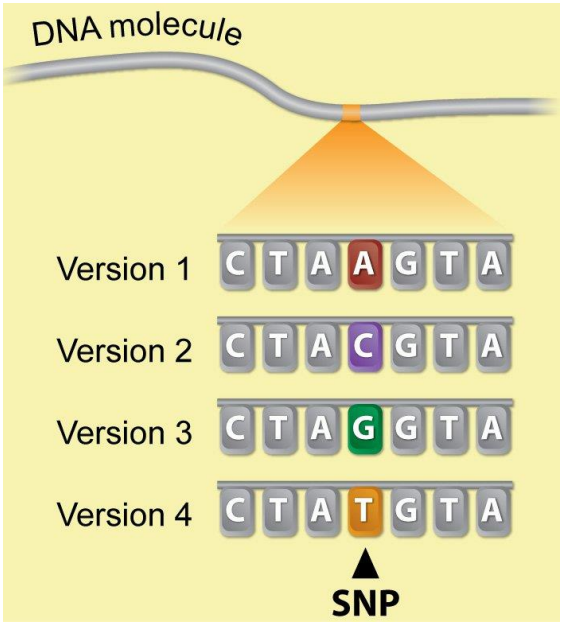

Figure (2.7): SNPs Identification [42].

Not all single-nucleotide changes are SNPs, though. To be classified as a SNP, two or more versions of a sequence must each be present in at least one

percent of the general population. SNPs occur throughout the human genome about one in every 300 nucleotide base pairs. This translates to about 10 million SNPs within the 3-billion-nucleotide human genome. The distribution of SNPs population depict in Figure (2.8) [44].

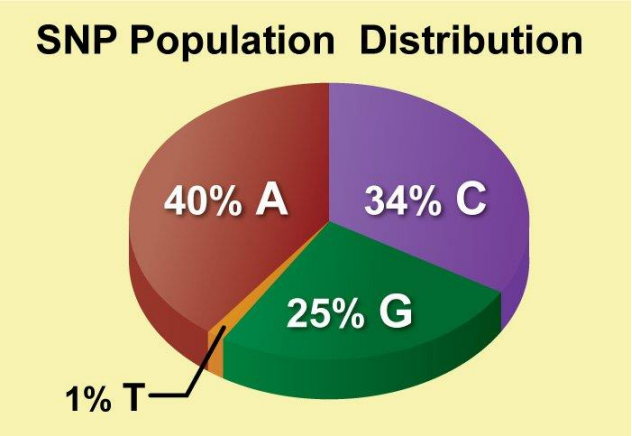

Figure (2.8): SNPs Population Distribution [38]

2.6 SNP Classification

To classify as a SNP, the change must be present in at least one percent of the general population. Second, most disease-causing mutations occur within a gene's coding or regulatory regions and affect the function of the protein encoded by the gene. Unlike mutations, SNPs are not necessarily located within genes, and they do not always affect the way a protein functions [42]. SNPs are divided into two main categories:

-Linked SNPs: (also called indicative SNPs) do not reside within genes and do not affect protein function. Nevertheless, they do correspond to a particular drug response or to the risk for getting a certain disease [45].

-Causative SNPs: affect the way a protein functions, correlating with a disease or influencing a person's response to medication. Causative SNPs come in two forms:

Coding SNPs, located within the coding region of a gene, change the amino acid sequence of the gene's protein product [24][45][46].

Non-coding SNPs, located within the gene's regulatory sequences, change the timing, location, or level of gene expression [47][48][49]. This classification is shown in figure (2.9)

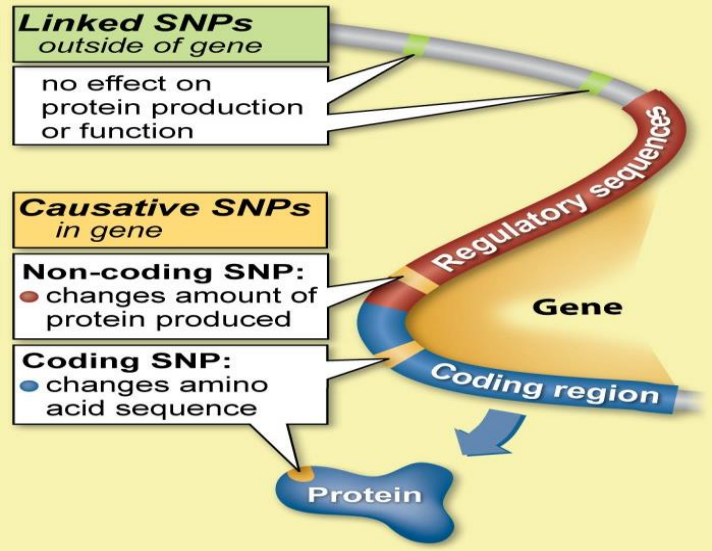

Figure (2.9): SNPs classification [45]

2.7-NCBI Database

The database is built and maintained by the National Center for Biotechnology Information (NCBI) . The NCBI houses a series of databases relevant to biotechnology and biomedicine and is an important resource for bioinformatics tools and services. Major databases include GenBank for DNA sequences. NCBI receives data from three sources: direct submissions from researchers, national and international collaborations or agreements with data providers and research consortia, and internal curation efforts. One notable effort is the Genome Reference Consortium (GRC) that provides the reference genome [51][52].

DNA-based steganography uses DNA sequence from the National Center for Biotechnology Information [49] .The required database is pulled from the website

(NCBI) as this website contains the databases for each vitamin. The vitamin D Receptor (vdr) database was chosen in the design of the program to embed the cipher text inside it. The size of the database is calculated It is required to embed the cipher text inside it [53] [54].

2.8 Metric Performance

This section illustrates some of the parameters used to evaluate the performance of the proposed algorithm such as:

2.8.1-Hiding Capacity:

Denotes the amount of data DNA sequence can tolerate. It is the maximum amount of data that can be contained in DNA [13] [55].

Capacity=Length of DNA Sequences…….(2.1)

2.8.2 Cracking Probability:

A measurement of the success probability attack to break the proposed security algorithm. It is the total probability to predict the confidential information hidden inside the reference DNA sequence. The attacker needs the following information to crack the secret message hidden in the reference DNA . The probability to predict these factors [12][56]:

Factor 1: Reference DNA sequence:

$$1/ (163\times10^{8}) \quad (2.2)$$

The reference DNA available is about $163*10^{8}$ (NCBI)

Factor 2: Find the message in the DNA reference

$$1/ (y-1) \quad (2.3)$$

Factor 3: Segments

$$1/\,2\,(m-1) \qquad (2.4)$$

Factor 4: Binary coding (A, C, G, T)

$$1/\,24 \qquad (2.5)$$

Factor 5: List of positions

$$1/\,l\,(y-1) \qquad (2.6)$$

Thus, the total probability to find the message hidden in the DNA sequence using the proposed method is

$$\mathbf{1/\,(163\times 10^{8})\times 1/\,(y-1)\times 1/\;\;2^{m}-1)\times 1/\,24} \qquad (2.7)$$

y: The length of the message after it is converted into DNA strings.

m: number of message segments.

2.8.3 Payload:

It is the amount of extra data (length) added to the DNA sequence due to the implementation of the DNA steganography algorithm. The best scenario occurs when the payload equals zero [13][41][42].

2.8.4 Modification Rate:

It denotes the ratio of change in the fake DNA sequence compared to the original one [57].

$$\textbf{Modification Rate}=\frac{\boldsymbol{Lenth\ of\ message\ in\ DNA}}{\boldsymbol{Lenth\ of\ DNA\ database}}\ldots\ldots\ldots(2.8)$$

2.8.5 Information Entropy:

The entropy of DNA refers to the measure of disorder or randomness in the sequence of nucleotides within a DNA molecule. In information theory, entropy is a mathematical concept used to quantify the amount of information contained in a

system**.**DNA is composed of four different nucleotides: adenine (A), cytosine (C), guanine (G), and thymine (T). The arrangement of these nucleotides along the DNA molecule forms the genetic code. Since there are four possibilities for each position in the DNA sequence, the entropy of DNA can be calculated based on the probabilities of finding each nucleotide at a particular position.
Mathematically, the entropy of DNA can be determined using the Shannon entropy formula:

$H = \Sigma P(i) \log 2 P(i)$

where H is the entropy, P(i) represents the probability of finding a specific nucleotide (A, C, G, or T) at a given position, and the sum is taken over all possible nucleotides [57] [58].

2.8.6 BER:

When the transmitter sends the information contained within the DNA this measure performs a match between the DNA before sending and the recipient's DNA to indicate the extent of the cover being distorted at the transmission or not [56].

$$\boldsymbol{BER} = \frac{number\ of\ nucleotides\ in\ sender}{number\ of\ nucleotides\ in\ receiver} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.9)$$

Chapter Three

The Proposed Work

CHAPTER THREE
PROPOSED WORK

3.1 Introduction

This chapter presents the theoretical implementation of the proposed work using the new steganography trend Single-Nucleotide Polymorphism (SNPs), presented as block diagrams, flow charts, figures, and algorithms with explanation to each presentation.

3.2 Proposed Work Structure

The general structure of the proposed work design divided in two sites: Sender and Receiver. Each side has its own activities to perform. Such that, the sender performs the security activities on the transferred message to protect it from hacking, while receiver site performs message extraction activities. Figure (3.1) depict the proposed work structure and figure (3.2) depict stages of proposed system.

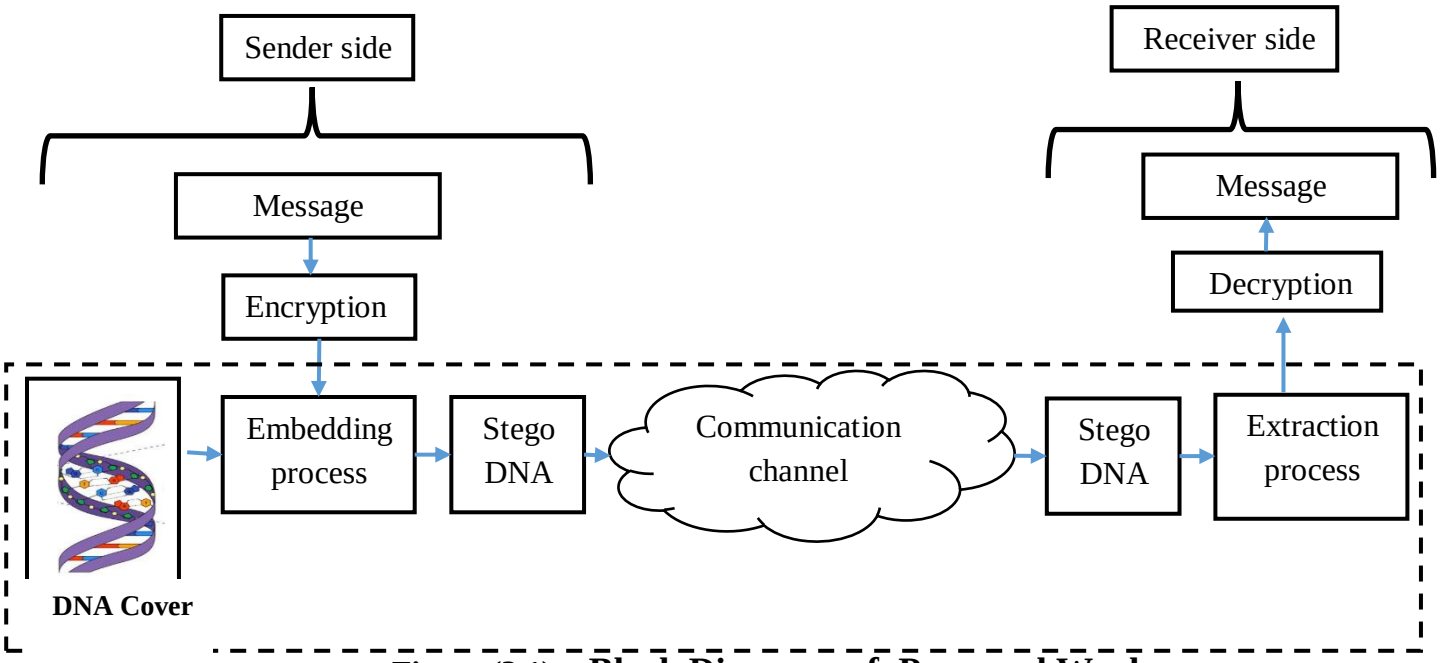

Figure (3.1) : Block Diagram of Proposed Work

3.3 Sender Side Activities

The Sender Side has to perform several stages to secure the transferred message before send it to the receiver. These stages include (Generating, Encryption, Embedding) stages as shown in figure (3.2).

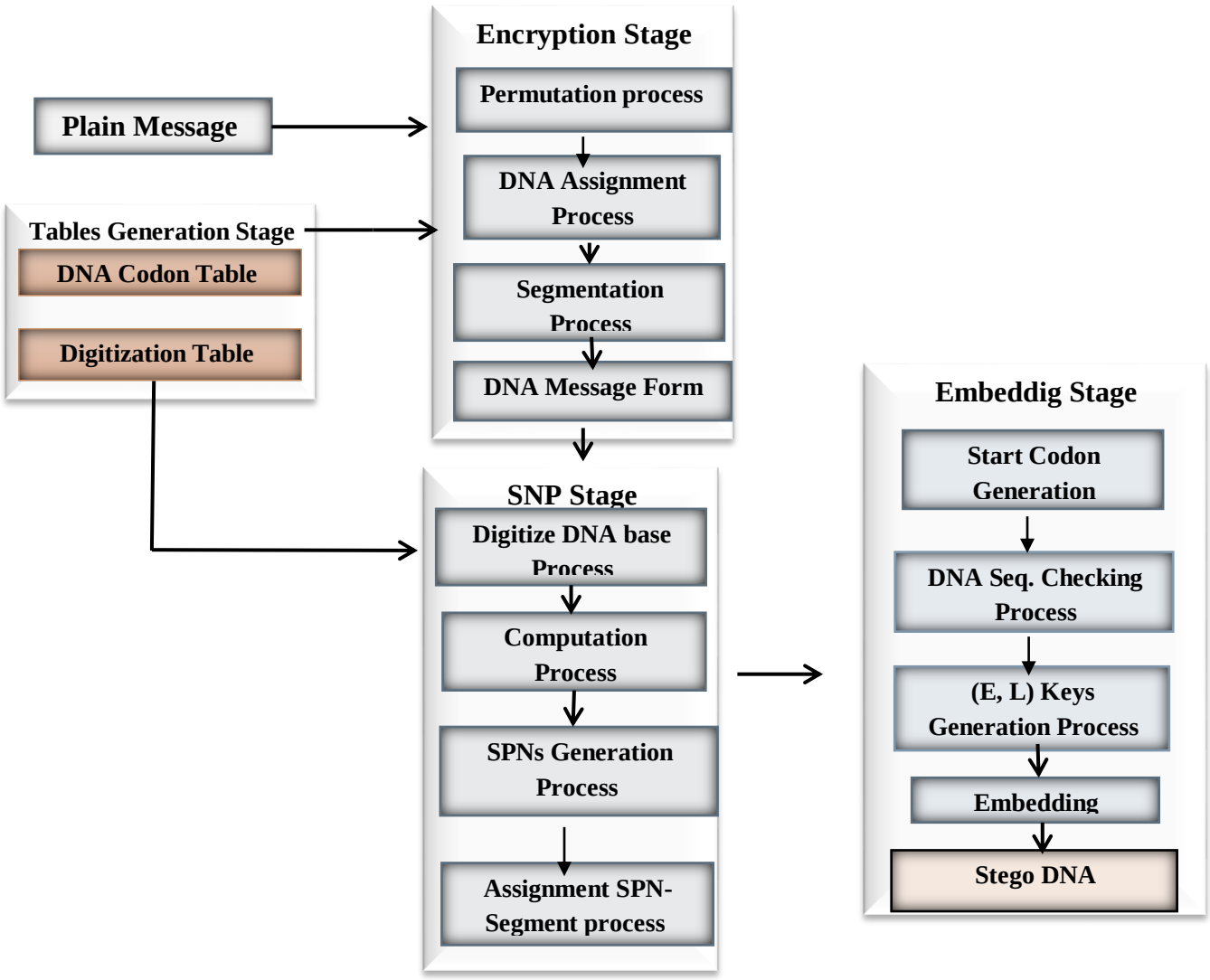

Figure (3.3): Sender Side Activities

The sender work passes through:

3.3.1 Tables Generating Stage

This stage is conducted to generate two types of tables: DNA codon table, base digitization Table to be used for the following work stages.

A- DNA Codons Table Generation Process

Generating this table is to assign the message characters to a DNA form. The message characters may include: (upper (A-Z), lower (a-z)) character cases, Special characters (@#* %.), and numbers (0-9).A bout 96 codons are generated in codons table. This process depicted in algorithm (3.1).

Algorithm (3.1): DNA Codons Table Generation

Input: String of character, [A]: array of codons

Output: DNA codon Table

*// **Generate the DNA codon table***

1: ***For*** *i=1 to 96*

2: *generated A[i]codons array // based on DNA bases* [A, T, C, G]

3: ***End // for***

4: *// **Assign each character to a specific codon such that:***

5: *// **Assign** the first 26 codons to uppercase character*

6: *// **Assign** the second 26 codons to lowercase character*

7: *// **Assign** the third 10 codons to number*

8: *// **Assign** the remaining 34 codons to special characters*

Table (3.1) : DNA Codons Table

GAA(K)	GAT(I)	CAT(M)	AAG(B)	TAG(F)
GAG(J)	CAG(N)	TAG(H)	AAC(D)	GAC(L)
ATA(R)	TTA(V)	GTA(Z)	CAA(d)	ATT(S)
TTT(W)	GGT(a)	CTT(e)	ATG(T)	TTG(X)
GTG(b)	ATC(Q)	CAT(f)	TTC(U)	GTC(Y)
CTA(c)	AGA(i)	GGA(q)	TGA(m)	CTA(u)
AGG(h)	TGG(l)	CGG(t)	GGG(p)	GCA(n)
GCT(0)	AGC(1)	ACC(2)	ACT(3)	CCT(4)
AGA(j)	GGA(r)	ATG(x)	ACA(w)	CCA(5)
CCG(6)	CGA(7)	CTC(8)	TTG(9)	TAC(.)
TAA([)	ATC(])	GAA(()	CCC())	ACC({)
TAC(})	TAA(z)	TGC(?)	GCA(#)	GCA(,)
GCT( )	ACT(y)	GAT(=)	GTT(*)	TTC(&)

The table is generated dynamically. Such that, for every new message session has its own DNA codon table is generated. For example, the character Z in some message represented by GTA, but in other session may represented by AAG.

B- Digitization Base Table Generation Process

The digitization table consist from DNA bases and their corresponding representation digit, this table is regenerated for each new message in a new

digitization form. The table content depicts in Table (3.1), and algorithm (3.2) describe the table generation process

Table (3.2): DNA Digitization Table

DNA Base	Coding
A	2
T	1
C	0
G	3

Algorithm (3.2): Digitization Table Generation

Input: DNA bases [A, T, C, G], R(random variable)
Output: Digitization Bases

```
1:  for each message do
2: for i= 0 : 3
3:     R = random (0….3)
4:     DNA bases [i]=: DNA bases [i]+ R
5:   end
```

For example, Digitization of bases in message (A=2 , T=1, C=0 ,G=3) ,So DNA codons of message are converted into string of coding number .

3.3.2 Encryption Stage

This stage has two levels of encryption. A: The first level is permutation process, while B: the second level is DNA encryption, This stage described in bellow.

A- The first level Encryption

This level of Encryption contains:

-Permutation Encryption level

In this level, the message indexes is permutated randomly in order get randomness in distributing message characters that gives a level of security. This process illustrated in algorithm (3.3).

Algorithm (3.3): Permutation Process

Input: plain Msg
Output: permutated Msg

1: Rand permutation (indexes of plain Msg)
2: Len=length of Msg.
3: for 1 to Len
4: Random permutation (indexes of)
5: End.

B -The Second Level Encryption

This level consists of three processes: (DNA Assignment, Segmentation and DNA Message form) processes.

- DNA codons Assignment Process

Every permutated character in the message will be assigned to a specific codons of DNA form to produce a DNA message.

The assignment procedure depends on Table (3.1).

- Segmentation Process

Implementing this process requires to compute the length of the message. This computation is necessary for determining the number of the segments that message will be divided into. The process of segmentation the message will give more randomness when hiding the encrypted message. Each segment will have a length of 64 to be initialized to the next stage. Algorithm (3.4) depict this process.

Algorithm (3.4): Segmentation Process

Input: DNA Msg
Output: number of Segments

```
// Segmentation Msg  processes.

1:      compute the length of mgs
2:      Len=length (Msg)
3:      if Len < 64 then
4:          pad the Msg with 0   // Each segment of 64 characters   long
5:       else
6:          if Len = 64 then
7:             msg.seg.no. =1
8:           else
9:              while   Len > 64
  10:                 msg.seg.no.:  = Len /64   //    number of segments
  11:              end
  12:          end // if
  13:     end // if
```

-DNA Message Form

In this step, the message is arranged in a triple codons of DNA and structured to generate the SNPs.

3.3.3 SNPs Generation Stage

To generate a SNP for each segment two processes are needed:

A-Segment Configuration Process

Each message segment is configured to a 2D- square array of (8*8). In order compute a SNP for each row and column according to a mathematical operation.

B- DNA base Digitization Process

Each base in the array (rows, columns) will be coded to a specific digit according to DNA digitization Table (3.1). A mathematical operation (sum & mod) will be applied for each (rows and columns) to generate a specific SNP for each row and column. Algorithm (3.5) illustrates this operation.

Algorithm (3.5): SNPs generation

Input: DNA array of bases

Output: DNA array of bases & SNPs

// assign DNA base with digit

```
For each segment
  Numbering Bases
  for each row
    Apply mathematical operation (sum & mod)
      generate SNPs and inserted at the end of row
  end for
 for each column
    Apply mathematical operation
      generate SNPs and inserted at the end of column
      end // for
End // for
```

3.3.4 Embedding Stage

To embed the final ciphered message, form several steps has to do such as (examine the pulled up DNA database length, generate start codon for each segment generate EK and LK keys). The name of the DNA database used here is D. vitamin which can be obtained from NCBI. Figure (3.3) depicts this process.

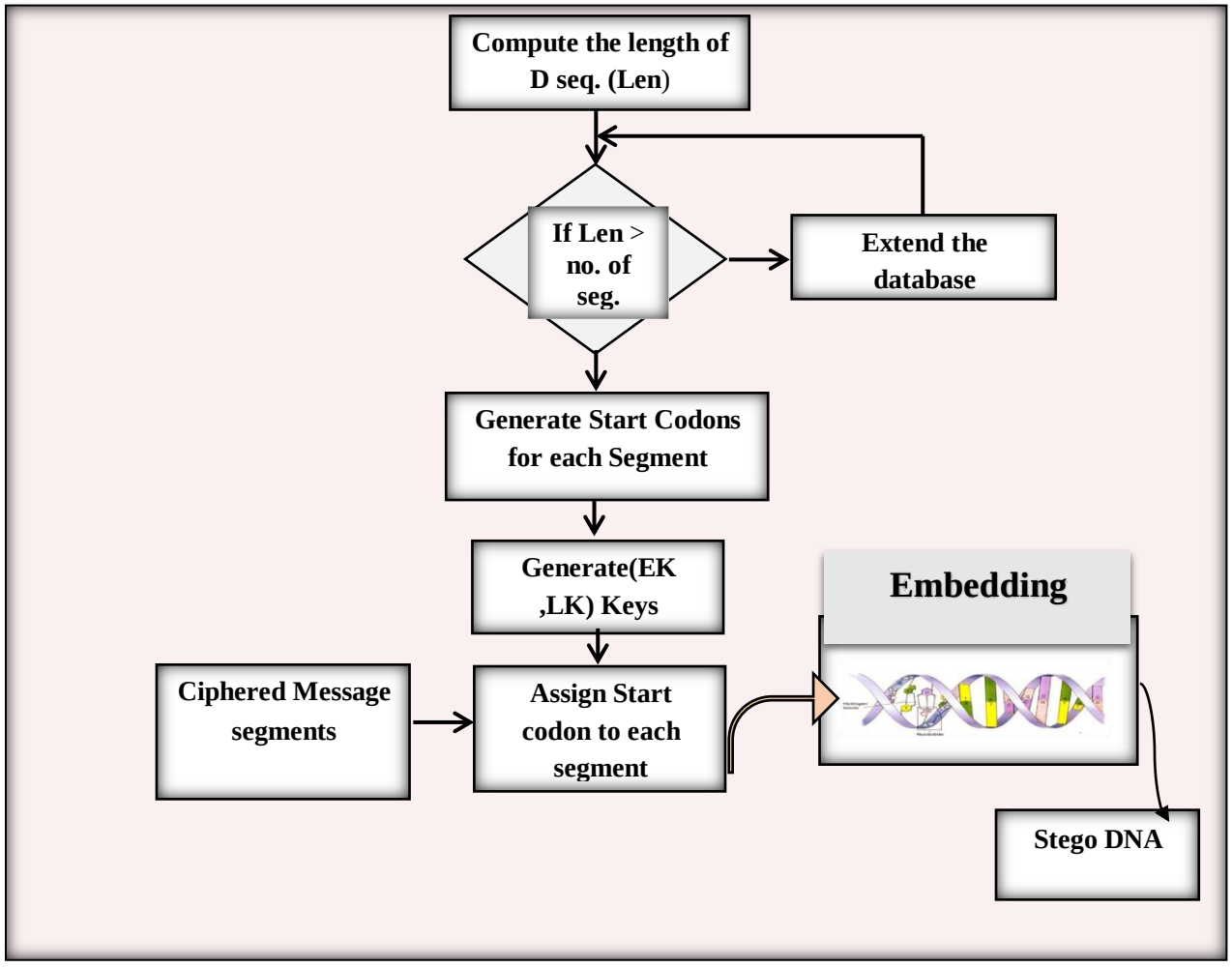

Figure (3.3) : Embedding stage

A- Database Checking Process

Testing the length of a selected DNA database is considered an important Process before embedding process, In order to know if the length is adequate to contain all the message segments or not. In case of not an extended process is needed to D vitamin. Algorithm (3.6) depict this extension.

Algorithm (3.6): Extending Database length

Input: length of selected Data Base (D), No of Msg segments
Output: adequate sequence length.

//Extended Data base processes.
For each new massage.
1: length =number of segments
2: If length > Data Base then
3: Factor of DB size = Int of (Msg length Require/ length of Data Base of D vitamin.
4: Extended length of Data Base sequence = Factor of DB size∗ lenght of Data Base of D vitamin.
5: Else
6: Embedding the cipher message in Data Base of D vitamin.

7: End

B-SNPs Complement Process

The resulted SNP will have complemented according to Table (3.3) such that each SNP has one complement according to DNA complementary rules [3].

Table (3.3) DNA complementary Rules [38]

DNA Base	Coding
A	**T**
T	**A**
C	**G**
G	**C**

C- Embedding Keys (Ek)and Location Keys (Lk) Generation Process

The generation of (EK) key depend on the number of ciphered message segments in order to generate a number of (EK) keys equal to the number of segments, for example if the message segmented to 5 segments, the (EK) will generates 5 numbers randomly (03,04,01,05,02). This key is used key as an index to determine the embedding location segment. While (LK) is generated within range (0~99) which considered as a measuring index to measure the distance between the message segments. Generating these keys explained in algorithm (3.7)

Algorithm (3.7): Embedding key(EK) and Location key(LK) Generation Process

Input : N: No. (Msg Seg) value ,L : length of D vitamin
Output: Ek and Lk

// EKey generation process
1: for i = 1: No. (Msg Seg) value
2: EK[i] = random set of (N)
3: end

// L Key generation process

4: for j = 0: L
5: LK[j] = random set of (L)
6: end

D- Embedding Process

The Embedding process is started whenever the DNA sequence is adequate and (EK) (LK) is available. For each segment, a suitable location will be selected within the database. The embedding process based substitution technique. Figure (3.4) explains the Embedding Process.

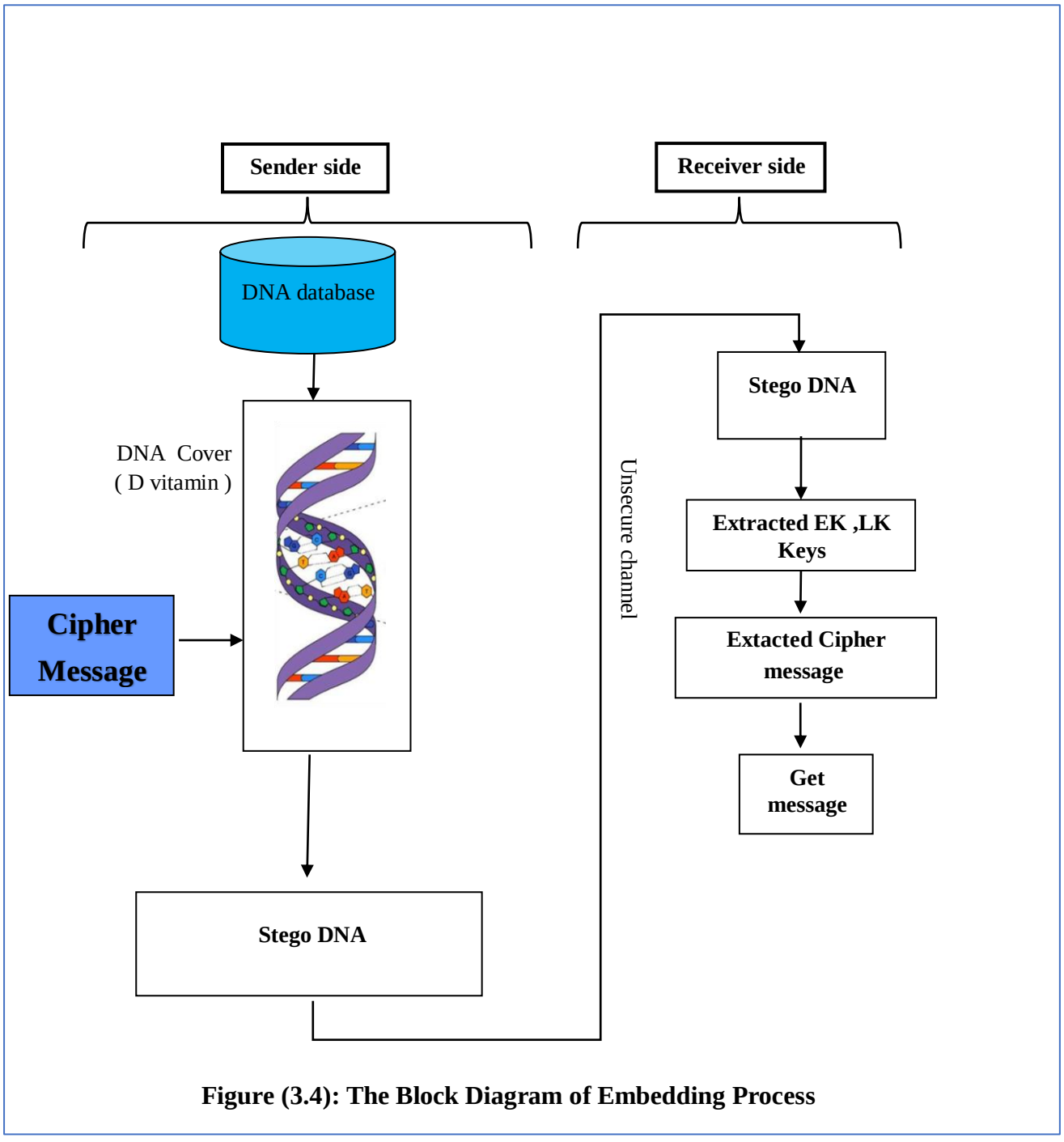

Figure (3.4): The Block Diagram of Embedding Process

3.4 Receiver Side Activities

At this side, the receiver has to extract the embedding message from the Stego DNA (D vitamin sequences). The receiver has to (Extract Embedding Message, Mutation Detection, and Message Retrieval) processes. Figure (3.5) illustrates this activity.

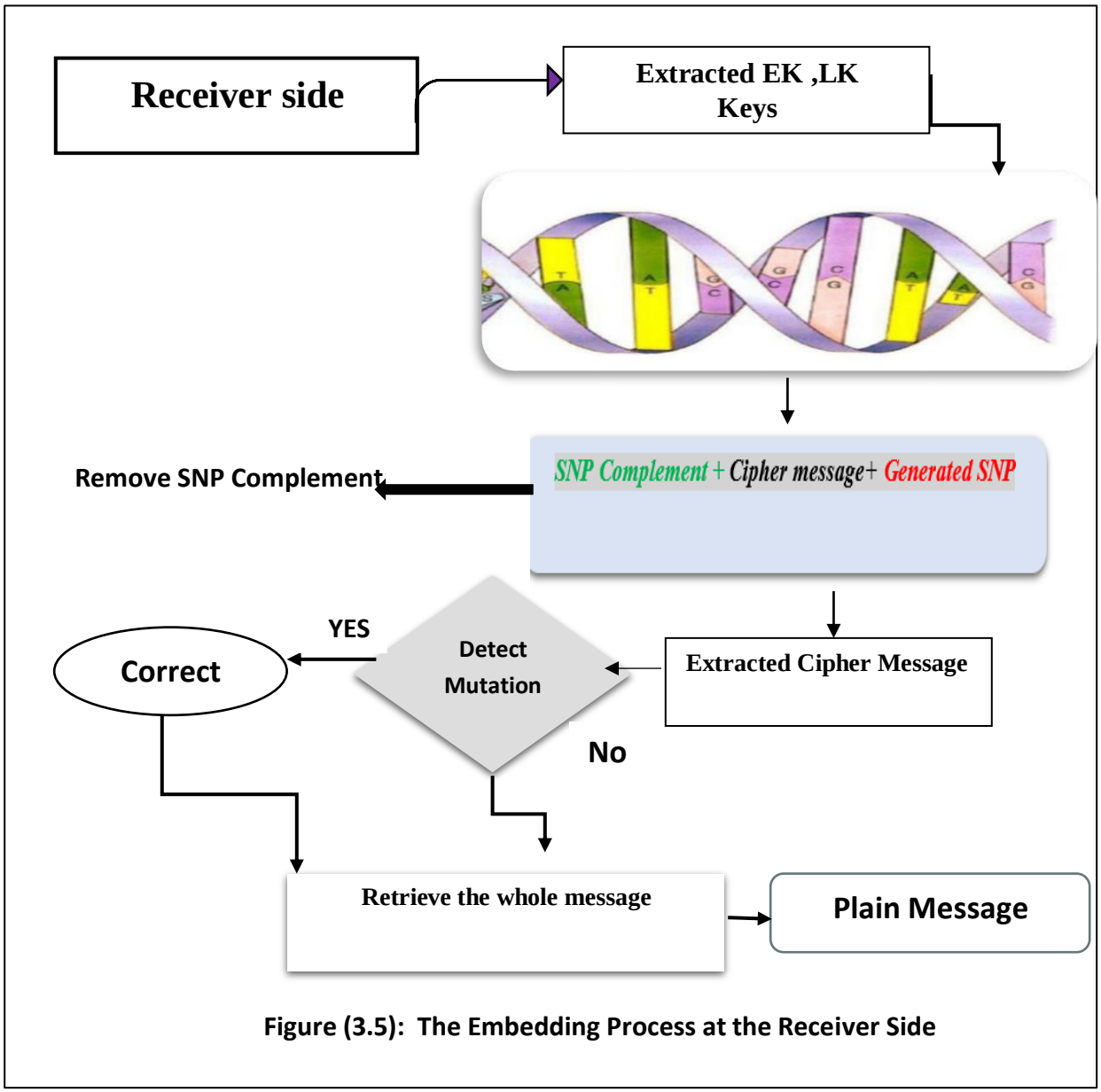

Figure (3.5): The Embedding Process at the Receiver Side

3.4.1 Extract the Embedding Message Process

At this point, the two keys (Ek, Lk) are retrieved from DNA cover and send it securely to the receiver to extract the message from D sequence,

Algorithm (3.8): Extract EK and LK keys

Input: L: length of D seq. ,N:number of segments
Output: Msg segments

```
// Extract the cipher message from D seq.
1:      let EK= N
2:    random set of (N)
3:   end if
//  L key generation process
4:  for j = 0: L
5:   L k = random set of (j)
6:  end //measuring key generation
```

3.4.2 Mutation Detection Process

To detect if there are a mutations exist, a comparison will be done of summation SNPs between the sent and received versions of message. As a result, the SNPs will notice if there is a mutation or not. The algorithm (3.9) depict this process.

Algorithm (3.9): Detect mutation

Input: re- assembled cipher message with SNPs

Output: Detect mutation.

1: S1= ∑ SNPs of cipher message
2: S2= ∑SNPs of Receiving message
3: if S1 <> S2 Then
4: mutation exists
correct The mutation
5: End if.

3.4.3 Message Retrieval Process

In this process, the SNPs and its complements are removed to get a pure codon to be converted to the original message characters. This conversion based on assigning each codon to its represented character. Algorithm (3.10) shown this process.

Algorithm (3.10): Decryption Process

Input : Cipher msg
Output: plain msg

// Extract The ciphered msg
1: Remove the SNPs. complement
2: Remove all SNPs
3: convert each codon (three bases) into character according to table (3.1)
4: the result is plain msg

Chapter Four

Results And Discussion

CHAPTER FOUR
RESULTS AND DISCUSSION

4.1 Introduction

This chapter introduces a discussion of the experimental work results of the proposed system. The experimental results were analyzed to clarify the results by some performance metrics such as cracking probability (CP), Hiding capacity, Modification rate, Payload and Information Entropy. The embedding process has adopted in a new method using SNPs technique.

4.2 Proposed System Implementation

The gained results from implementing the proposed system are demonstrated from:

4.2.1 Tables Generation Stage

The result of this stage are two base tables: DNA codons and DNA digitization based on algorithms (3.1),(3.2) . These Tables are static for all new message session. In this section, two messages with different lengths has been entered. The first message as shown in figure (4.1) and the second message will be explained in Appendix A.

The First Message:

One of the most interesting features of DNA is single nucleotide polymorphism (SNP)

Figure (4.1): Entered The First Message

Table (4.1): Codon's Table of The First Message

AAA(k)	TAA(G)	GAA(y)	'CAA(O)	CAT(M)
AAT(A)	TAT(E)	GAT(I)	AAT(A)	CAT(M)
TAG(Q)	AAG(B)	GAG(J)	CAG(4)	AAC(Z)
TAC(H)	GAC(L)	CAC(P)	ATA(R)	TTA(V)
ATT(S)	TTT(W)	GGT(a)	CAA(d)	GTA(Z)
CTT(e)	ATG(T)	TTG(X)	GTG(b)	CAT(f)
ATC(F)	TTC(U)	GTC(Y)	CTA(c)	AGA(i)
TGA(m)	GGA(q)	CTA(u)	AGT(g)	TGT(C)
GGT(o)	CAT(s)	TGG(l)	AGG(h)	GGG(p)
CGG(t)	GCA(n)	GCT(0)	AGC(9)	ACC(2)
AGA(j)	GGA(r)	ACA(w)	ATG(x)	ACT(3)
CCT(N)	CCA(5)	CCG(6)	CGA(7)	TTG(1)
TAC(.)	ATC ([)	TAA(])	GAA(()	CTC(8)
CCC())	TGC(?)	GAT(=)	TTC(&)	GTT(*)
ACT(y)	TCC(;)	GCT( )	GCA(,)	GCA(#)
TAC(})	TAA(D)	ACC({)		

4.3 Encryption Stage Results

During this Stage, the message will be encrypted by two level:

4.3.1 First Level Encryption

The results of this level come from Permutation Process.

- Permutation Process

This Process permute random swapping between the locations of message contents, so that the message becomes more randomness and incomprehensible. The result of permutation process shown in figures (4.3) and based on algorithm (3.3).

)PNS(msihpromylop editoelcun elgnis eht si AND fo serutaef gnitseretni tsom eht fo enO.

Figure (4.2) Permutation of The First Message

4.3.2 The Second Level Encryption

A- DNA Assignment Process

In This Process, each character, symbol, or number in the message will be assigned to a specific DNA codon , based on the generated codon table (3.1). the result of assignment is shown in figure (4.5).

CCCCACCCTATTCAGATTGAAGCTTGACCTAGAAGGGGG
GGAGGTTGAACTTGGGGTGGGGCTCTTCAAAGACGGGGT
CTTTGGCTACGGGCAGCTCTTTGGAGTGCAAGACCTGCT
CCTAGAGCTAATCAGAACGCTGCCGGTGCTCCTCTTGGA
CGGCGGGGTCTTGCCGCTAGTGCAAGACGGCCTCTTGGA
CTTCGGGCAAGAGCTCGGCCTGGTTGAGCTCTTAGGCGG
GCTGCCGGTGCTCTTGCACAA

Figure (4.3): DNA Assignment of first message

B- Segmentation Process

This process will result in a number of message segments. The result of this process shown in Table (4.5) and based on algorithm (3.4). Segmentation process will give more randomness when the cipher message hiding in Database.

Table (4.2): Segmentation Process

Message	The First Message
Number of segments	2 segments

C-DNA Message Form

Each segment will be arranged in 2D square array . Figures (4.7) and explain these Forms of the first message**.**

{CCC}	{CAC}	{CAG}	{ATT}	{GAA}	{GCT}	{TGA}
{CCT}	{AGA}	{AGG}	{GGG}	{GGA}	{GGT}	{TGA}
{ACT}	{TGG}	{GGT}	{GGG}	{GCT}	{CTT}	{CAA}
{AGA}	{CGG}	{GGT}	{CTT}	{TGG}	{CTA}	{CGG}
{GCA}	{GCT}	{CTT}	{TGG}	{AGT}	{GCA}	{AGA}
{CCT}	{GCT}	{CCT}	{AGA}	{GCT}	{AAT}	{CAG}
{AAC}	{GCT}	{GCC}	{GGT}	{GCT}	{CCT}	{CTT}

Figure (4.4.1) : The First Segment of DNA Form Related to the First Message

{AGT}	{GCA}	{AGA}	{CGG}	{CCT}	{CTT}	{GGA}
{CTT}	{CGG}	{GCA}	{AGA}	{GCT}	{CGG}	{CCT}
{GGT}	{TGA}	{GCT}	{CTT}	{AGG}	{CGG}	{GCT}
{GCC}	{GGT}	{GCT}	{CTT}	{GCA}	{CAA}	{GCA}

Figure (4.4.2): The Second Segment of DNA Form Related to the First Message

4.4 SNPs Generation Stage

This stage has four Processes (Digitize DNA base Process, Computation Process, SNPs values Generation and Assignment SNPs).

4.4.1 Digitize DNA base Process

DNA will be digitized according to the Digitization table (3.2). Figure (4.9) explains the result of this process.

{222}	{202}	{201}	{033}	{100}	{123}	{310}
{223}	{010}	{011}	{111}	{110}	{113}	{310}
{023}	{311}	{113}	{111}	{123}	{233}	{200}
{010}	{211}	{113}	{233}	{311}	{230}	{211}
{120}	{123}	{233}	{311}	{013}	{120}	{010}
{223}	{123}	{223}	{010}	{123}	{003}	{201}
{002}	{123}	{122}	{113}	{123}	{223}	{233}
{110}	{211}	{211}	{113}	{233}	{122}	{123}

{013}	{120}	{010}	{211}	{223}	{233}	{110}
{233}	{211}	{120}	{010}	{123}	{211}	{223}
{113}	{310}	{123}	{233}	{011}	{211}	{123}
{122}	{113}	{123}	{233}	{120}	{200}	{120}

Figure (4.5) : Digitization The first message

4.4.2 Computation Process

To assign a SNP for each rows and columns for each segment , Algorithm (3.5) applied for the message to get a numeric value that relates to a specific DNA base. The base will represent a specific SNP as shown in figure (4.11) .

{223}	{010}	{011}	{111}	{110}	{113}	{310}	{1}
{023}	{311}	{113}	{111}	{123}	{233}	{200}	{0}
{010}	{211}	{113}	{233}	{311}	{230}	{211}	{3}
{120}	{123}	{233}	{311}	{013}	{120}	{010}	{0}
{223}	{123}	{223}	{010}	{123}	{003}	{201}	{1}
{002}	{123}	{122}	{113}	{123}	{223}	{233}	{3}
{110}	{211}	{211}	{113}	{233}	{122}	{123}	{1}
{3}	{1}	{0}	{1}	{1}	{3}	{0}	{3}

{013}	{120}	{010}	{211}	{223}	{233}	{110}	{3}
{233}	{211}	{120}	{010}	{123}	{211}	{223}	{3}
{113}	{310}	{123}	{233}	{011}	{211}	{123}	{0}
{122}	{113}	{123}	{233}	{120}	{200}	{120}	{0}
{0}	{0}	{0}	{3}	{3}	{1}	{1}	{3}

Figure (4.6) : Computation Process of The first message

4.4.3 SNPs Assignment Process

SNPs will be assigned based on computation process as shown in Figures (4.13) and According to the resulted values, the SNP base is assigned

{GGT}	{GCT}	{CTT}	{TGA}	{GGT}	{CCT}	{GCT}	{T}
{ACT}	{GTG}	{GCT}	{CAA}	{CTT}	{CAA}	{AGA}	{T}
{CGA}	{GGT}	{GGA}	{GGG}	{GCT}	{CCT}	{AGA}	{T}
{GCT}	{ACT}	{CGG}	{AGA}	{GGA}	{CGG}	{CTA}	{G}
{CTT}	{CCT}	{GCT}	{GCC}	{GGT}	{GCT}	{GGA}	{A}
{CTT}	{ACT}	{GGT}	{TGG}	{GCT}	{CTT}	{TGG}	{A}
{GTG}	{CGG}	{GGT}	{CAA}	{GCT}	{AAT}	{GCT}	{A}
{A}	{A}	{T}	{T}	{A}	{G}	{G}	{T}

Figure (4.7.1) :SNPs Assignment of The First Message

{AGT}	{GCA}	{AGA}	{CGG}	{CCT}	{CTT}	{GGA}	{T}
{CTT}	{CGG}	{GCA}	{AGA}	{GCT}	{CGG}	{CCT}	{T}
{GGT}	{TGA}	{GCT}	{CTT}	{AGG}	{CGG}	{GCT}	{A}
{GCC}	{GGT}	{GCT}	{CTT}	{GCA}	{CAA}	{GCA}	{A}
{A}	{A}	{A}	{T}	{T}	{G}	{G}	{T}

Figure (4.7.2) :SNPs Assignment of The First Message

4.5 Embedding Process

The Embedding Process is as follows:

4.5.1 Database Examination Process

The length of the selected database is (70441) bases, so the length of first message and the second message do not need to double the size of the database.

4.5.2 Start Codon Generation

Start codon will be generated by giving the complementary to all SNPs in the last row of each segment. So the complement of SNP will act as a dynamic starting codons to get the integrity of cipher message .The Structure of SNPs complementary are shown in figure(4.8) below.

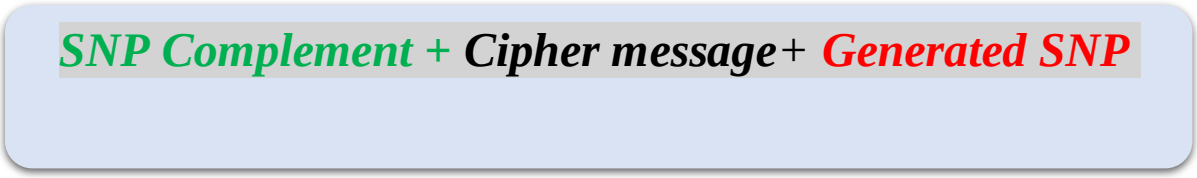

Figure (4.8): Embedding Message Structure

4.5.3 Embedding Key (EK) and Location Key (LK) Generation:

According to the number of segments to each message, The value EK for the first message is (2) and second message is (20), while the LK value for ranged between (0…Length of Database) generated random values. These keys create scattering hiding locations that change with each message

4.5.4 Embed Process

The Embedding Process depends on EK generated value such that for the first message EK is random values (01,02). While LK generated value represent the segment location. Figure (4.9) illustrates part of D vitamin of length (70441) bases and figure (4.10) illustrates Embed Process.

```
68041 aaaatgagtt tttatggggc tgaacgggga gaaaaggtca tcatcgattc tactttagaa
68101 tgagagtgtg aaatagacat ttgtaaatgt aaaactttta aggtatatca ttataactga
68161 aggagaaggt gccccaaaat gcaagatttt ccacaagatt cccagagaca ggaaaatcct
68221 ctggctggct aactggaagc atgtaggaga atccaagcga ggtcaacaga gaaggcagga
68281 atgtgtggca gatttagtga aagctagaga tatggcagcg aaaggatgta aacagtgcct
68341 gctgaatgat ttccaaagag aaaaaaagtt tgccagaagt ttgtcaagtc aaccaatgta
68401 gaaagctttg cttatggtaa taaaaatggc tcatacttat atagcactta ctttgttgca
68461 agtactgctg taaataaatg ctttatgcaa accaatttgc cttatcctta taaggacctt
68521 atgggagatg aatcattatt acccccattt gacagaaagg atagcttgag caatgccaca
68581 ctagcaaggg atgggatttg aaccttcagc agctaggttc agaagccaca aattaactgc
68641 tacattgtcc tgcttcctat tgagttgggg gacctgacag acgactgatg gtcttgctag
68701 ctctctccta gagaggagat aaaagaggtt cccattccta aagcaggccc tgagccagga
68761 aaattagagg tgctggacca aactgtgctc tactcccagg aagtgtgcag tcaatatatg
68821 acacctacgt gagaccctca aaaatgaaaa ccaaacagct actggcaaaa ctgtgtctgc
68881 cattagagat ggcggctgtg ccagtgacct ggaggattac aaatgactgc tgtgcagaaa
68941 caggactcct aaggggccca acttatgccg atgcactcca ttctgcttcc caaggaagtg
69001 gggtttatga tgaagggtag cattgctagg cacagtaaac aagaacacag cattgtgatc
69061 tgaaaataag gaaatcatgc cagctaatgt attgattgag gataagttgg cctggggatg
69121 tgattcactc taatttttca gaaacatctg aaaatatttc aaaccaaagg ctaaaatgtg
69181 tttcagtggg atgagatgga cttaggggaa ttggggttag aacttgaggg ttattttgtg
69241 aaacatgaag ggacttagag aaaggaaatc aacagctgca taaatgggca tgtctctggc
69301 tggagaaatg tggagaatgg agttctgata cactgttaga aggatcttat gtagcatttt
69361 tatagctgac ctagaagaac acaaaatttc caaggctgtg ttataatgcg cttttccagg
69421 taaaccaaga ggaatatacc ccaggaaggt tgcataatta ggatcaagtg ttttcaagtt
69481 ttcatattcc aagctttggt tctatgccta cactgttcaa tccagtagcc actagctaca
69541 tgtgagtatt taaatgaaat aaaggtaaac atctagcttg tcaaccgcac aagccacagt
69601 tccagtattt gataacctca gggctaccgt aagagacagt gcaaatacac aacattttct
69661 tccttttttc tttcttcttt ctttcttttt tctttctttt ttcttctttt tttttttga
69721 gacagagtct tgctctgtca cccaggctgg agtgcagtgg cacaatctcg gctcactgca
69781 acctctgcct cccagtttca aaccattctc ctgcctcagc ctcatgagta gctgggatta
69841 caggcacctg acaccatgcc tggctaagtt ttgtattttt agtagagaca gggtttcacc
69901 atgttgtcca ggctggtctt gaactcctga cctcaagtta tctgcccgcc tcagcctccc
69961 aaagtgctgg gattacaagc gtgacatttt catcatcgca gaatagtcta tggggcagca
70021 ctggtctaca caatgcattc ttatctggta ctaattgtga atgactccat gaggatgctg
70081 gcgtcatgtg cttctgttga tctgtagggc agaatggcca ctaacttgac atcatatgga
70141 agtgctatag ggaacatcct ccccttacaa tgggctatgc cacacctggg gtagttcgaa
70201 tgagtctgct tcttaaaaga gacataaagc aaaaacactg cacagaccat ggggttgata
70261 ggctcaaagc atcatgtggt ataaatagct cactggtgtg ctaggagtat tgattccttt
70321 agccctggag caagcaaaca gggcctgcca ggagtgacca cagcccttca atttccccag
70381 cttctaccag gctccttgca ggctgcctgt gcagtgcagg tcggtctgcc tgccccatgg
70441 tccctgcaga tgacaagaag gatggatgct gtctgacacc tccagcatgg ccaag
```

Figure (4.9) : Part of Vitamin D

agacatgagctgggctactgggaatggagccctggagccagatggattt
ggagtgaataactggtggtgccctactctttctagggcaggctgagtctgt
gtgcattcacttctcatggcattgcacaggactccttggagctgttctgttct
gttctggagcagtcaagatttctgagatgccataggcatcatgaggctaaa
agaacccccaaacctggaaaatacttgtgagtggtaatatatcagttgag
taaaggccacattaagggcgcgtacgtaaaagcaaacattgggtgtcag

TTTACCACCCCCACCAGATTGAAGCTTGAACCTAGAAGGGGGGGAGGTTGA
AACTTGGGGTGGGGCTCTTCAAAAGACGGGGTCTTTGGCTACGGAGCAGC
TCTTTGGAGTGCAAGATCCTGCTCCTAGAGCTAATCAGAAACGCTGCCGGTG
CTCCTCTTTGGACGGCGGGGTCTTGCCGCTG AATGGTG
tgacctaaccctgaccatgcacccttggaggtagaatcaccaaaaccatc
atgtatcccctgcagggttctgaggggctacttgtaagccattgatggtct
cgcaaatgtatttctgtgaatctcatgttccagattctaaatgaggggtcct
atggtgagagggacttgttctgtcattcagaaaggcaagattccatcaactc
cctggatggagtcttctgtcagttggctactcaaacaataaagtccgcctag
cccatgatctagggcaggatttttttttttttttttttgagacagggtctcactc
tgcctcccaggcctcaagcatagtggcatgatcatagctcactgtgaccta
accctgaccatgcacccttggaggtagaatcaccaaaaccatcatgtatcc

Figure (4.10): Embed Process of The First Message.

Figure (4.10) above is Embed Process of The first segments, For each segment of the message, black bases: DNA bases of segment1, red bases: SNPs of rows ,green bases:SNPs of columns , blue bases :SNPs complement. The same Embed process will be repeated for another segments.

4.6 Message Decryption

At this end, the receiver receives the DNA stego that contains the message with a table includes details about the processes applied to the message in order to inform the receiver can be retrieved the original message by reverse all the steps of the sender depend on the sending details. This stage consists of three Processes (Message Extraction, Detect Mutation and Message Retrieval).

4.6.1 Extract Embedding Message

Extracting the message depends on the random values of the keys (EK, LK) within the database.

4.6.2 Detect Mutation

At this stage, the message will be arranged, old SNPs will be removed, and new SNPs will be generated. The process of generating SNPs is a recall of the generation of SNPs in the Encryption stage. The receiver will compare the summation of old SNPs and new SNPs to detect mutation. Table (4.3) explains detect mutation technique. (Green bases are SNPs of rows, Red bases are SNPs of columns, Blue base is SNP has mutation).

Table (4.3): Mutation Detection Technique

DNA Sequences of segment1	Without Mutation	Detect Mutation
TGATTAATTAATGTGA	**TGATTAATTAATGTGA** Sum= 24	**TGATTAATTTATGTGA** Sum=27

4.6.3 Extracted Message

The third step is to retrieve the reverse message based on the codon table and convert the triple codons into letters or their corresponding script.

Figure (4.11): Extracted Message

4.7 Security Analysis

To measure the security performance of the proposed method using several measuring.

4.7.1 *Hiding capacity*

The capacity of any encrypted message is calculated using Equation (2.1). The proposed work achieves a high capacity because it is based on the substation method. Therefore, the capacity will represent the length of the DNA sequences of message.

4.7.2 *Cracking probability*

The Cracking probability is computed for encrypted message according to the equation (2.7). Cracking probability of The first message and The second message is (11e-1.821), these results clarify the difficulty of cracking the

required sequences. The Cracking would be difficult to detect due to the long DNA sequences within the NCBI database.

4.7.3 Payload

The value of Payload is zero because we are using a Substitution method that does not change the embedding database.

4.7.4 Information Entropy

The entropy of any encrypted message has been calculated using equation (2.8), the ideal value of entropy for an encrypted message should be (2).

4.7.5- Modification Rate

The percentage of change in the database is very small, due to the large size of the database.

4.7.6- Bit Error Rate (BER)

BER computed for encrypted message according to the equation (2.9). the ideal value of BER=1.

Table (4.4) and table (4.5) show The results .

Table (4.4): Performance measurement of first and second Messages

Message No.	Capacity	Cracking probability	Payload	Entropy	Modification rate	BER
Message1	2940	11e- 1.821	0	2	0.04	1
Message2	177	11e-1.821	0	2	0.003	1

Table (4.5) : Encryption and Decryption Time

Message No.	Encryption Time	Decryption Time
Message1	0.394ms	0.395 ms

Message2	**0.021 ms**	**0.022 ms**

4.8 Comparison

In this section, a comparison of the proposed work will be studied from the view of working principle and the security level of the work. Table (4.7) explains comparison between our proposed work and other Related works and Table (4.8) illustrates the comparison between our proposed work and Reference [20].

Table (4.7) : Comparison between our Proposed work and other References

Ref	Implementation Method	Blindness Applicable?	Capacity	Cracking probability	Payload	Entropy	BER
[13]		yes	Number of DNA Bases	Very low	0	-	-
[14]		yes	2	Low	-	-	-
[17]	Substitution based method	yes	0.82	Low	-	-	-
[19]	The neural network algorithm and DNA bases	-	High	Very low	-	-	-
[38]	Substitution Method	yes	Length of DNA reference	Low	0	-	-
[38]	Complementary Substitution Method	yes	2*length of DNA reference	Low	Length of DNA reference	-	-

Proposed work	Substitution Method	yes	Length of Message DNA	Very low	0	2	1

Table (4.8) : Compare between our proposed work and Reference [20]

	Table of codons	**Capacity of Table**	**Message Structure**	**DNA-Steganography**	**Mutation**
Ref [20]	lookup tables are static, and using the same table in every message transition session.	Limited capacity Only (characters , number).	Segment+SNPs	used adjacent or sequential locations in the DNA sequence (LSBase).	Block checksum fails in addition, deletion and swapping cases.
Our Proposed System	Lookup tables are dynamic for each new message session.	Characters , number and Special charcter.	Using SNPscomplement + ciphered message+ SNP.	Based on EK, LK Keys. That give random location in DB.	Enhance the performance By using Summation Algorithm

Chapter Five

Conclusion and future Works

CHAPTER FIVE
CONCLUSION AND FUTURE WORKS

5.1 Conclusion

Through the implementation of the proposed system and the achieved results, several indicators have been deduced to illustrate the strength of the system and the security level. These indicators:

The suggested approach increases the level of data security by utilizing encryption at the first level and a new DNA-steganography method called SNP at the second level to embed data into a dedicated DNA data set.

1. Using a dynamic coding table randomly generated at each transition instead of using static table decrease the risk of compromising the proposed method.
2. Scattering the hiding locations all over the sequence will make deducing the secret message from the covering medium harder for an attacker. using a special keys called embedding and distance for hiding a message.
3. Generating SNPs will give more randomness, so it is difficult to distinguish the used codon is 3 bases or 4 bases.
4. The transmission of the fake DNA sequence from the sender to a receiver over an untrusted channel may encounter different types of obstacles. Some of these obstacles are mutations, transition errors, and message modification by a third party. To preserve the integrity of secret message a dedicated DNA sequence used to guarantee the integrity of the fake DNA sequence and will offer more locations for data hiding and data checking.
5. DNA has proved it can be the optimal medium for data hiding and transmission.

5.2 Future Work Suggestions

Several suggestions are adopted here to extend the current proposal as a future work in this thesis:

1- Chaotic map can be applied with encryption technique to increase the message randomness and for generating SNPs.

2- Applying zig-zag computation method for generating SNPs.

3- Another approach is to “watermark” the engineered cells to be used to protect the intellectual property of engineered cells.

References

References

[1] Liu, Q., Yang, K., Xie, J., & Sun, Y. (2021). DNA-based molecular computing, storage, and communications. *IEEE Internet of Things Journal*, *9*(2), 897-915.

[2] Zebari, N. A., Zebari, D. A., Zeebaree, D. Q., & Saeed, J. N. (2021). Significant features for steganography techniques using deoxyribonucleic acid: a review. *Indonesian Journal of Electrical Engineering and Computer Science*, 21(1), 338-347.

[3] Terkawi, N. S., Berriche, L., Alamar, A. A., Ibrahim, M. A., & Alsaffar, W. S. (2021). Comparative Study of Three DNA-based Information Hiding Methods. International Journal of Computer Science and Security (IJCSS), 15(2), 45-59.

[4] Simonini. Benedetta, (2020). Cryptography and Data Security: From Cryptographic Techniques to Data Protection.

[5] Kadhum. Rafal Najeh, Ali. Nada Hussein M. 2022). Using steganography techniques for implicit authentication to enhance sensitive data hiding, Int. J. Nonlinear Anal.1, pp: 3973-3983, ISSN: 2008-6822 (electronic), available at : http://dx.doi.org/10.22075/ijnaa.2022.6211.

[6] Narayana, V. L., & Kumar, N. A. (2018). Different techniques for hiding the text information using text steganography techniques: A survey. Ingénierie des Systèmes d'Information, 23(6).

[7] Hammad, B. T., Sagheer, A. M., Ahmed, I. T., & Jamil, N. (2020). A comparative review on symmetric and asymmetric DNA-based cryptography. Bulletin of Electrical Engineering and Informatics, 9(6), 2484-2491.

[8] Y. Niu, K. Zhao, X. Zhang, and G. Cui, (2020),"Review on DNA cryptography," in Bio-inspired Computing: Theories and Applications. Singapore: Springer Singapore, 2020, pp. 134–148

References

[9] Hassan, S., Muztaba, M. A., Hossain, M. S., & Narman, H. S. (2022, October). A Hybrid Encryption Technique based on DNA Cryptography and Steganography. In 2022 IEEE 13th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON) (pp. 0501-0508). IEEE.

[10] H Mohammed, M., H Ali, B., & I Taloba, A. (2019). Self-adaptive dna-based steganography using neural networks. *Information Sciences Letters*, *8*(1), 2.

[11] Nabi, S. H., Sarosh, P., Parah, S. A., & Mohiuddin Bhat, G. (2021). Information Embedding Using DNA Sequences for Covert Communication. In *Multimedia Security* (pp. 111-129). Springer, Singapore.

[12] Zebari, N. A., Zebari, D. A., Zeebaree, D. Q., & Saeed, J. N. (2021). Significant features for steganography techniques using deoxyribonucleic acid: a review. Indonesian Journal of Electrical Engineering and Computer Science, 21(1), 338-347.

[13] Mohammed. Marghny H., Abdel-Razeq. Alaa, (2020), DNA-based steganography using genetic algorithm, Information Sciences Letters, Volume 9, Issue 3.

[14] Khalifa. Amal,(2021), A Secure Steganographic Channel Using DNA Sequence Data and a Bio-Inspired XOR Cipher, information (MDPI), vol: 12,issue: 253, available at: https://doi.org/10.3390/info12060253

[15] Nabi, S. H., Sarosh, P., Parah, S. A., & Mohiuddin Bhat, G. (2021). Information Embedding Using DNA Sequences for Covert Communication. Multimedia Security, Springer, Singapore, pp: 111-129.

[16] Khalifa. Amal, (2021), Hiding Information in DNA Sequence Data using Open Reading Frame Guided Splicing, *Advances in Science,* Technology and5Engineering Systems Journal Vol. 6, No. 3, pp:164-171.

[17] El-deeb. Amany E., El-Sisi Ashraf B., and Youssef. Anas, (2021), A Substitution-Based Method for Data Hiding in DNA Sequences, International Journal of Computers and Information, doi:10.21608/IJCI.2021.56184.1037.

References

[18] Saha. Partha, Pinky. Lubna Yasmin, et al., (2019), Higher Payload Capacity in DNA Steganography using Balanced Tree Data Structure, International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878 (Online), Volume-8 Issue-4.

[19] Mohammed. Marghny H., Ali Botheina H., Taloba. Ahmed I., (2019), Self-adaptive DNA-based Steganography Using Neural Networks, Information Sciences Letters Volume 8 Issue 1.

[20] Al-kateeb. Zeena N., Jader. Melad, (2020), Encryption and hiding text using DNA coding and hyperchaotic system, Indonesian Journal of Electrical Engineering and Computer Science Vol. 19, No. 2, pp. 766~774 ISSN: 2502-4752, DOI: 10.11591/ijeecs.v19.i2.pp766-774.

[21] Na, D. (2020). DNA steganography: hiding undetectable secret messages within the single nucleotide polymorphisms of a genome and detecting mutation-induced errors. Microbial cell factories, 19(1), 1-9.

[22] Harjito, B., UYS, D. S., & Rahutomo, F. (2023). Analysis and Implementation of Steganography Using Playfair Techniques and DNA Substitution To Improve Message Security. Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika, 9(1).

[23] Ettiyan, R., & Geetha, V. (2023). A hybrid logistic DNA-based encryption system for securing the Internet of Things patient monitoring systems. Healthcare Analytics, 3, 100149.

[24] Sadkhan, S. B. (2021, June). Information security based on DNA-importance and future trends. In *2021 International Conference on Communication & Information Technology (ICICT)* (pp. 310-314). IEEE.

[25] Ma, Q., Zhang, C., Zhang, M., Han, D., & Tan, W. (2021). DNA Computing: Principle, Construction, and Applications in Intelligent Diagnostics. *Small Structures*, *2*(11), 2100051.

References

[26] Iliyasu, M. A., Abisoye, O. A., Bashir, S. A., & Ojeniyi, J. A. (2021, February). A review of DNA cryptograhic approaches. In *2020 IEEE 2nd International Conference on Cyberspac (CYBER NIGERIA)* (pp. 66-72). IEEE.

[27] **Kumari.** Priya, (2022), What is DNA Computing and Why is it Important, available at: https://datafloq.com/read/what-dna-computing-important/

[28] Kolate, V., & Joshi, R. B. (2021). An information security using DNA cryptography along with AES algorithm. Turkish Journal of Computer and Mathematics Education, 12(1S), 183-192.

[29] **Kumari.** Priya, (2022), What is DNA Computing and Why is it Important, available at: https://datafloq.com/read/what-dna-computing-important/ 3/2/2022.

[30] Kolate, V., & Joshi, R. B. (2021). An information security using DNA cryptography along with AES algorithm. *Turkish Journal of Computer and Mathematics Education*, *12*(1S), 183-192.

[31] Yadav. Kusum, (2020), Structure and Properties of Nucleic Acids, available at: lkouniv.ac.in·https://www.lkouniv.ac.in

[32] Medline Plus, (2021), What is DNA?, U.S. National Library of Medicine National Institutes of Health Department of Health & Human Services, available at: **https://medlineplus.gov/genetics/understanding/basics/dna** 2/10/2022.

[33] Singh. Gambhir, Yadav. Rakesh Kumar, (2019), DNA Based Cryptography Techniques with Applications and Limitations, International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249-8958 (Online), Vol:8, Issue:6.

[34] Mondal, M., & Ray, K. S. (2023). Review on DNA Cryptography. International Journal of Bioinformatics and Intelligent Computing, 2(1), 44-72.

[35] Padmanabhan. Abhishek Sharma, Sapna. S., (2022), Secure Image Transmission Scheme based on DNA Sequences, International Journal of

Engineering Trends and Technology, Vol: 70 Issue: 9, 194-206, ISSN: 2231 – 5381 / https://doi.org/10.14445/22315381/IJETT-V70I9P220.

[36] Hamed, G., Marey, M., El-Sayed, S. A., & Tolba, M. F. (2015, November). Hybrid technique for steganography-based on DNA with n-bits binary coding rule. In 2015 7th International Conference of Soft Computing and Pattern Recognition (SoCPaR) (pp. 95-102). IEEE.

[37] Khalifa, A., & Hamad, S. (2015). Hiding secret information in dna sequences using silent mutations. British Journal of Mathematics & Computer Science, 11(5), 1.

[38] Suliman. Nisreen et al., (2021), Comparative Study of Three DNA-based Information Hiding Methods**,** International Journal of Computer Science and Security (IJCSS), Volume (15) : Issue (2).

[39] Alsaffar. Qusay S., (2022), An encryption by using DNA algorithm for hiding a compressed message in Image**,** Wasit Journal of Engineering Sciences, Doi: 10.31185/ejuow.Vol:10, Issue:1.249.

[40] Al-Harbi . O.A., Alahmadi. W.E., and Aljahdali. A.O., (2020)"Security analysis of DNA based steganography techniques" SN Applied Sciences, 2 (2).

[41] Alhabeeb. Omar Haitham, Fauzi. Fariza, Sulaiman. Rossilawati, " A Review of Modern DNA-based Steganography Approaches, International Journal of Advanced Computer Science and Applications, Vol. 12, No. 10, 2021.

[42] Abed, A., & Belzile, F. (2019). Comparing single-SNP, multi-SNP, and haplotype-based approaches in association studies for major traits in barley. The Plant Genome, 12(3), 190036.

[43] https://www.britannica.com/science/single-nucleotide-polymorphism. 15/9/2022.

[44] Gunter. Chris, (2023), Single Nucleotide Polymorphisms (SNPS), available at: https://www.genome.gov/genetics-glossary/single-nucleotide-polymorphisms

[45]-Making SNPs make Sense available at: https://learn.genetics.utah.edu/content/precision/snips/. 3/5/2022.

References

[46] Singh, B., & Nath, S. K. (2019). Identification of proteins interacting with single nucleotide polymorphisms (SNPs) by DNA pull-down assay. Electrophoretic Separation of Proteins: Methods and Protocols, 355-362.

[47] Zook, J. M., McDaniel, J., Parikh, H., Heaton, H., Irvine, S. A., Trigg, L., ... & Salit, M. (2018). Reproducible integration of multiple sequencing datasets to form high-confidence SNP, indel, and reference calls for five human genome reference materials. *BioRxiv*, 281006.

[48] Al-Mousawi, H. T. M. (2022). Remarkable association of the highly frequent rs1801133 snp of mthfr Gene with growth hormone deficiency in children. Web of Scientist: International Scientific Research Journal, 3(3), 114-124.

[49] Barrett, T., Wilhite, S. E., Ledoux, P., Evangelista, C., Kim, I. F., Tomashevsky, M., ... & Soboleva, A. (2012). NCBI GEO: archive for functional genomics data sets—update. Nucleic acids research, 41(D1), D991-D995.

[50] Sayers, E. W., Beck, J., Bolton, E. E., Bourexis, D., Brister, J. R., Canese, K., ... & Sherry, S. T. (2021). Database resources of the national center for biotechnology information. Nucleic acids research, 49(D1), D10.

[51] Database resources of the national center for biotechnology information." Nucleic acids research 46, no. D1 (2018): D8-D13.

[52] Fathi, N., Ahmadian, E., Shahi, S., Roshangar, L., Khan, H., Kouhsoltani, M., ... & Sharifi, S. (2019). Role of vitamin D and vitamin D receptor (VDR) in oral cancer. Biomedicine & Pharmacotherapy, 109, 391-401.

[53] Wang, Z., Huang, C., Lv, H., Zhang, M., & Li, X. (2020). In silico analysis and high-risk pathogenic phenotype predictions of non-synonymous single nucleotide polymorphisms in human Crystallin beta A4 gene associated with congenital cataract. Plos one, 15(1), e0227859.

[54] The National Center for Biotechnology Information (NCBI) on November https://blast.ncbi.nlm.nih/Blast.cgi.

References

[55] Hamed, G., Marey, M., Amin, S. E. S., & Tolba, M. F. (2018). Hybrid, randomized and high capacity conservative mutations DNA-based steganography for large sized data. Biosystems, 167, 47-61.

[56] Malathi, P., Manoaj, M., Manoj, R., Raghavan, V., & Vinodhini, R. E. (2017). Highly improved DNA based steganography. *Procedia Computer Science*, *115*, 651-659.

[57] Eskov, V. M., Eskov, V. V., Vochmina, Y. V., Gorbunov, D. V., & Ilyashenko, L. K. (2017). Shannon entropy in the research on stationary regimes and the evolution of complexity. Moscow university physics bulletin, 72, 309-317.

[58] Hamad, S., Elhadad, A., & Khalifa, A. (2017). DNA watermarking using Codon Postfix technique. IEEE/ACM transactions on computational biology and bioinformatics, 15(5), 1605-1610.

[59] Ding, J., & Schmidt, D. (2005, June). Rainbow, a new multivariable polynomial signature scheme. In *ACNS* (Vol. 5, pp. 164-175).

[60] Perlner, R., & Smith-Tone, D. (2020). Rainbow band separation is better than we thought. Cryptology ePrint Archive.

Appendixes

Appendix A : The Second Message

The transmission of sensitive information through the Internet faces high risks. Therefore, exchanging messages between senders and receivers is required to be in a confidential manner to avoid attacks. Sensitive data protection from unauthorized access is provided by two major techniques; steganography and cryptography [1]. Cryptography is a technique for preventing third parties from reading a secret message by converting it to an encrypted format which is incomprehensible for intruders. Some of the methods applied in the encryption are Playfair, Rivest Shamir Adleman (RSA), and Advanced Encryption Standard (AES) [2]. Steganography is a technique of hiding a secret message inside a cover message making it unnoticeable to any illegal read. The cover media could be text [3], image [4] [5][6], audio [7], video [8], and the Deoxyribonucleic Acid (DNA) sequence [9]. A double layer of security is provided by some

Figure (1): Entered The second Message

The generated DNA codons for this message are clarified in table (1) .The resulted table contains: alphabets (upper case, lower case), numbers, and special characters.

Table (1): DNA Codons of the second message

AAA(C)	TAA(G)	CAA(O)	AAT(A)	TAT(E)
GAA(K)	GAT(I)	CAT(M)	AAG(B)	TAG(F)
GAG(J)	CAG(N)	TAG(H)	AAC(D)	GAC(L)
ATA(R)	TTA(V)	GTA(Z)	CAA(d)	ATT(S)
TTT(W)	GGG(a)	CTT(e)	ATG(T)	TTG(X)
GTG(b)	ATC(Q)	CAT(f)	TTC(U)	GTC(Y)
CTA(c)	AGA(i)	GGA(q)	TGA(m)	CTA(u)
CTA(h)	TGG(l)	CGG(t)	'GGG(p)'	GCA(n)
GCT(0)	AGC(1)	ACC(2)	ACT(3)	CCT(4)
AGA(j)	GGA(r)	ATG(x)	ACA(w)	CCA(5)
CCG(6)	CGA(7)	CTC(8)	TTG(9)	ACG(.)
TAA(])	ATC(])	GAA(()	CCC())	ACC({)
TAC(})	TAA(z)	TGC(?)	GCA(#)	GCA(,)
GCT( )	TTA(y)	GAT(=)	GTT(*)	TTC(&)

Appendix B :

1-The First Level Encryption

A-Permutation Process

. yhpargotpyrc htiw yhpargonagets enibmoc taht sehcaorppa

emos yb dedivorp si ytiruces fo reyal elbuod A .]9[ecneuqes)AND(dicA cielcunobiryxoeDeht dna ,]8[oediv ,]7[oidua ,]6[]5[ ]4[egami ,]3[txet eb dluoc aidem revoc ehT .daer lagelli

yna ot elbaecitonnu ti gnikam egassem revoc a edisni egassem terces a gnidih fo euqinhcet a si yhpargonagetS .]2[ )SEA(dradnatS noitpyrcnE decnavdA dna ,)ASR(nameldA rimahS tseviR ,riafyalP era noitpyrcne eht ni deilppa sdohtem eht fo emoS .sredurtni rof elbisneherpmocnisi hcihw tamrof detpyrcne na ot ti gnitrevnoc yb egassem terces a gnidaer morf seitrap driht gnitneverp rof euqinhcet a si yhpargotpyrC .]1[yhpargotpyrc dna yhpargonagets ;seuqinhcet rojam owt yb dedivorp si ssecca dezirohtuanu morf noitcetorp atad evitisneS .skcatta diova ot rennam laitnedifnoc a ni eb ot deriuqer si sreviecer dna srednes neewteb segassem gnignahcxe ,eroferehT .sksir hgih secaf tenretnI eht hguorht noitamrofni evitisnes fo noissimsnart ehT

Figure (2): Permutation Process

Appendix C :

2-The second Level Encryption

A-DNA Assignment Process

ACGTTACTAGGGGGGGTGGA AGTGGTCGGGGGACTGGACTA

GCTAGGCGGAGAACAGCTACT AGGGGGGGTGGAAGTGGTGCA

GGTAGTCTTCGGCCTGCTCTT GCAAGAGTGTGAGGTCTAGCT

CGGGGTAGGCGGGCTCCTCTT AGGCTAGGTGGTGGAGGGGGG

GGTGCTCTTTGAGGTCCTGCT ACTGTGGCTCAACTTCAAAGA

CGAGGTGGAGGGGCTCCTAGA GCTACTCGGAGAGGACGGCTA

CTTCCTGCTGCCGGTGCTGGA CTTACTGGTTGGGCTCTTTGG

GTGCGGGGTCAAGCTAATGCT TACATCTTGTAAGCTCTTCTA

GCACTTCGGGGACTTCCTGCT CCCAATCAGAACGAAGCTCAA

AGACTAAATGCTCTAAGACTT TGGCTACGGGCAGGTGTGAGA

GGAACTATGGGTCTTAACGCT CTTAGGCGGGCTCAAGCAGGT

Figure (3): Part of DNA Assignment of the second message

B-Segmentation Process

Table (4.2): Segmentation Process

Message	The Message
Number of segments	2 segments

C-DNA Message Form

{TAC}	{GCT}	{ACT}	{AGG}	{GGG}	{GGT}	{GGA}
{AGT}	{GGT}	{CGG}	{GGG}	{ACT}	{GGA}	{CTA}
{GCT}	{AGG}	{CGG}	{AGA}	{ACA}	{GCT}	{ACT}
{AGG}	{GGG}	{GGT}	{GGA}	{AGT}	{GGT}	{GCA}
{GGT}	{AGT}	{CTT}	{CGG}	{CCT}	{GCT}	{CTT}
{GCA}	{AGA}	{GTG}	{TGA}	{GGT}	{CTA}	{GCT}
{CGG}	{GGT}	{AGG}	{CGG}	{GCT}	{CCT}	{CTT}
{AGG}	{CTA}	{GGT}	{GGT}	{GGA}	{GGG}	{GGG}

Figure (4.1) : Segment1 of The second message

{GGT}	{GCT}	{CTT}	{TGA}	{GGT}	{CCT}	{GCT}
{ACT}	{GTG}	{GCT}	{CAA}	{CTT}	{CAA}	{AGA}
{CGA}	{GGT}	{GGA}	{GGG}	{GCT}	{CCT}	{AGA}
{GCT}	{ACT}	{CGG}	{AGA}	{GGA}	{CGG}	{CTA}
{CTT}	{CCT}	{GCT}	{GCC}	{GGT}	{GCT}	{GGA}
{CTT}	{ACT}	{GGT}	{TGG}	{GCT}	{CTT}	{TGG}
{GTG}	{CGG}	{GGT}	{CAA}	{GCT}	{AAT}	{GCT}

Figure(4.2) : Segment2 of The second message

{TAC}	{ATC}	{TTG}	{TAA}	{GCT}	{CTT}	{CTA}
{GCA}	{CTT}	{CGG}	{GGA}	{CTT}	{CCT}	{GCT}
{CCC}	{AAT}	{CAG}	{AAC}	{GAA}	{GCT}	{CAA}
{AGA}	{CTA}	{AAT}	{GCT}	{CTA}	{AGA}	{CTT}
{TGG}	{CTA}	{CGG}	{GCA}	{GGT}	{GTG}	{AGA}
{GGA}	{ACT}	{ATG}	{GGT}	{CTT}	{AAC}	{GCT}
{CTT}	{AGG}	{CGG}	{GCT}	{CAA}	{GCA}	{GGT}

Figure(4.3) : Segment3 of The second Message

{GCT}	{GCA}	{ATC}	{CTC}	{TAA}	{GCT}	{GGT}
{CTT}	{CAA}	{AGA}	{CGA}	{GCT}	{GCA}	{ATC}
{CGA}	{TAA}	{GCT}	{GGT}	{AGA}	{CAA}	{CGG}
{GGT}	{GCT}	{GCA}	{ATC}	{CCG}	{TAA}	{ATC}
{CCA}	{TAA}	{GCT}	{ATC}	{CCT}	{TAA}	{GCT}
{CTT}	{AGT}	{GGT}	{TGA}	{AGA}	{GCT}	{GCA}
{ATC}	{ACT}	{TAA}	{GCT}	{CGG}	{ATG}	{CTT}

Figure (4.4) : Segment4 of The second message

{CGG}	{GCT}	{CTT}	{GTG}	{GCT}	{CAA}	{TGG}
{CGG}	{GGT}	{CTA}	{GCT}	{GGT}	{AGA}	{CAA}
{CTT}	{TGA}	{GCT}	{GGA}	{CTT}	{CGA}	{GGT}
{CTA}	{GCT}	{CTT}	{AGG}	{ATG}	{GCT}	{TAC}
{CAA}	{GGT}	{CTT}	{GGA}	{GCT}	{TGG}	{GGT}
{AGT}	{CTT}	{TGG}	{TGG}	{AGA}	{GCT}	{ACT}
{GCA}	{GGT}	{GCT}	{GGT}	{CGG}	{GCT}	{CTT}

Figure(4.5) : Segment5 of The second Message

{TGG}	{GTG}	{GGT}	{CTT}	{CTA}	{AGA}	{CGG}
{GGT}	{GCA}	{GCA}	{CGG}	{GCT}	{CGG}	{AGA}
{GCT}	{AGT}	{GCA}	{AGA}	{TGT}	{GGT}	{TGA}
{GCT}	{CTT}	{AGT}	{GGT}	{CCT}	{CCT}	{CTT}
{TGA}	{GCT}	{GGA}	{CTT}	{CGA}	{GGT}	{CTA}
{GCT}	{GGT}	{GCT}	{CTT}	{CAA}	{AGA}	{CCT}
{GCA}	{AGA}	{GCT}	{CTT}	{AGT}	{GGT}	{CCT}

Figure (4.6) : Segment6 of The second Message

{CCT}	{CTT}	{TGA}	{GCT}	{CGG}	{CTT}	{GGA}
{CTA}	{CTT}	{CCT}	{GCT}	{GGT}	{GCT}	{AGT}
{GCA}	{AGA}	{CAA}	{AGA}	{AGG}	{GCT}	{GCC}
{GGT}	{GCT}	{CTT}	{CGG}	{GGA}	{AGA}	{GCA}
{AGG}	{CTA}	{CTT}	{CGG}	{GCT}	{GGT}	{GCT}
{CCT}	{AGA}	{GCT}	{ACT}	{AGG}	{GGG}	{GGT}
{GGA}	{AGT}	{GGT}	{GCA}	{GGT}	{AGT}	{CTT}

Figure (4.7) : Segment7 of The second message

{CGG}	{ATT}	{GCT}	{TAC}	{ATC}	{ACC}	{TAA}
{GCT}	{CCC}	{ATT}	{TAT}	{AAT}	{GAA}	{GCT}
{CAA}	{GGA}	{GGT}	{CAA}	{GCA}	{GGT}	{CGG}
{ATT}	{GCT}	{GCA}	{GGT}	{AGA}	{CGG}	{GGG}
{ACT}	{GGA}	{CTA}	{GCA}	{TAT}	{GCT}	{CAA}
{CTT}	{CTA}	{GCA}	{GGT}	{CGA}	{CAA}	{AAT}
{GCT}	{CAA}	{GCA}	{GGT}	{GCT}	{GCA}	{CCC}

Figure (4.8) : Segment8 of The Second Message

{AAT}	{ATT}	{ATA}	{GAA}	{GCT}	{GCA}	{GGT}
{TGA}	{CTT}	{TGG}	{CAA}	{AAT}	{GCT}	{GGA}
{AGA}	{TGA}	{GGT}	{AGG}	{ATT}	{GCT}	{CGG}
{CCT}	{CTT}	{CGA}	{AGA}	{ATA}	{GCT}	{GCA}
{GGA}	{AGA}	{GGT}	{GCC}	{ACT}	{GGT}	{TGG}
{CAC}	{GCT}	{CTT}	{GGA}	{GGT}	{GCT}	{GCA}
{GGT}	{AGA}	{CGG}	{GGG}	{ACT}	{GGA}	{CTA}

Figure (4.9) : Segment9 of The Second Message

{GCA}	{CTT}	{GCT}	{CTT}	{AGG}	{CGG}	{GCT}
{GCA}	{AGA}	{GCT}	{CAA}	{CTT}	{AGA}	{TGG}
{GGG}	{GGG}	{GGT}	{GCT}	{CCT}	{CAA}	{GGT}
{AGG}	{CGG}	{CTT}	{TGA}	{GCT}	{CTT}	{AGG}
{CGG}	{GCT}	{GCC}	{GGT}	{GCT}	{CTT}	{TGA}
{GGT}	{ATT}	{GCT}	{TAC}	{CCT}	{GGA}	{CTT}
{CAA}	{CGG}	{GGA}	{CGG}	{GCA}	{AGA}	{GCT}

Figure (4.10) : Segment10 of The Second Message

{GGA}	{GGT}	{GCC}	{GCT}	{CTT}	{TGG}	{GTG}
{AGA}	{CCT}	{GCA}	{CTT}	{AGG}	{CTT}	{GGA}
{GGG}	{TGA}	{GGT}	{CTA}	{GCA}	{AGA}	{GCT}
{CCT}	{AGA}	{GCT}	{AGG}	{CTA}	{AGA}	{AGG}
{ACA}	{GCT}	{CGG}	{GGT}	{TGA}	{GGA}	{GGT}
{GCC}	{GCT}	{CAA}	{CTT}	{CGG}	{GGG}	{ACT}
{GGA}	{CTA}	{GCA}	{CTT}	{GCT}	{GCA}	{GGT}

Figure (4.11) : Segment11 of The Second Message

{GCT}	{GGT}	{CGG}	{GCT}	{CGG}	{AGA}	{GCT}
{AGT}	{GCA}	{AGA}	{CGG}	{GGA}	{CTT}	{CGA}
{GCA}	{GGT}	{CTA}	{GCT}	{ACT}	{GTG}	{GCT}
{CTT}	{AGT}	{GGT}	{CCT}	{CCT}	{CTT}	{TGA}
{GCT}	{CGG}	{CTT}	{GGA}	{CTA}	{CTT}	{CCT}
{GCT}	{GGT}	{GCT}	{AGT}	{GCA}	{AGA}	{CAA}
{GGT}	{CTT}	{GGA}	{GCT}	{TGA}	{GGT}	{GGA}

Figure (4.12) : Segment12 of The Second Message

{GCC}	{GCT}	{CCT}	{CTT}	{AGA}	{CGG}	{GGA}
{GGT}	{GGG}	{GCT}	{CAA}	{GGA}	{AGA}	{AGG}
{CGG}	{GCT}	{AGT}	{GCA}	{AGA}	{CGG}	{GCA}
{CTT}	{CGA}	{CTT}	{GGA}	{GGG}	{GCT}	{GGA}
{GGT}	{GCC}	{GCT}	{CTT}	{CGG}	{GGA}	{AGA}
{GCA}	{AGG}	{CTA}	{CTT}	{CGG}	{GCT}	{GGT}
{GCT}	{CCT}	{AGA}	{GCT}	{ACT}	{AGG}	{GGG}

Figure (4.13) : Segment 13 of The Second Message

{GGT}	{GGA}	{AGT}	{GGT}	{CGG}	{GGG}	{ACT}
{GGA}	{AAA}	{GCT}	{TAC}	{ATC}	{AGC}	{TAA}
{GCT}	{ACT}	{AGG}	{GGG}	{GGT}	{GGA}	{AGT}
{GGT}	{CGG}	{GGG}	{ACT}	{GGA}	{CTA}	{GCT}
{CAA}	{GCA}	{GGT}	{GCT}	{ACT}	{AGG}	{GGG}
{GGT}	{GGA}	{AGT}	{GGT}	{GCA}	{GGT}	{AGT}
{CTT}	{CGG}	{CCT}	{GCT}	{TCC}	{CCT}	{CTT}

Figure (4.14) : Segment 14 of The Second Message

{CGG}	{GGA}	{AGA}	{GCA}	{AGG}	{CTA}	{CTT}
{CGG}	{GCT}	{GGA}	{GGT}	{AGA}	{GGT}	{TGA}
{GCT}	{GGT}	{ACA}	{CGG}	{GCT}	{ACT}	{GTG}
{GCT}	{CAA}	{CTT}	{CAA}	{AGA}	{CGA}	{GGT}
{GGA}	{GGG}	{GCT}	{CCT}	{AGA}	{GCT}	{CCT}
{CCT}	{CTT}	{CTA}	{CTA}	{GGT}	{GCT}	{CAA}
{CTT}	{TAA}	{AGA}	{GGA}	{GGT}	{AGG}	{CGG}

Figure (4.15) : Segment 15 of The Second Message

{CGG}	{GGT}	{GCA}	{CGG}	{GCT}	{TGA}	{GGT}
{GGA}	{GCC}	{GCT}	{GCA}	{GGT}	{AGA}	{CGG}
{CTA}	{CTT}	{CGG}	{GGT}	{GGA}	{GGG}	{GCT}
{GGT}	{CGG}	{GGT}	{CAA}	{GCT}	{CTT}	{CGA}
{AGA}	{CGG}	{AGA}	{CCT}	{GCA}	{CTT}	{ATT}
{GCT}	{TAC}	{CCT}	{TGT}	{CTA}	{GGT}	{CGG}
{CGG}	{GGT}	{GCT}	{CAA}	{AGA}	{GGT}	{CGA}

Figure (4.16) : Segment16 of The Second Message

{GGT}	{GCT}	{GGT}	{CGG}	{GCT}	{GGA}	{CTT}
{GCA}	{GCA}	{GGT}	{TGA}	{GCT}	{TGG}	{GGT}
{AGA}	{CGG}	{GCA}	{CTT}	{CAA}	{AGA}	{GCC}
{GCA}	{GGT}	{CTA}	{GCT}	{GGT}	{GCT}	{GCA}
{AGA}	{GCT}	{CTT}	{GTG}	{GCT}	{GGT}	{CGG}
{GCT}	{CAA}	{CTT}	{GGA}	{AGA}	{CGG}	{GGA}
{CTT}	{GGA}	{GCT}	{CCT}	{AGA}	{GCT}	{CCT}

Figure (4.17) : Segment 17 of The Second Message

{GGA}	{CTT}	{CGA}	{AGA}	{CTT}	{CTA}	{CTT}
{GGA}	{GCT}	{CAA}	{GCA}	{GGT}	{GCT}	{CCT}
{GGA}	{CTT}	{CAA}	{GCA}	{CTT}	{CCT}	{GCT}
{GCA}	{CTT}	{CTT}	{ACA}	{CGG}	{CTT}	{GTG}
{GCT}	{CCT}	{CTT}	{AGT}	{GGT}	{CCT}	{CCT}
{CTT}	{TGA}	{GCT}	{AGT}	{GCA}	{AGA}	{AGT}
{GCA}	{GGT}	{AGG}	{CTA}	{ATG}	{CTT}	{GCT}

Figure (4.18) : Segment 18 of The Second Message

{GCA}	{CTT}	{GGA}	{GGT}	{GCC}	{CTT}	{GGA}
{CTT}	{AGG}	{ATG}	{GCT}	{TAC}	{CCT}	{TGT}
{CCT}	{AGA}	{GGA}	{GCT}	{AGG}	{AGT}	{AGA}
{AGG}	{GCT}	{CCT}	{CTT}	{CTA}	{GGT}	{GCC}
{GCT}	{CGG}	{CTT}	{GCA}	{GGA}	{CTT}	{CGG}
{GCA}	{GAT}	{GCT}	{CTT}	{AGG}	{CGG}	{GCT}
{AGG}	{AGT}	{CGG}	{GGT}	{GGA}	{AGG}	{CGG}

Figure (4.19) : Segment19 of The Second Message

{GCT}	{GCA}	{GGT}	{AGA}	{CGG}	{GGT}	{TGA}
{GGA}	{GGT}	{GCC}	{GCA}	{AGA}	{GCT}	{CTT}
{CGA}	{AGA}	{CGG}	{AGA}	{CCT}	{GCA}	{CTT}
{CCT}	{GCT}	{GCC}	{GGT}	{GCT}	{GCA}	{GGT}
{AGA}	{CCT}	{CCT}	{AGA}	{TGA}	{CCT}	{GCA}
{GGT}	{GGA}	{CGG}	{GCT}	{CTT}	{AGG}	{ATG}
{GCA}	{GCA}	{GCA}	{GCA}	{GCA}	{GCA}	{GCA}

Figure (4.20) : Segment10 of The Second Message

Appendix D :

3-SNPs Generation Stage

3-1.Digitize DNA base Process

{302}	{123}	{023}	{011}	{111}	{113}	{110}
{013}	{113}	{211}	{111}	{023}	{110}	{230}
{123}	{011}	{211}	{010}	{020}	{123}	{023}
{011}	{111}	{113}	{110}	{013}	{113}	{120}
{113}	{013}	{233}	{211}	{223}	{123}	{233}
{120}	{010}	{131}	{310}	{113}	{230}	{123}
{211}	{113}	{011}	{211}	{123}	{223}	{233}
{011}	{230}	{113}	{113}	{110}	{111}	{111}

Figure (5.1): Digitization of Segment 1

{113}	{123}	{233}	{310}	{113}	{223}	{123}
{023}	{131}	{123}	{200}	{233}	{200}	{010}
{210}	{113}	{110}	{111}	{123}	{223}	{010}
{123}	{023}	{211}	{010}	{110}	{211}	{230}
{233}	{223}	{123}	{122}	{113}	{123}	{110}
{233}	{023}	{113}	{311}	{123}	{233}	{311}
{131}	{211}	{113}	{200}	{123}	{003}	{123}

Figure (5.2): Digitization of segment 2

{302}	{032}	{331}	{300}	{123}	{233}	{230}
{120}	{233}	{211}	{110}	{233}	{223}	{123}
{222}	{003}	{201}	{002}	{100}	{123}	{200}
{010}	{230}	{003}	{123}	{230}	{010}	{233}
{311}	{230}	{211}	{120}	{113}	{131}	{010}
{110}	{023}	{031}	{113}	{233}	{002}	{123}
{233}	{011}	{211}	{123}	{200}	{120}	{113}

Figure (5.3) :Digitization of segment 3

{123}	{120}	{032}	{232}	{300}	{123}	{113}
{233}	{200}	{010}	{210}	{123}	{120}	{032}
{210}	{300}	{123}	{113}	{010}	{200}	{211}
{113}	{123}	{120}	{032}	{221}	{300}	{032}
{220}	{300}	{123}	{032}	{223}	{300}	{123}
{233}	{013}	{113}	{310}	{010}	{123}	{120}
{032}	{023}	{300}	{123}	{211}	{031}	{233}

Figure (5.4) :Digitization of segment 4

{211}	{123}	{233}	{131}	{123}	{200}	{311}
{211}	{113}	{230}	{123}	{113}	{010}	{200}
{233}	{310}	{123}	{110}	{233}	{210}	{113}
{230}	{123}	{233}	{011}	{031}	{123}	{302}
{200}	{113}	{233}	{110}	{123}	{311}	{113}
{013}	{233}	{311}	{311}	{010}	{123}	{023}
{120}	{113}	{123}	{113}	{211}	{123}	{233}

Figure (5.5) :Digitization of segment 5

{311}	{131}	{113}	{233}	{230}	{010}	{211}
{113}	{120}	{120}	{211}	{123}	{211}	{010}
{123}	{013}	{120}	{010}	{313}	{113}	{310}
{123}	{233}	{013}	{113}	{223}	{223}	{233}
{310}	{123}	{110}	{233}	{210}	{113}	{230}
{123}	{113}	{123}	{233}	{200}	{010}	{223}
{120}	{010}	{123}	{233}	{013}	{113}	{223}

Figure (5.6) :Digitization of segment 6

{223}	{233}	{310}	{123}	{211}	{233}	{110}
{230}	{233}	{223}	{123}	{113}	{123}	{013}
{120}	{010}	{200}	{010}	{011}	{123}	{122}
{113}	{123}	{233}	{211}	{110}	{010}	{120}
{011}	{230}	{233}	{211}	{123}	{113}	{123}
{223}	{010}	{123}	{023}	{011}	{111}	{113}
{110}	{013}	{113}	{120}	{113}	{013}	{233}

Figure (5.7) : Digitization of segment 7

{211}	{033}	{123}	{302}	{032}	{022}	{300}
{123}	{222}	{033}	{303}	{003}	{100}	{123}
{200}	{110}	{113}	{200}	{120}	{113}	{211}
{033}	{123}	{120}	{113}	{010}	{211}	{111}
{023}	{110}	{230}	{120}	{303}	{123}	{200}
{233}	{230}	{120}	{113}	{210}	{200}	{003}
{123}	{200}	{120}	{113}	{123}	{120}	{222}

Figure (5.8) :Digitization of segment 8

{003}	{033}	{030}	{100}	{123}	{120}	{113}
{310}	{233}	{311}	{200}	{003}	{123}	{110}
{010}	{310}	{113}	{011}	{033}	{123}	{211}
{223}	{233}	{210}	{010}	{030}	{123}	{120}
{110}	{010}	{113}	{122}	{023}	{113}	{311}
{202}	{123}	{233}	{110}	{113}	{123}	{120}
{113}	{010}	{211}	{111}	{023}	{110}	{230}

Figure (5.9) : Digitization of segment 9

{120}	{233}	{123}	{233}	{011}	{211}	{123}
{120}	{010}	{123}	{200}	{233}	{010}	{311}
{111}	{111}	{113}	{123}	{223}	{200}	{113}
{011}	{211}	{233}	{310}	{123}	{233}	{011}
{211}	{123}	{122}	{113}	{123}	{233}	{310}
{113}	{033}	{123}	{302}	{223}	{110}	{233}
{200}	{211}	{110}	{211}	{120}	{010}	{123}

Figure (5.10) : Digitization of segment 10

{110}	{113}	{122}	{123}	{233}	{311}	{131}
{010}	{223}	{120}	{233}	{011}	{233}	{110}
{111}	{310}	{113}	{230}	{120}	{010}	{123}
{223}	{010}	{123}	{011}	{230}	{010}	{011}
{020}	{123}	{211}	{113}	{310}	{110}	{113}
{122}	{123}	{200}	{233}	{211}	{111}	{023}
{110}	{230}	{120}	{233}	{123}	{120}	{113}

Figure (5.11) : Digitization of segment 11

{123}	{113}	{211}	{123}	{211}	{010}	{123}
{013}	{120}	{010}	{211}	{110}	{233}	{210}
{120}	{113}	{230}	{123}	{023}	{131}	{123}
{233}	{013}	{113}	{223}	{223}	{233}	{310}
{123}	{211}	{233}	{110}	{230}	{233}	{223}
{123}	{113}	{123}	{013}	{120}	{010}	{200}
{113}	{233}	{110}	{123}	{310}	{113}	{110}

Figure (5.12) :Digitization of segment 12

{122}	{123}	{223}	{233}	{010}	{211}	{110}
{113}	{111}	{123}	{200}	{110}	{010}	{011}
{211}	{123}	{013}	{120}	{010}	{211}	{120}
{233}	{210}	{233}	{110}	{111}	{123}	{110}
{113}	{122}	{123}	{233}	{211}	{110}	{010}
{120}	{011}	{230}	{233}	{211}	{123}	{113}
{123}	{223}	{010}	{123}	{023}	{011}	{111}

Figure (5.13) : Digitization of segment 13

{113}	{110}	{013}	{113}	{211}	{111}	{023}
{110}	{000}	{123}	{302}	{032}	{012}	{300}
{123}	{023}	{011}	{111}	{113}	{110}	{013}
{113}	{211}	{111}	{023}	{110}	{230}	{123}
{200}	{120}	{113}	{123}	{023}	{011}	{111}
{113}	{110}	{013}	{113}	{120}	{113}	{013}
{233}	{211}	{223}	{123}	{322}	{223}	{233}

Figure (5.14) : Digitization of segment 14

{211}	{110}	{010}	{120}	{011}	{230}	{233}
{211}	{123}	{110}	{113}	{010}	{113}	{310}
{123}	{113}	{020}	{211}	{123}	{023}	{131}
{123}	{200}	{233}	{200}	{010}	{210}	{113}
{110}	{111}	{123}	{223}	{010}	{123}	{223}
{223}	{233}	{230}	{230}	{113}	{123}	{200}
{233}	{300}	{010}	{110}	{113}	{011}	{211}

Figure (5.15) : Digitization of segment 15

{211}	{113}	{120}	{211}	{123}	{310}	{113}
{110}	{122}	{123}	{120}	{113}	{010}	{211}
{230}	{233}	{211}	{113}	{110}	{111}	{123}
{113}	{211}	{113}	{200}	{123}	{233}	{210}
{010}	{211}	{010}	{223}	{120}	{233}	{033}
{123}	{302}	{223}	{313}	{230}	{113}	{211}
{211}	{113}	{123}	{200}	{010}	{113}	{210}

Figure (5.16) : Digitization of segment 16

{211}	{113}	{120}	{211}	{123}	{310}	{113}
{110}	{122}	{123}	{120}	{113}	{010}	{211}
{230}	{233}	{211}	{113}	{110}	{111}	{123}
{113}	{211}	{113}	{200}	{123}	{233}	{210}
{010}	{211}	{010}	{223}	{120}	{233}	{033}
{123}	{302}	{223}	{313}	{230}	{113}	{211}
{211}	{113}	{123}	{200}	{010}	{113}	{210}

Figure (5.17) : Digitization of segment 17

{110}	{233}	{210}	{010}	{233}	{230}	{233}
{110}	{123}	{200}	{120}	{113}	{123}	{223}
{110}	{233}	{200}	{120}	{233}	{223}	{123}
{120}	{233}	{233}	{020}	{211}	{233}	{131}
{123}	{223}	{233}	{013}	{113}	{223}	{223}
{233}	{310}	{123}	{013}	{120}	{010}	{013}
{120}	{113}	{011}	{230}	{031}	{233}	{123}

Figure (5.18) : Digitization of segment 18

{120}	{233}	{110}	{113}	{122}	{233}	{110}
{233}	{011}	{031}	{123}	{302}	{223}	{313}
{223}	{010}	{110}	{123}	{011}	{013}	{010}
{011}	{123}	{223}	{233}	{230}	{113}	{122}
{123}	{211}	{233}	{120}	{110}	{233}	{211}
{120}	{103}	{123}	{233}	{011}	{211}	{123}
{011}	{013}	{211}	{113}	{110}	{011}	{211}

Figure (5.19) : Digitization of segment 19

{123}	{120}	{113}	{010}	{211}	{113}	{310}
{110}	{113}	{122}	{120}	{010}	{123}	{233}
{210}	{010}	{211}	{010}	{223}	{120}	{233}
{223}	{123}	{122}	{113}	{123}	{120}	{113}
{010}	{223}	{223}	{010}	{310}	{223}	{120}
{113}	{110}	{211}	{123}	{233}	{011}	{031}
{120}	{120}	{120}	{120}	{120}	{120}	{120}

Figure (5.20) : Digitization of segment 20

3-2 Computation Process

{302}	{123}	{023}	{011}	{111}	{113}	{110}	{3}
{013}	{113}	{211}	{111}	{023}	{110}	{230}	{1}
{123}	{011}	{211}	{010}	{020}	{123}	{023}	{0}
{011}	{111}	{113}	{110}	{013}	{113}	{120}	{3}
{113}	{013}	{233}	{211}	{223}	{123}	{233}	{3}
{120}	{010}	{131}	{310}	{113}	{230}	{123}	{0}
{211}	{113}	{011}	{211}	{123}	{223}	{233}	{0}
{011}	{230}	{113}	{113}	{110}	{111}	{111}	{3}
{3}	{0}	{0}	{3}	{1}	{3}	{1}	{0}

Figure (6.1) : Computation Process of segment1

{113}	{123}	{233}	{310}	{113}	{223}	{123}	{3}
{023}	{131}	{123}	{200}	{233}	{200}	{010}	{3}
{210}	{113}	{110}	{111}	{123}	{223}	{010}	{3}
{123}	{023}	{211}	{010}	{110}	{211}	{230}	{1}
{233}	{223}	{123}	{122}	{113}	{123}	{110}	{0}
{233}	{023}	{113}	{311}	{123}	{233}	{311}	{0}
{131}	{211}	{113}	{200}	{123}	{003}	{123}	{0}
{0}	{0}	{3}	{3}	{0}	{1}	{1}	{3}

Figure (6.2) :Computation Process of Segment2

{302}	{032}	{331}	{300}	{123}	{233}	{230}	{0}
{120}	{233}	{211}	{110}	{233}	{223}	{123}	{1}
{222}	{003}	{201}	{002}	{100}	{123}	{200}	{0}
{010}	{230}	{003}	{123}	{230}	{010}	{233}	{1}
{311}	{230}	{211}	{120}	{113}	{131}	{010}	{0}
{110}	{023}	{031}	{113}	{233}	{002}	{123}	{3}
{233}	{011}	{211}	{123}	{200}	{120}	{113}	{1}
{0}	{3}	{3}	{1}	{1}	{3}	{3}	{1}

Figure (6.3) : Computation Process of Segment3

{123}	{120}	{032}	{232}	{300}	{123}	{113}	{1}
{233}	{200}	{010}	{210}	{123}	{120}	{032}	{0}
{210}	{300}	{123}	{113}	{010}	{200}	{211}	{3}
{113}	{123}	{120}	{032}	{221}	{300}	{032}	{3}
{220}	{300}	{123}	{032}	{223}	{300}	{123}	{1}
{233}	{013}	{113}	{310}	{010}	{123}	{120}	{3}
{032}	{023}	{300}	{123}	{211}	{031}	{233}	{1}
{3}	{3}	{0}	{3}	{0}	{0}	{0}	{1}

Figure (6.4) :Computation Process of Segment4

{211}	{123}	{233}	{131}	{123}	{200}	{311}	{0}
{211}	{113}	{230}	{123}	{113}	{010}	{200}	{1}
{233}	{310}	{123}	{110}	{233}	{210}	{113}	{3}
{230}	{123}	{233}	{011}	{031}	{123}	{302}	{3}
{200}	{113}	{233}	{110}	{123}	{311}	{113}	{1}
{013}	{233}	{311}	{311}	{010}	{123}	{023}	{1}
{120}	{113}	{123}	{113}	{211}	{123}	{233}	{1}
{3}	{3}	{1}	{3}	{0}	{1}	{0}	{0}

Figure (6.5) : Computation Process of Segment5

{311}	{131}	{113}	{233}	{230}	{010}	{211}	{0}
{113}	{120}	{120}	{211}	{123}	{211}	{010}	{3}
{123}	{013}	{120}	{010}	{313}	{113}	{310}	{1}
{123}	{233}	{013}	{113}	{223}	{223}	{233}	{3}
{310}	{123}	{110}	{233}	{210}	{113}	{230}	{1}
{123}	{113}	{123}	{233}	{200}	{010}	{223}	{0}
{120}	{010}	{123}	{233}	{013}	{113}	{223}	{3}
{3}	{0}	{1}	{3}	{0}	{3}	{1}	{3}

Figure (6.6) : Computation Process of Segment6

{223}	{233}	{310}	{123}	{211}	{233}	{110}	{3}
{230}	{233}	{223}	{123}	{113}	{123}	{013}	{3}
{120}	{010}	{200}	{010}	{011}	{123}	{122}	{0}
{113}	{123}	{233}	{211}	{110}	{010}	{120}	{1}
{011}	{230}	{233}	{211}	{123}	{113}	{123}	{0}
{223}	{010}	{123}	{023}	{011}	{111}	{113}	{0}
{110}	{013}	{113}	{120}	{113}	{013}	{233}	{0}
{0}	{3}	{1}	{1}	{1}	{0}	{3}	{3}

Figure (6.7) :Computation Process of Segment7

{211}	{033}	{123}	{302}	{032}	{022}	{300}	{3}
{123}	{222}	{033}	{303}	{003}	{100}	{123}	{1}
{200}	{110}	{113}	{200}	{120}	{113}	{211}	{0}
{033}	{123}	{120}	{113}	{010}	{211}	{111}	{0}
{023}	{110}	{230}	{120}	{303}	{123}	{200}	{1}
{233}	{230}	{120}	{113}	{210}	{200}	{003}	{0}
{123}	{200}	{120}	{113}	{123}	{120}	{222}	{0}
{1}	{3}	{1}	{1}	{0}	{3}	{3}	{3}

Figure (6.8) : Computation Process of Segment8

{003}	{033}	{030}	{100}	{123}	{120}	{113}	{3}
{310}	{233}	{311}	{200}	{003}	{123}	{110}	{0}
{010}	{310}	{113}	{011}	{033}	{123}	{211}	{1}
{223}	{233}	{210}	{010}	{030}	{123}	{120}	{0}
{110}	{010}	{113}	{122}	{023}	{113}	{311}	{0}
{202}	{123}	{233}	{110}	{113}	{123}	{120}	{1}
{113}	{010}	{211}	{111}	{023}	{110}	{230}	{1}
{0}	{0}	{0}	{0}	{0}	{3}	{3}	{1}

Figure (6.9) :Computation Process of Segment9

{120}	{233}	{123}	{233}	{011}	{211}	{123}	{0}
{120}	{010}	{123}	{200}	{233}	{010}	{311}	{0}
{111}	{111}	{113}	{123}	{223}	{200}	{113}	{1}
{011}	{211}	{233}	{310}	{123}	{233}	{011}	{1}
{211}	{123}	{122}	{113}	{123}	{233}	{310}	{3}
{113}	{033}	{123}	{302}	{223}	{110}	{233}	{1}
{200}	{211}	{110}	{211}	{120}	{010}	{123}	{0}
{0}	{1}	{3}	{0}	{3}	{1}	{0}	{0}

Figure (6.10) : Computation Process of Segment10

{110}	{113}	{122}	{123}	{233}	{311}	{131}	{1}
{010}	{223}	{120}	{233}	{011}	{233}	{110}	{1}
{111}	{310}	{113}	{230}	{120}	{010}	{123}	{3}
{223}	{010}	{123}	{011}	{230}	{010}	{011}	{1}
{020}	{123}	{211}	{113}	{310}	{110}	{113}	{0}
{122}	{123}	{200}	{233}	{211}	{111}	{023}	{3}
{110}	{230}	{120}	{233}	{123}	{120}	{113}	{0}
{0}	{0}	{3}	{3}	{3}	{0}	{0}	{1}

Figure (6.11): Computation Process of Segment11

{123}	{113}	{211}	{123}	{211}	{010}	{123}	{1}
{013}	{120}	{010}	{211}	{110}	{233}	{210}	{3}
{120}	{113}	{230}	{123}	{023}	{131}	{123}	{3}
{233}	{013}	{113}	{223}	{223}	{233}	{310}	{3}
{123}	{211}	{233}	{110}	{230}	{233}	{223}	{3}
{123}	{113}	{123}	{013}	{120}	{010}	{200}	{3}
{113}	{233}	{110}	{123}	{310}	{113}	{110}	{0}
{1}	{0}	{1}	{3}	{1}	{0}	{3}	{1}

Figure (6.12) : Computation Process of Segment12

{122}	{123}	{223}	{233}	{010}	{211}	{110}	{0}
{113}	{111}	{123}	{200}	{110}	{010}	{011}	{0}
{211}	{123}	{013}	{120}	{010}	{211}	{120}	{0}
{233}	{210}	{233}	{110}	{111}	{123}	{110}	{1}
{113}	{122}	{123}	{233}	{211}	{110}	{010}	{0}
{120}	{011}	{230}	{233}	{211}	{123}	{113}	{0}
{123}	{223}	{010}	{123}	{023}	{011}	{111}	{1}
{3}	{0}	{3}	{0}	{0}	{3}	{0}	{1}

Figure (6.13) : Computation Process of Segment13

{113}	{110}	{013}	{113}	{211}	{111}	{023}	{0}
{110}	{000}	{123}	{302}	{032}	{012}	{300}	{1}
{123}	{023}	{011}	{111}	{113}	{110}	{013}	{0}
{113}	{211}	{111}	{023}	{110}	{230}	{123}	{1}
{200}	{120}	{113}	{123}	{023}	{011}	{111}	{3}
{113}	{110}	{013}	{113}	{120}	{113}	{013}	{1}
{233}	{211}	{223}	{123}	{322}	{223}	{233}	{0}
{0}	{3}	{3}	{3}	{1}	{3}	{0}	{1}

Figure (6.14) : Computation Process of Segment14

{211}	{110}	{010}	{120}	{011}	{230}	{233}	{3}
{211}	{123}	{110}	{113}	{010}	{113}	{310}	{1}
{123}	{113}	{020}	{211}	{123}	{023}	{131}	{3}
{123}	{200}	{233}	{200}	{010}	{210}	{113}	{0}
{110}	{111}	{123}	{223}	{010}	{123}	{223}	{3}
{223}	{233}	{230}	{230}	{113}	{123}	{200}	{0}
{233}	{300}	{010}	{110}	{113}	{011}	{211}	{3}
{0}	{1}	{1}	{0}	{0}	{3}	{0}	{0}

Figure (6.15) : Computation Process of Segment15

{113}	{123}	{113}	{211}	{123}	{110}	{233}	{3}
{120}	{120}	{113}	{310}	{123}	{311}	{113}	{0}
{010}	{211}	{120}	{233}	{200}	{010}	{122}	{0}
{120}	{113}	{230}	{123}	{113}	{123}	{120}	{3}
{010}	{123}	{233}	{131}	{123}	{113}	{211}	{1}
{123}	{200}	{233}	{110}	{010}	{211}	{110}	{0}
{233}	{110}	{123}	{223}	{010}	{123}	{223}	{1}
{0}	{1}	{3}	{3}	{1}	{3}	{0}	{3}

Figure (6.16) Computation Process of Segment16

{113}	{123}	{113}	{211}	{123}	{110}	{233}	{3}
{120}	{120}	{113}	{310}	{123}	{311}	{113}	{0}
{010}	{211}	{120}	{233}	{200}	{010}	{122}	{0}
{120}	{113}	{230}	{123}	{113}	{123}	{120}	{3}
{010}	{123}	{233}	{131}	{123}	{113}	{211}	{1}
{123}	{200}	{233}	{110}	{010}	{211}	{110}	{0}
{233}	{110}	{123}	{223}	{010}	{123}	{223}	{1}
{0}	{1}	{3}	{3}	{1}	{3}	{0}	{3}

Figure (6.17) : Computation Process of Segment17

{110}	{233}	{210}	{010}	{233}	{230}	{233}	{3}
{110}	{123}	{200}	{120}	{113}	{123}	{223}	{1}
{110}	{233}	{200}	{120}	{233}	{223}	{123}	{0}
{120}	{233}	{233}	{020}	{211}	{233}	{131}	{0}
{123}	{223}	{233}	{013}	{113}	{223}	{223}	{0}
{233}	{310}	{123}	{013}	{120}	{010}	{013}	{1}
{120}	{113}	{011}	{230}	{031}	{233}	{123}	{0}
{0}	{3}	{0}	{1}	{3}	{3}	{1}	{0}

Figure (6.18) : Computation Process of Segment18

{120}	{233}	{110}	{113}	{122}	{233}	{110}	{0}
{233}	{011}	{031}	{123}	{302}	{223}	{313}	{0}
{223}	{010}	{110}	{123}	{011}	{013}	{010}	{1}
{011}	{123}	{223}	{233}	{230}	{113}	{122}	{0}
{123}	{211}	{233}	{120}	{110}	{233}	{211}	{0}
{120}	{103}	{123}	{233}	{011}	{211}	{123}	{1}
{011}	{013}	{211}	{113}	{110}	{011}	{211}	{1}
{1}	{0}	{0}	{1}	{3}	{1}	{1}	{0}

Figure (6.19): Computation Process of Segment19

{123}	{120}	{113}	{010}	{211}	{113}	{310}	{1}
{110}	{113}	{122}	{120}	{010}	{123}	{233}	{0}
{210}	{010}	{211}	{010}	{223}	{120}	{233}	{0}
{223}	{123}	{122}	{113}	{123}	{120}	{113}	{3}
{010}	{223}	{223}	{010}	{310}	{223}	{120}	{1}
{113}	{110}	{211}	{123}	{233}	{011}	{031}	{0}
{120}	{120}	{120}	{120}	{120}	{120}	{120}	{1}
{0}	{0}	{3}	{0}	{1}	{3}	{0}	{0}

Figure (6.20) : Computation Process of Segment20

{TAC}	{GCT}	{ACT}	{AGG}	{GGG}	{GGT}	{GGA}	{T}
{AGT}	{GGT}	{CGG}	{GGG}	{ACT}	{GGA}	{CTA}	{G}
{GCT}	{AGG}	{CGG}	{AGA}	{ACA}	{GCT}	{ACT}	{A}
{AGG}	{GGG}	{GGT}	{GGA}	{AGT}	{GGT}	{GCA}	{T}
{GGT}	{AGT}	{CTT}	{CGG}	{CCT}	{GCT}	{CTT}	{T}
{GCA}	{AGA}	{GTG}	{TGA}	{GGT}	{CTA}	{GCT}	{A}
{CGG}	{GGT}	{AGG}	{CGG}	{GCT}	{CCT}	{CTT}	{A}
{AGG}	{CTA}	{GGT}	{GGT}	{GGA}	{GGG}	{GGG}	{T}
{T}	{A}	{A}	{T}	{G}	{T}	{G}	{A}

Figure (6.1) :SNPs Assingment of Segment 1

{GGT}	{GCT}	{CTT}	{TGA}	{GGT}	{CCT}	{GCT}	{T}
{ACT}	{GTG}	{GCT}	{CAA}	{CTT}	{CAA}	{AGA}	{T}
{CGA}	{GGT}	{GGA}	{GGG}	{GCT}	{CCT}	{AGA}	{T}
{GCT}	{ACT}	{CGG}	{AGA}	{GGA}	{CGG}	{CTA}	{G}
{CTT}	{CCT}	{GCT}	{GCC}	{GGT}	{GCT}	{GGA}	{A}
{CTT}	{ACT}	{GGT}	{TGG}	{GCT}	{CTT}	{TGG}	{A}
{GTG}	{CGG}	{GGT}	{CAA}	{GCT}	{AAT}	{GCT}	{A}
{A}	{A}	{T}	{T}	{A}	{G}	{G}	{T}

Figure (6.2) :SNPs Assingment of segment 2

{TAC}	{ATC}	{TTG}	{TAA}	{GCT}	{CTT}	{CTA}	{A}
{GCA}	{CTT}	{CGG}	{GGA}	{CTT}	{CCT}	{GCT}	{G}
{CCC}	{AAT}	{CAG}	{AAC}	{GAA}	{GCT}	{CAA}	{A}
{AGA}	{CTA}	{AAT}	{GCT}	{CTA}	{AGA}	{CTT}	{G}
{TGG}	{CTA}	{CGG}	{GCA}	{GGT}	{GTG}	{AGA}	{A}
{GGA}	{ACT}	{ATG}	{GGT}	{CTT}	{AAC}	{GCT}	{T}
{CTT}	{AGG}	{CGG}	{GCT}	{CAA}	{GCA}	{GGT}	{G}
{A}	{T}	{T}	{G}	{G}	{T}	{T}	{G}

Figure (6.3) :SNPs Assingment of segment 3

{GCT}	{GCA}	{ATC}	{CTC}	{TAA}	{GCT}	{GGT}	{G}
{CTT}	{CAA}	{AGA}	{CGA}	{GCT}	{GCA}	{ATC}	{A}
{CGA}	{TAA}	{GCT}	{GGT}	{AGA}	{CAA}	{CGG}	{T}
{GGT}	{GCT}	{GCA}	{ATC}	{CCG}	{TAA}	{ATC}	{T}
{CCA}	{TAA}	{GCT}	{ATC}	{CCT}	{TAA}	{GCT}	{G}
{CTT}	{AGT}	{GGT}	{TGA}	{AGA}	{GCT}	{GCA}	{T}
{ATC}	{ACT}	{TAA}	{GCT}	{CGG}	{ATG}	{CTT}	{G}
{T}	{T}	{A}	{T}	{A}	{A}	{A}	{G}

Figure (6.4) :SNPs Assingment of segment 4

{CGG}	{GCT}	{CTT}	{GTG}	{GCT}	{CAA}	{TGG}	{A}
{CGG}	{GGT}	{CTA}	{GCT}	{GGT}	{AGA}	{CAA}	{G}
{CTT}	{TGA}	{GCT}	{GGA}	{CTT}	{CGA}	{GGT}	{T}
{CTA}	{GCT}	{CTT}	{AGG}	{ATG}	{GCT}	{TAC}	{T}
{CAA}	{GGT}	{CTT}	{GGA}	{GCT}	{TGG}	{GGT}	{G}
{AGT}	{CTT}	{TGG}	{TGG}	{AGA}	{GCT}	{ACT}	{G}
{GCA}	{GGT}	{GCT}	{GGT}	{CGG}	{GCT}	{CTT}	{G}
{T}	{T}	{G}	{T}	{A}	{G}	{A}	{A}

Figure (6.5) :SNPs Assingment of segment 5

{TGG}	{GTG}	{GGT}	{CTT}	{CTA}	{AGA}	{CGG}	{A}
{GGT}	{GCA}	{GCA}	{CGG}	{GCT}	{CGG}	{AGA}	{T}
{GCT}	{AGT}	{GCA}	{AGA}	{TGT}	{GGT}	{TGA}	{G}
{GCT}	{CTT}	{AGT}	{GGT}	{CCT}	{CCT}	{CTT}	{T}
{TGA}	{GCT}	{GGA}	{CTT}	{CGA}	{GGT}	{CTA}	{G}
{GCT}	{GGT}	{GCT}	{CTT}	{CAA}	{AGA}	{CCT}	{A}
{GCA}	{AGA}	{GCT}	{CTT}	{AGT}	{GGT}	{CCT}	{T}
{T}	{A}	{G}	{T}	{A}	{T}	{G}	{T}

Figure (6.6) :SNPs Assingment of segment 6

{CCT}	{CTT}	{TGA}	{GCT}	{CGG}	{CTT}	{GGA}	{T}
{CTA}	{CTT}	{CCT}	{GCT}	{GGT}	{GCT}	{AGT}	{T}
{GCA}	{AGA}	{CAA}	{AGA}	{AGG}	{GCT}	{GCC}	{A}
{GGT}	{GCT}	{CTT}	{CGG}	{GGA}	{AGA}	{GCA}	{G}
{AGG}	{CTA}	{CTT}	{CGG}	{GCT}	{GGT}	{GCT}	{A}
{CCT}	{AGA}	{GCT}	{ACT}	{AGG}	{GGG}	{GGT}	{A}
{GGA}	{AGT}	{GGT}	{GCA}	{GGT}	{AGT}	{CTT}	{A}
{A}	{T}	{G}	{G}	{G}	{A}	{T}	{T}

Figure (6.7) :SNPs Assingment of segment 7

{CGG}	{ATT}	{GCT}	{TAC}	{ATC}	{ACC}	{TAA}	{T}
{GCT}	{CCC}	{ATT}	{TAT}	{AAT}	{GAA}	{GCT}	{G}
{CAA}	{GGA}	{GGT}	{CAA}	{GCA}	{GGT}	{CGG}	{A}
{ATT}	{GCT}	{GCA}	{GGT}	{AGA}	{CGG}	{GGG}	{A}
{ACT}	{GGA}	{CTA}	{GCA}	{TAT}	{GCT}	{CAA}	{G}
{CTT}	{CTA}	{GCA}	{GGT}	{CGA}	{CAA}	{AAT}	{A}
{GCT}	{CAA}	{GCA}	{GGT}	{GCT}	{GCA}	{CCC}	{A}
{G}	{T}	{G}	{G}	{A}	{T}	{T}	{T}

Figure (6.8) :SNPs Assingment of segment 8

{AAT}	{ATT}	{ATA}	{GAA}	{GCT}	{GCA}	{GGT}	{T}
{TGA}	{CTT}	{TGG}	{CAA}	{AAT}	{GCT}	{GGA}	{A}
{AGA}	{TGA}	{GGT}	{AGG}	{ATT}	{GCT}	{CGG}	{G}
{CCT}	{CTT}	{CGA}	{AGA}	{ATA}	{GCT}	{GCA}	{A}
{GGA}	{AGA}	{GGT}	{GCC}	{ACT}	{GGT}	{TGG}	{A}
{CAC}	{GCT}	{CTT}	{GGA}	{GGT}	{GCT}	{GCA}	{G}
{GGT}	{AGA}	{CGG}	{GGG}	{ACT}	{GGA}	{CTA}	{G}
{A}	{A}	{A}	{A}	{A}	{T}	{T}	{G}

Figure (6.9) :SNPs Assingment of segment 9

{GCA}	{CTT}	{GCT}	{CTT}	{AGG}	{CGG}	{GCT}	{A}
{GCA}	{AGA}	{GCT}	{CAA}	{CTT}	{AGA}	{TGG}	{A}
{GGG}	{GGG}	{GGT}	{GCT}	{CCT}	{CAA}	{GGT}	{G}
{AGG}	{CGG}	{CTT}	{TGA}	{GCT}	{CTT}	{AGG}	{G}
{CGG}	{GCT}	{GCC}	{GGT}	{GCT}	{CTT}	{TGA}	{T}
{GGT}	{ATT}	{GCT}	{TAC}	{CCT}	{GGA}	{CTT}	{G}
{CAA}	{CGG}	{GGA}	{CGG}	{GCA}	{AGA}	{GCT}	{A}
{A}	{G}	{T}	{A}	{T}	{G}	{A}	{A}

Figure (6.10) :SNPs Assingment of segment 10

{GGA}	{GGT}	{GCC}	{GCT}	{CTT}	{TGG}	{GTG}	{G}
{AGA}	{CCT}	{GCA}	{CTT}	{AGG}	{CTT}	{GGA}	{G}
{GGG}	{TGA}	{GGT}	{CTA}	{GCA}	{AGA}	{GCT}	{T}
{CCT}	{AGA}	{GCT}	{AGG}	{CTA}	{AGA}	{AGG}	{G}
{ACA}	{GCT}	{CGG}	{GGT}	{TGA}	{GGA}	{GGT}	{A}
{GCC}	{GCT}	{CAA}	{CTT}	{CGG}	{GGG}	{ACT}	{T}
{GGA}	{CTA}	{GCA}	{CTT}	{GCT}	{GCA}	{GGT}	{A}
{A}	{A}	{T}	{T}	{T}	{A}	{A}	{G}

Figure (6.11) :SNPs Assingment of segment 11

{GCT}	{GGT}	{CGG}	{GCT}	{CGG}	{AGA}	{GCT}	{G}
{AGT}	{GCA}	{AGA}	{CGG}	{GGA}	{CTT}	{CGA}	{T}
{GCA}	{GGT}	{CTA}	{GCT}	{ACT}	{GTG}	{GCT}	{T}
{CTT}	{AGT}	{GGT}	{CCT}	{CCT}	{CTT}	{TGA}	{T}
{GCT}	{CGG}	{CTT}	{GGA}	{CTA}	{CTT}	{CCT}	{T}
{GCT}	{GGT}	{GCT}	{AGT}	{GCA}	{AGA}	{CAA}	{T}
{GGT}	{CTT}	{GGA}	{GCT}	{TGA}	{GGT}	{GGA}	{A}
{G}	{A}	{G}	{T}	{G}	{A}	{T}	{G}

Figure (6.12) :SNPs Assingment of segment 12

{GCC}	{GCT}	{CCT}	{CTT}	{AGA}	{CGG}	{GGA}	{A}
{GGT}	{GGG}	{GCT}	{CAA}	{GGA}	{AGA}	{AGG}	{A}
{CGG}	{GCT}	{AGT}	{GCA}	{AGA}	{CGG}	{GCA}	{A}
{CTT}	{CGA}	{CTT}	{GGA}	{GGG}	{GCT}	{GGA}	{G}
{GGT}	{GCC}	{GCT}	{CTT}	{CGG}	{GGA}	{AGA}	{A}
{GCA}	{AGG}	{CTA}	{CTT}	{CGG}	{GCT}	{GGT}	{A}
{GCT}	{CCT}	{AGA}	{GCT}	{ACT}	{AGG}	{GGG}	{G}
{T}	{A}	{T}	{A}	{A}	{T}	{A}	{G}

Figure (6.13) :SNPs Assingment of segment 13

{GGT}	{GGA}	{AGT}	{GGT}	{CGG}	{GGG}	{ACT}	{A}
{GGA}	{AAA}	{GCT}	{TAC}	{ATC}	{AGC}	{TAA}	{G}
{GCT}	{ACT}	{AGG}	{GGG}	{GGT}	{GGA}	{AGT}	{A}
{GGT}	{CGG}	{GGG}	{ACT}	{GGA}	{CTA}	{GCT}	{G}
{CAA}	{GCA}	{GGT}	{GCT}	{ACT}	{AGG}	{GGG}	{T}
{GGT}	{GGA}	{AGT}	{GGT}	{GCA}	{GGT}	{AGT}	{G}
{CTT}	{CGG}	{CCT}	{GCT}	{TCC}	{CCT}	{CTT}	{A}
{A}	{T}	{T}	{T}	{G}	{T}	{A}	{G}

Figure (6.14) :SNPs Assingment of segment 14

{CGG}	{GGA}	{AGA}	{GCA}	{AGG}	{CTA}	{CTT}	{T}
{CGG}	{GCT}	{GGA}	{GGT}	{AGA}	{GGT}	{TGA}	{G}
{GCT}	{GGT}	{ACA}	{CGG}	{GCT}	{ACT}	{GTG}	{T}
{GCT}	{CAA}	{CTT}	{CAA}	{AGA}	{CGA}	{GGT}	{A}
{GGA}	{GGG}	{GCT}	{CCT}	{AGA}	{GCT}	{CCT}	{T}
{CCT}	{CTT}	{CTA}	{CTA}	{GGT}	{GCT}	{CAA}	{A}
{CTT}	{TAA}	{AGA}	{GGA}	{GGT}	{AGG}	{CGG}	{T}
{A}	{G}	{G}	{A}	{A}	{T}	{A}	{A}

Figure (6.15) :SNPs Assingment of segment 15

{CGG}	{GGT}	{GCA}	{CGG}	{GCT}	{TGA}	{GGT}	{T}
{GGA}	{GCC}	{GCT}	{GCA}	{GGT}	{AGA}	{CGG}	{G}
{CTA}	{CTT}	{CGG}	{GGT}	{GGA}	{GGG}	{GCT}	{G}
{GGT}	{CGG}	{GGT}	{CAA}	{GCT}	{CTT}	{CGA}	{A}
{AGA}	{CGG}	{AGA}	{CCT}	{GCA}	{CTT}	{ATT}	{T}
{GCT}	{TAC}	{CCT}	{TGT}	{CTA}	{GGT}	{CGG}	{T}
{CGG}	{GGT}	{GCT}	{CAA}	{AGA}	{GGT}	{CGA}	{G}
{T}	{A}	{A}	{A}	{G}	{T}	{A}	{T}

Figure (6.16) :SNPs Assingment of segment 16

{GGT}	{GCT}	{GGT}	{CGG}	{GCT}	{GGA}	{CTT}	{T}
{GCA}	{GCA}	{GGT}	{TGA}	{GCT}	{TGG}	{GGT}	{A}
{AGA}	{CGG}	{GCA}	{CTT}	{CAA}	{AGA}	{GCC}	{A}
{GCA}	{GGT}	{CTA}	{GCT}	{GGT}	{GCT}	{GCA}	{T}
{AGA}	{GCT}	{CTT}	{GTG}	{GCT}	{GGT}	{CGG}	{G}
{GCT}	{CAA}	{CTT}	{GGA}	{AGA}	{CGG}	{GGA}	{A}
{CTT}	{GGA}	{GCT}	{CCT}	{AGA}	{GCT}	{CCT}	{G}
{A}	{G}	{T}	{T}	{G}	{T}	{A}	{T}

Figure (6.17) :SNPs Assingment of segment 17

{GGA}	{CTT}	{CGA}	{AGA}	{CTT}	{CTA}	{CTT}	{T}
{GGA}	{GCT}	{CAA}	{GCA}	{GGT}	{GCT}	{CCT}	{G}
{GGA}	{CTT}	{CAA}	{GCA}	{CTT}	{CCT}	{GCT}	{A}
{GCA}	{CTT}	{CTT}	{ACA}	{CGG}	{CTT}	{GTG}	{A}
{GCT}	{CCT}	{CTT}	{AGT}	{GGT}	{CCT}	{CCT}	{A}
{CTT}	{TGA}	{GCT}	{AGT}	{GCA}	{AGA}	{AGT}	{G}
{GCA}	{GGT}	{AGG}	{CTA}	{ATG}	{CTT}	{GCT}	{A}
{A}	{T}	{A}	{G}	{T}	{T}	{G}	{A}

Figure (6.18) :SNPs Assingment of segment 18

{GCA}	{CTT}	{GGA}	{GGT}	{GCC}	{CTT}	{GGA}	{A}
{CTT}	{AGG}	{ATG}	{GCT}	{TAC}	{CCT}	{TGT}	{A}
{CCT}	{AGA}	{GGA}	{GCT}	{AGG}	{AGT}	{AGA}	{G}
{AGG}	{GCT}	{CCT}	{CTT}	{CTA}	{GGT}	{GCC}	{A}
{GCT}	{CGG}	{CTT}	{GCA}	{GGA}	{CTT}	{CGG}	{A}
{GCA}	{GAT}	{GCT}	{CTT}	{AGG}	{CGG}	{GCT}	{G}
{AGG}	{AGT}	{CGG}	{GGT}	{GGA}	{AGG}	{CGG}	{G}
{G}	{A}	{A}	{G}	{T}	{G}	{G}	{A}

Figure (6.19) :SNPs Assingment of segment 19

{GCT}	{GCA}	{GGT}	{AGA}	{CGG}	{GGT}	{TGA}	{G}
{GGA}	{GGT}	{GCC}	{GCA}	{AGA}	{GCT}	{CTT}	{A}
{CGA}	{AGA}	{CGG}	{AGA}	{CCT}	{GCA}	{CTT}	{A}
{CCT}	{GCT}	{GCC}	{GGT}	{GCT}	{GCA}	{GGT}	{T}
{AGA}	{CCT}	{CCT}	{AGA}	{TGA}	{CCT}	{GCA}	{G}
{GGT}	{GGA}	{CGG}	{GCT}	{CTT}	{AGG}	{ATG}	{A}
{GCA}	{GCA}	{GCA}	{GCA}	{GCA}	{GCA}	{GCA}	{G}
{A}	{A}	{T}	{A}	{G}	{T}	{A}	{A}

Figure (6.20) :SNPs Assingment of segment 20

Appendix E

4-Embedding Process

Figure (7) : Embedding Process of Segment1

الخلاصة

جذبت تسلسلات الحمض النووي اهتمامًا كبيرًا مثل قطع المعلومات الرباعية التي يمكن استخدامها لتخزين المعلومات وحل المشكلات وتشفير الرسائل وإخفائها.

تميل الأبحاث الحديثة إلى استغلال خصائص الحمض النووي في الجانب الأمني. لذلك استخدموا ميزات الحمض النووي كفكرة جديدة في تضمين البيانات المنقولة بدلاً من استخدام الحمض النووي كحامل مخفي فقط كما في الطرق القديمة. تستخدم هذه الأبحاث سمات الحمض النووي التي تسمى تعدد أشكال النوكليوتيدات المفردة ((SNPsكواحدة من طرق إخفاء الحمض النووي الحديثة. ولكن، هناك العديد من القيود في استخدام أساليب البحوث التي تسلط الضوء على بعض نقاط الضعف، مثل التخزين المتسلسل لمقطع الرسالة الذي يمكن الكشف عنه بسهولة، باستخدام نفس جدول البحث لجميع الرسائل المدخلة، وهناك قيود أخرى مثل الغموض أو الطفرات ستقلل بشكل كبير من قدرة الاختباء، لذلك تميل هذه الأطروحة إلى حل هذه القيود. في هذه الأطروحة، تم اقتراح منهجية مطورة في إخفاء الحمض النووي لإخفاء رسالة في مناطق متغيرة (SNPs) تستغل ميزات الحمض النووي. من خلال هذه الطريقة، تم تشفير رسالة إلى نموذج DNA ليتم إخفاؤها عشوائيًا في جينوم SNPs باستخدام جدول بحث ديناميكي لكل رسالة. أثبتت منهجية إخفاء الحمض النووي المقترحة في هذه الأطروحة ميزتها في حماية هندسة الخلايا كما هو موضح من النتائج التي تم الحصول عليها. بالإضافة إلى ذلك، استخدم SNPs في الحمض النووي حيث توفر مواقع الاختباء ضوضاء جيدة، وحمولة صفرية، وسعة عالية، ومعدل تعديل منخفض، واحتمال تكسير منخفض، والحفاظ على الوظائف.

وزارة التعليم العالي والبحث العلمي
جامعة بابل كلية العلوم للبنات
قسم علوم الحاسوب

طريقة إخفاء الرسائل على أساس تعدد أشكال النوكليوتيدات المفردة

رسالة مقدمة الى مجلس كلية العلوم للبنات في جامعة بابل وهي جزء من
متطلبات الحصول على درجة الماجستير في علوم الحاسبات

مقدمة من قبل
ضفاف شاكر كاظم

بأشراف
الاستاذ الدكتور
سحر عادل الباوي

1444 هـ

2023 م