$N°$ d'ordre: 54/2016-C/MT

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université des Sciences et de la Technologie Houari Boumediene
Faculté de Mathématiques

THÈSE

Prsentée pour l'obtention du diplôme de **Doctorat LMD**
En: Mathématiques et Applications
Option: Algèbre et Codage et Cryptographie

Par : **BENNENN Nabil**

Construction des codes cycliques sur les anneaux finis pour computation d'ADN

Soutenue publiquement le 23/10/2016, devant le jury composé de:

M. Kamel BETINA	Professeur	USTHB	Président
Mme. Kenza GUENDA	MCA	USTHB	Directrice
Mme. Sihem MESNAGER	HDR	TélécomParisTech	Co-directrice
M. Abdallah MOKRANE	Professeur	U.Paris8	Examinateur
M. Lemnouar NOUI	Professeur	U.Batna	Examinateur
Mme. Aicha BATOUL	MCA	USTHB	Examinatrice
Mme. Aini LAOUDI	MCA	USTHB	Examinatrice

Acknowledgments

I am thankful to my supervisor, Dr. Guenda Kenza, who supported me throughout my thesis with her patience, kindness and knowledge. Without her support and guidance, my thesis would not have been possible. I am grateful also to my second supervisor, Professor Sihem Mesnager for all her help and encouragement. My thanks also the members of my committee, president, Pr. Kamel Betina, Pr. Abdellah Mokrane, Dr. Aicha Batoul, Dr. Aini Laoudi and Pr. Lemnouar Noui for reading my research proposal and previous draft of this dissertation and providing valuable comments and suggestions that help with the completion of this dissertation. Finally, I thank my family, my parents for supporting me throughout my study.

Contents

List of Tables

List of Figures

Introduction

DNA computing combines genetic data analysis with the computational science in order to tackle computationally difficult problems. This new field started by Leonard Adleman [3] who solved a hard (NP-complete) computational problem by DNA molecule in a test tube.
In DNA computing, the data is encoded using DNA strands and molecular biology techniques are used to simulate arithmetic and logical operations. The main advantages of this approach are huge memory capacity, massive parallelism and low power molecular hardware and software systems. Indeed $1g$ of DNA can be used to store about $4.2 * 10^{21}$ bits, while a conventional storage medium has a capacity of at most 109 bits. The massive parallelism is due to simultaneous biochemical reactions. However, this potential is limited by the constraints imposed by the combinatorial and the thermodynamic structure of DNA.
In this thesis we deal with the problem of the construction of the DNA codes with good combinatorial and thermodynamical structures in order to use these codes for DNA computing. we present Another results in the chapter 4, chapter 5 and chapter 6. More precisely In the chapter 4, we construct linear codes over $\mathbb{Z}_4$ with bounded GC-content. The codes are obtained using a greedy algorithm over $\mathbb{Z}_4$. Further, upper and lower bounds on the maximum size of DNA codes of length n with constant GC-content w and edit distance d are given. The choice of the ring $\mathbb{Z}_4$ comes from the fact that the bounded GC-content and bounded edit distance properties are multiplicative over $\mathbb{Z}_4$. This is not the case over $\mathbb{F}_4$. The bounded GC-constraint ensures that all codewords have thermodynamic characteristics below some threshold. This is an important criteria for DNA sequences as it reduces the probability of erroneous cross-hybridization. In [11], Chee and Ling gave an algorithm to construct DNA codes with large GC-content which are optimal only up to $n = 12$. Bishop et al. [10] considered the construction of random codes with

fixed GC-content using a probabilistic model. King [24] and Condon et al. [28] gave several upper and lower bounds on the maximum size of DNA codes of length n with constant GC-content w and Hamming distance d. It is well known that the Hamming distance does not capture the thermodynamic and the combinatorial properties of DNA strand. Thus, in the second part of this chapter upper and lower bounds are derived for the maximum size of DNA codes of length n with constant GC-content w and edit distance d. DNA codes, which are a set of equal length words over the alphabet $\{A, G, C, T\}$ that satisfy certain properties, are central to research on DNA computing. These codes allow information to be sent reliably over noisy channels [26]. When designing a code for DNA computing, it is important that the code is with a large number of codewords having the same melting temperature for stability. The WCC of each codeword is in this code. We call a DNA code with these properties *a DNA code with optimal thermodynamic and combinatorial properties. If the codewords are short, this temperature is determined by the neighborhood energy. This energy is maximal when the number of occurrences of C and G in the codewords is important [25]. Those codes are called with optimal thermodynamic property. The authors in [6] used a greedy algorithm to construct DNA codes over the ring $\mathbb{Z}_4$ with bounded GC content. Upper and lower bounds on the maximum size of a DNA code of length n over the alphabet $\{A, G, C, T\}$ were also given. Bishop et al. [10] considered the desired construction of random codes with fixed GC content using a probabilistic model. King [24] and Condon et al. [28] gave bounds on the maximum size of DNA codes of length n with constant GC content w and Hamming distance d.*

In the chapter 5, linear codes are constructed over the ring $\mathbb{Z}_{16}$. These codes are obtained using a greedy algorithm considering the code properties. Further, upper and lower bounds on the maximum size of a DNA code of length n with constant GC content w and deletion similarity distance D are given. This distance is important from the hybridization energy perspective. The choice of the ring $\mathbb{Z}_{16}$ comes from the fact that there is a one-to-one correspondence between the base pairs and the elements of $\mathbb{Z}_{16}$, and the GC content and deletion similarity distance properties are multiplicative over this ring. This is not the case with other finite rings of cardinality 16. Further, codes over $\mathbb{Z}_{16}$ can be mapped to DNA codes of length $2n$ which have a larger GC content than codes over $\mathbb{Z}_4$, and thus have a higher hybridization energy.

Several authors have contributed to provide constructions of cyclic DNA codes over fixed rings. In [2, 32], the authors gave DNA cyclic codes over finite

field with four elements. Further, Siap et al. have studied in [35] cyclic DNA codes over the ring $\mathbb{F}_2[u]/(u^2-1)$ *using the deletion distance. More recently, Guenda et al. have studied in [20] cyclic DNA codes of arbitrary length over the ring* $\mathbb{F}_2[u]/(u^4-1)$*. Those codes have several applications as well as high hybridization energy. In the chapter 6, we consider the DNA codes of length* n *over the ring* $R=\mathbb{F}_2[u]/(u^6)$*. The ring* R *is a principal commutative ring with* 64 *elements. With four possible bases, the three nucleotides can give* $4^3=64$ *different possibilities, called codons. These combinations are used to specify the* 20 *different amino acids used in the living organisms [4]. To this end, we construct a one-to-one correspondence between the elements of* R *and the 64 codons over the alphabet* $\{A,G,C,T\}^3$*. Such a correspondence allows us to extend the notion of the edit distance to the ring* R*. The edit distance is an important combinatorial notion for the DNA strands. It can be used for the correction of the insertion, deletion and substitution errors between the codewords. This it is not the case for the Hamming, deletion, and the additive stem distances. For that in this chapter, we design cyclic reverse-complement DNA codes over the ring* R *with designed edit distance* D*. We also give some upper and lower bounds on* D*. We define a Lee weight and a Gray map over* R*. The images of our DNA codes under the mapping are quasi-cyclic codes of index* 6 *and of length* $6n$ *over the alphabet* $\{A,G,C,T\}$*. There are several advantages in using codes over the ring* R*. We list some of them below:*

1. *There exists a one-to-one correspondence between the codons and the elements of the ring* R*.*

2. *A code over* R *can contains more codewords than codes of the same length over fields.*

3. *The factorization of* x^n-1 *is the same over the field* $\mathbb{F}_2$ *but is not the same over other rings. This fact simplifies the construction of cyclic codes over* R*.*

4. *The structure of the cyclic codes of any length over* R *is well-known [21], whereas little is know concerning the structure of cyclic codes of any length over rings.*

5. *The cyclic character of the DNA strands is desired because the genetic code should represent an equilibrium status [33]. Another advantage*

of cyclic codes, as indicated by Milenkovic and Kashyap [29], is that the complexity of the dynamic programming algorithm for testing DNA codes for secondary structure will be less for cyclic codes.

6. *The binary image of the cyclic codes over R under our Gray map are linear quasi-cyclic codes.*

In this thesis, we study the skew cyclic DNA codes over the ring $\tilde{R} = \mathbb{F}_2 + v\mathbb{F}_2 = \{0, 1, v, v+1\}$, where $v^2 = v$. The codes obtained satisfy the reverse-constraint. Further we give the binary images of the skew cyclic DNA codes and provide some examples. The advantage of studying the reversible DNA code in skew polynomial rings is to exhibit several factorizations. Therefore, many reverse-complement DNA code could be obtained in a skew polynomial ring (which is not the case in a commutative ring).

The rest of thesis is organized as follows:
In chapter 2 we give some preliminaries and definitions of DNA computing and some backgrounds on DNA codes.
In chapter 3, we give some other generality of the linear codes over finite chain ring and some other generality of linear codes and cyclic codes over the ring $\mathbb{Z}_4$.
The remainder of chapter 4 is organized as follows, some preliminary results are presented. used a greedy algorithm to obtain DNA codes with bounded GC-content, DNA lexicodes are constructed with bounded edit distance. Upper and lower bounds on the edit distance are also presented. In addition, examples of DNA codes with bounded GC-content and edit distance are given. The remainder of the chapter 5 is organized as follows. We give, some preliminary results are presented. employing a greedy algorithm to obtain DNA codes with bounded GC-content, and DNA lexicodes with distance deletion similarity distance D are constructed. Examples of DNA codes with bounded GC content are also given. The chapter 6 is organized as follows. We start by presenting some preliminaries results as well as the one-to-one correspondence between the element of the ring $R = \mathbb{F}_2[u]/(u^6)$ and the codons. Next, we give the algebraic structure of the cyclic codes over $R = \mathbb{F}_2[u]/(u^6)$ and we study the DNA cyclic codes and reverse-complement of these codes. Moreover, we define the Lee weight related to such codes and give the binary image of the cyclic DNA code. Some explicit examples of such codes are presented. We describe a skew cyclic DNA codes over $\tilde{R} = \mathbb{F}_2 + v\mathbb{F}_2 = \{0, 1, v, v+1\}$

where $v^2 = v$. We study their property of being reverse-complement and provide explicit examples of such codes with minimum Hamming distance.

Chapter 1

Some Backgrounds on DNA Codes

Deoxyribonucleic acid (DNA) contains the genetic program for the biological development of life. DNA is formed by strands linked together and twisted in the shape of a double helix. Each strand is a sequence of four possible nucleotides, two purines; adenine (A)*, guanine* (G) *and two pyrimidines; thymine* (T) *and cytosine* (C)*. The ends of the DNA strand are chemically polar with* $5'$ *and* $3'$ *ends. Hybridization, known as base pairing, occurs when a strand binds to another strand, forming a double strand of DNA. The strands are linked following the Watson-Crick model; every* (A) *is linked with a* (T)*, and every* (C) *with a* (G)*, and vice versa. We denote by* $\hat{x}$ *the complement of* x *defined as follows,* $\hat{A} = T, \hat{T} = A, \hat{G} = C$ *and* $\hat{C} = G$ *(for instance if* $x = (AGCTAC)$*, then its complement* $\hat{x} = (TCGATG)$*. The genetic code DNA stores the genetic information which consists of "codons" of three nucleotides. With four possible base, the three nucleotides can give* $4^3 = 64$ *different possibilities, and these combinations are used to specify the* 20 *different amino acids used by living organisms. The ribonucleic acid (RNA) that is directly involved in the transcription of pattern of bases from the DNA to provide a blueprint for the construction of the proteins is called messenger RNA. The pattern for protein synthesis is then read and translated into the language amino for protein construction with the help of transfer RNA.*
The biological distinction between the base positions in the codon, the chemical type of bases (purine and pyrimidine) and their hydrogen bond number have been most relevant codon properties used in the genetic code analysis.

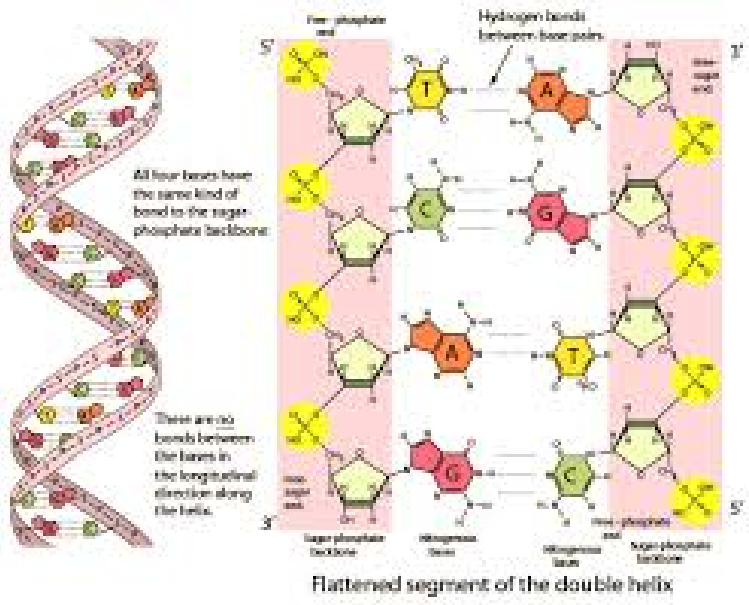

Figure 1.1: Image corresponding to DNA Codes

1.0.1 The GC-content over the DNA strands

In molecular biology and genetics, GC-content is the percentage of nitrogenous bases on a DNA molecule that are either guanine or cytosine (from a possibility of four different ones, also including adenine and thymine). This may refer to a specific fragment of DNA or RNA, or that of the whole genome. When it refers to a fragment of the genetic material, it may denote the GC-content of part of a gene (domain), single gene, group of genes (or gene clusters), or even a non-coding region. G (guanine) and C (cytosine) undergo a specific hydrogen bonding, whereas A (adenine) bonds specifically with T (thymine). The GC pair is bound by three hydrogen bonds, while AT pairs are bound by two hydrogen bonds. DNA with high GC-content is more stable than DNA with low GC-content; however, the hydrogen bonds do not stabilize the DNA significantly, and stabilization is due mainly to stacking interactions. In spite of the higher thermostability conferred to the genetic material, it is envisaged that cells with DNA of high GC-content undergo autolysis, thereby reducing the longevity. Due to the robustness endowed to the genetic materials in high GC organisms, it was commonly believed that

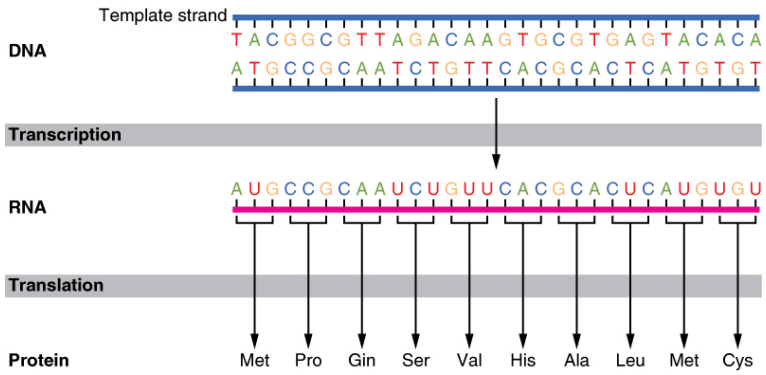

Figure 1.2: Image corresponding to genetic code

the GC content played a vital part in adaptation temperatures, a hypothesis that has recently been refuted [5]. However, the same study showed a strong correlation between higher temperatures and the GC content of structured RNAs (such as ribosomal RNA, transfer RNA, and many other non-coding RNAs); GC base pairs are more stable than AT base pairs, due to the fact that GC bonds have 3 hydrogen bonds and AT only has 2 hydrogen bonds, which makes high-GC-content RNA structures more tolerant of high temperatures. More recently, the first large-scale systematic gene-centric association analysis demonstrated the correlation between GC content and temperature for certain genomic regions while not for others. The GC-content percentages as well as GC-ratio can be measured by several means, but one of the simplest methods is to measure what is called the melting temperature of the DNA double helix using spectrophotometry. The absorbance of DNA at a wavelength of 260 nm increases fairly sharply when the double-stranded DNA separates into two single strands when sufficiently heated. The most commonly used protocol for determining GC ratios uses flow cytometry for large number of samples.

1.0.2 Secondary Structure of DNA

The DNA secondary structure plays an important role in biology, genotypic diagnostics, a variety of molecular biology techniques, in vitro-selected DNA catalysts, nontechnically, and DNA-based computing. Probably the most important criterion in designing codewords for DNA computing purposes is that

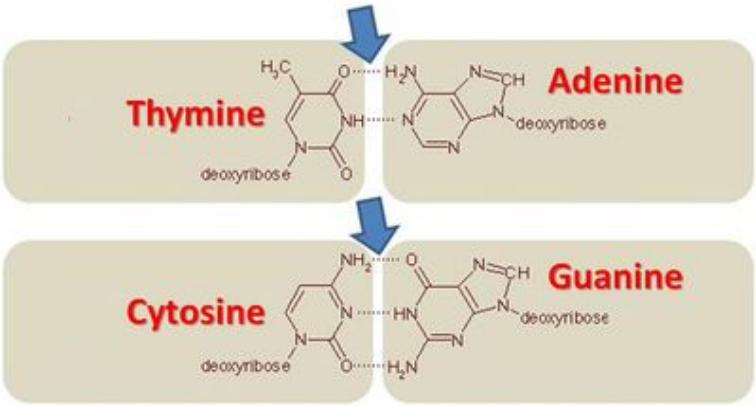

Figure 1.3: Image corresponding to nucleotide bonds hydrogen showing AT and GC pairs

the codewords should not form secondary structures that cause them to become computationally inactive. The secondary structure of a DNA codeword $(x_1, x_2 \cdot \cdot, x_n)$ *is a set* S*, of disjoint pairings between complementary bases* (x_i, x_j) *with* $i < j$*. A secondary structure is formed by a chemically active oligonucleotide sequence folding back onto itself due to self-hybridization, i.e., hybridization between complementary base pairs belonging to the same sequence. As a consequence of the bending, elaborate spatial structures are formed, the most important components of which are loops (including branching, internal, hairpin and bulge loops), stem helical regions, as well as unstructured single strands. Figure 2.4 illustrates these concepts for an RNA strand 1. It was shown experimentally that the most important factors influencing the secondary structure of a DNA sequence are the number of base pairs in stem regions, the number of base pairs in a hairpin loop region as well as the number of unpaired bases.*

Determining the exact pairings in a secondary structure of a DNA sequence is a complicated task, as we shall try to explain briefly. For a system of interacting entities, one measure commonly used for assessing the system's property is the free energy. The stability and form of a secondary configuration is usually governed by this energy, the general rule-of-thumb being that a secondary structure minimizes the free energy associated with a DNA sequence. The free energy of a secondary structure is determined by the energy of its constituent pairings. Now, the energy of a pairing depends on the bases

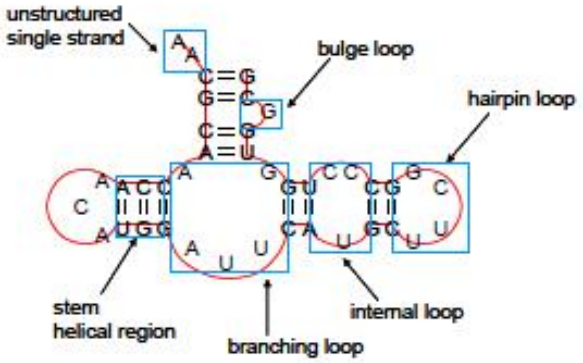

Figure 1.4: Image corresponding to DNA/RNA secondary structure model [29].

involved in the pairing as well as all bases adjacent to it. Adding complication is the fact that in the presence of other neighboring pairings, these energies change according to some nontrivial rules. Accurate prediction of DNA secondary structure and hybridization using dynamic programming algorithms requires a database of thermodynamic parameters for several motifs including Watson-Crick base pairs, internal mismatches, terminal mismatches, terminal dangling ends, hairpins, bulges, internal loops, and multi-branched loops, please see [29]. Among these techniques, Nussinov's folding algorithm is based on the assumption that in DNA sequence $(x_1, x_2 \cdot \cdot, x_n)$*, the energy between a pairs base* $\alpha(x_i, x_j)$*, is independent of the all other pairs base. For simplicity of exposition, we assume that* $\alpha(x_i, x_j) = -1$ *if* $\{x_i, x_j\} = \{A, T\}$*,* $\alpha(x_i, x_j) = -2$ *if* $\{x_i, x_j\} = \{G, C\}$ *and* $\alpha(x_i, x_j) = 0$ *otherwise. Let* $E_{i,j}$ *denote the minimum free energy of the subsequence* $(x_i, \cdot \cdot x_j)$ *. The independence assumption allows us to compute the minimum free energy of the sequence* $(x_1, x_2 \cdot \cdot, x_n)$ *through the recursion*

$$E_{i,j} = \min \begin{cases} E_{i+1,j-1} + \alpha(x_i, x_j); \\ E_{i,k-1} + E_{k,j}, i < k \leq j; \end{cases}$$

where $E_{i,i} = 0$ *for* $i = 1, 2, \cdot \cdot, n$ *and* $E_{i,i-1} = 0$ *for* $i = 2, \cdot \cdot, n$*. The value of* $E_{1,n}$ *is the minimum free energy of a secondary structure of* $(x_1, x_2, \cdot \cdot, x_n)$*.*

Note that $E_{1,n} = 0$. A very large negative value for the free energy $E_{1,n}$ of a sequence is a good indicator of the presence of stacked base pairs and loops, i.e., a secondary structure, in the physical DNA sequence.

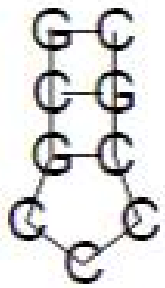

Figure 1.5: Image corresponding to secondary structure of the sequence GCGCCCCGC.

Chapter 2

DNA Computing

DNA computing has emerged as an interdisciplinary field that draws together molecular biology, chemistry, computer science and mathematics. Our knowledge on DNA computing increases exponentially with every passing year. The annual international meeting in the field began in 1995 right after the landmark work by L. Adleman, who solved an instance of the Hamiltonian path problem by DNA molecules and opened the door to this new field. Lipton wrote the paper in which it was show that the Adleman techniques could also be used to solve the NP-complete satisfiability (SAT) problem.

The Adleman-Lipton model of DNA computing: *The DNA operation in the Adleman-Lipton model will be used for figuring out solution of the vertex cover problem. A (test) tube is a set of the molecules of DNA (i.e.; a multi-set of the finite string over the alphabet A,G,C,T. Given a tube, one can perform the following operation,*

1. *Extract. Given a tube P a short single strand of the DNA, S, produce two tubes $+(P,S)$ and $-(P,S)$, where $+(P,S)$ is all of the molecules of DNA in P, which contain the strand S as a sub-strand and $-(P,S)$ is all of the molecules of the DNA in P, which do not contain the short strand S.*

2. *Merge. Given a tube P_1 and P_2, yield $\bigcup(P_1,P_2)$, where $\bigcup(P_1,P_2) = P_1 \bigcap P_2$. This operation is to pour two tubes into one, with no change of the individual strand.*

3. *Detect. Given a tube P, say 'yes' if P includes at least one DNA molecule, and say 'no' if it contains none.*

4. *Discard. Given a tube P, the operation will discard the tube P.*

5. *Read. Given a tube P, the operation is used to describe a single molecule, which is contained vertex-colorability problem, the clique problem and the independent-set problem.*

2.1 DNA Storage Medium

Artificial Genes and DNA computing is proposed as a means of data storage and cryptographic data transportation for a very high level of the security and classified data transmission, this type of the data needs to be protected with the state of art advanced security solutions.
A bioengineer and geneticist at Harvard's Wyss Institute have successfully stored 5.5 *petabits of data around* 700 *terabytes in a single gram of DNA,see[37] smashing the previous DNA data density record by a thousand times. The work, carried out by George Church and Sri Kosuri, basically treats DNA as just another digital storage device. Instead of binary data being encoded as magnetic regions on a hard drive platter, strands of DNA that store 96 bits are synthesized, with each of the bases $\{TGAC\}$ representing a binary value (T and $G = 1$, A and $C = 0$). Scientists have been eyeing up DNA as a*

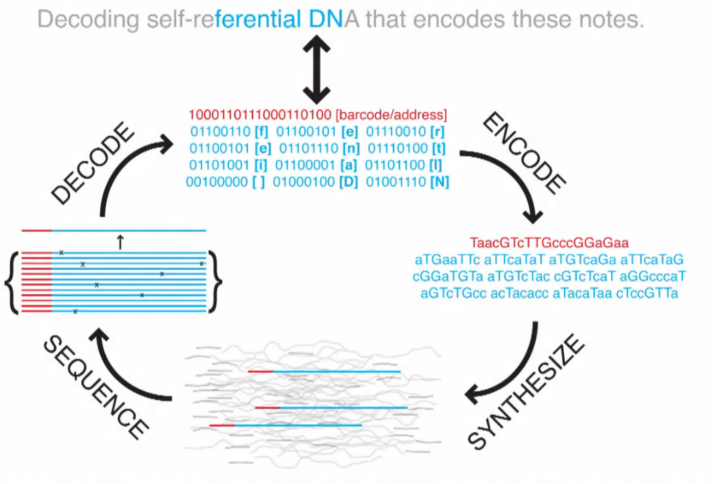

Figure 2.1: Image corresponding to DNA encoding and decoding [37]

potential storage medium for long time, for five very good reasons:

1. *The DNA code is incredibly dense (you can store one bit per base, and a base is only a few atoms large). Extremely compact DNA storage that 1 GRAM of DNA contains* 2.1×10^{21} *DNA based, can store approximately* $4.2 \times 10^{12} = 4.2$ *billion bits, potential data storage capacity of DNA a factors of* 4×10^{21} *more compact than conventional storage technologies.*

2. *The DNA code is volumetric rather than hard disk. In (Church et al. 2012) a book consisting of* 53.426 *words,* 11 *JPEG images and JavaScript program was encode as a* 5.27 *megabits. This was stored in* $54,898$ *single strands of synthetic. Each strand had length* 159 *bases,* A, G, T, C*, but only* 96 *of the base were used for the data storage.*

3. *The DNA code is incredibly stable where other bleeding-edge storage mediums need to be kept in sub-zero vacuums. The chemistry of DNA is very stable. It can remain readable for many thousands of years under quite mild storage conditions, such as those found in a seed storage vault. This is evidenced by the fact that is has proved possible to sequence DNA a woolly mammoth that has been dead for* $20,000$ *years. The DNA code is really suitable for long term archives that are rarely accessed. On the other hand, it needs no maintenance comparable for example, the moving of the data between servers every 5-10 years as they updated, and a process not without cost over long time periods. It is estimated in (Goldman et al., 2013) that may be cost effective now for the archives of the several megabytes with about* $600 - 5000$ *years storage.*

4. *In order to keep away from cloning and sequence verifying construct the manufacture and use of stored and sequenced copies of each individual oligo was done. As error in synthesis and sequencing are not often coincident, thus each molecular copy corrects errors in the other copies.*

5. *The use of next-generation technologies in the both DNA synthesis and sequencing to allow for encoding and decoding of large amounts of information for* $100,000$*-fold cost than fist-generation encoding.*

For give DNA based information storage system DNA information density is total amount of data that can be stored in unit gram of the DNA.
The theoretical maximal, one gram of single strand genetic DNA code can encode 455 EB (exabytes) of information. Goldman achieved information

density 2.2 PB (petabytes) per gram of DNA. Using DNA Golay code computationally, we achieved information density for DNA based storage medium as $1.15 \times 10^{20} = 115EB$ ($115000PB$). In following the description of derivation of DNA information density.
Consider total information that can be encoded in one gram of DNA is x bytes.

1. *Number of nucleotides required to store file is denoted by I*

$$I = 11 \times x + 11 \times 2 + 11 \times log_{10}11 \times (x)$$

   where $x \times 11$ are nucleotides for x bytes, 2×11 are nucleotides for 2 separators mentioned in this algorithm, $log_{10}11 \times (x)$ are nucleotides for storing file size on DNA.

2. *Number of chunk $C = \lceil I/99 \rceil$, each chunk will have 99 nucleotide of the file information.*

3. *Length of the chunk $l = \lceil log_3(C) \rceil + 102$, $\lceil log_3(C) \rceil$ is chunk index information.*

4. *total Number of nucleotides for storing files on DNA*

$$[102 + log_3(I/99)] \times (x \times 11 + 2 \times 11 + log_{10}(x) \times 11)/99$$

   Since maximum storage capacity of DNA is 443 we can consider $log_{10}(x) =$ 20.

5. *1 gram of the DNA consists of 182×10^{19} nucleotides, so for calculating number of the bytes that can be stored on 1 gram of DNA:*

$$182 \times 10^{19} = (102 + log_3(I/99)) \times (x \times 11 + 2 \times 11 + log_{10}(x) \times 11)/99$$

   $x = 1.15 \times 10^{20}$ *bytes* (115 *Exabytes*).

DNA is remarkably robust for an organic molecule. When it's protected inside fossils, this is what the ETH Zurich team is working on keeping the data viable by protecting the DNA carrier. They basically made tiny synthetic fossils.

A number of studies have demonstrated encoding and decoding information in DNA, but the storage times have been short. Even when DNA is well-protected, there's the possibility of data loss. This new study accounts

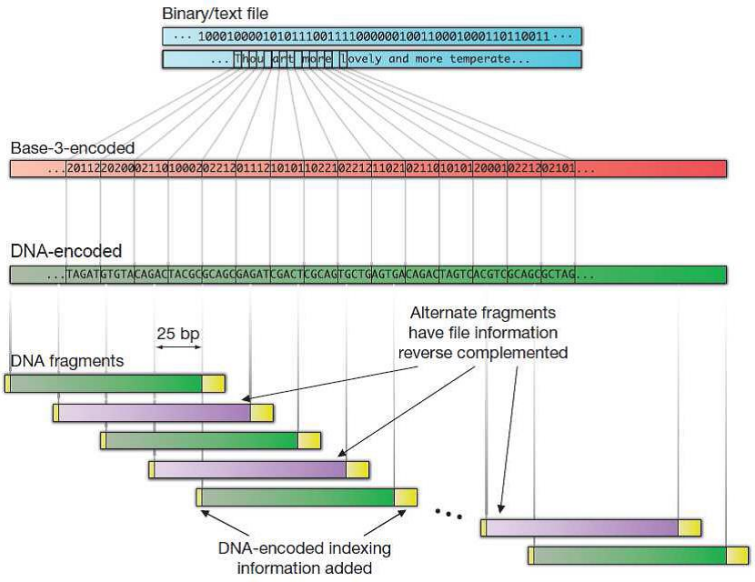

Figure 2.2: Digital information encoding in DNA by [17]

for both storage and error correction. The team encoded Switzerland's Federal Charter of 1291 and The methods of mechanical theorems by Archimedes on DNA strands and encapsulated the samples in tiny silica spheres 150nm in diameter. To simulate long periods of time, the DNA was heated to a temperature between 60 and 70 degrees Celsius for one month. At the end of the simulated centuries, researchers were able to read the original data from the DNA. The team estimates that DNA stored in these spheres could survive over one million years in the Svalbard Global Seed Vault, which stays at a frosty -18 degrees Celsius. This doesn't mean there won't be any errors, but error correction is built into the data. The algorithm used is similar to those used to ensure radio communications with spacecraft are free of errors. This scheme worked well in tests.

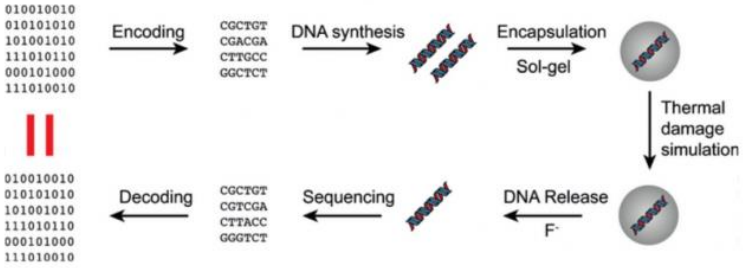

Figure 2.3: Image of the basically made tiny synthetic fossils [40]

2.2 General Information on Error Correcting Codes

In this section we introduce and discuss the notion of the error -detection and error correction. We also introduce some well know methods that retrieve the original message sent by detecting and correcting the errors that have occurred in the transmission.
The error correction codes are a tool to improve the reliability of transmissions a noisy channel. They are used in practice as follows: We have a message through of $\mathbb{F}_q$ elements (finite field with q elements) we want to send through a channel. The channels are often considered the communication channels (satellites, optics fiber ...) or CD storage channels, DVD ... an error correcting code used to transmit the message, according to the diagram in the following figure. We recall in the following some basic definitions.

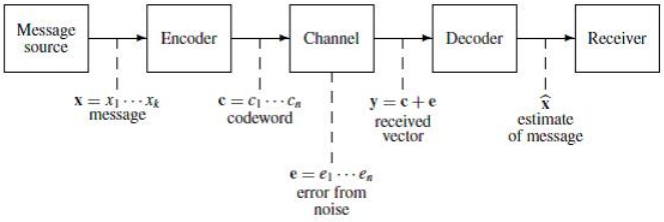

Figure 2.4: Image corresponding of channel communication .

Definition 2.1. *Let $\mathcal{A} = \{a_1, a_2, \cdots, a_q\}$ be a set of size q, which we refer to*

as a code alphabet and whose elements are called code symbols.

1. *A q-ary word of length n over $\mathcal{A}$ is a sequence $w = w_1, w_2, \cdots, w_n$ with each $w_i \in \mathcal{A}$ for all i. Equivalently, w may also be regarded as the vector $(w_1, \cdots, w_n)$.*

2. *A q-ary block code of length n over $\mathcal{A}$is a nonempty set $\mathcal{C}$ of q-ary words having the same length n.*

3. *An element of $\mathcal{C}$ is called a codeword in $\mathcal{C}$.*

4. *The number of codewords in $\mathcal{C}$, denoted by $|\mathcal{C}|$, is called the size of $\mathcal{C}$.*

5. *The (information) rate of a code $\mathcal{C}$ of length n is defined to be $(log_q|\mathcal{C}|)/n$.*

6. *A code of length n and size M is called an (n, M)-code.*

Example 2.2. A code over alphabet $\mathbb{F}_2 = \{0, 1\}$ is called binary code. $C_1 = \{000, 011, 101, 110\}$ is $(3, 4)-$ code.
A code over the alphabet $\mathbb{Z}_4 = \{0, 1, 2, 3\}$ is called quaternary code. $C_2 = \{000, 121, 202, 323\}$ is $(3, 4)-$ code.

Definition 2.3. *Let $\mathcal{C}$ is (n, M)-code over binary field $\mathbb{F}_2$, let $(c_1, \cdots, c_M)$ is basis of $\mathcal{C}$. Then the matrix*

$$G = \begin{pmatrix} \boxed{c_1} \\ \boxed{c_2} \\ \boxed{.} \\ \boxed{.} \\ \boxed{c_k} \end{pmatrix}$$

is called generator matrix of the code $\mathcal{C}$.

Definition 2.4. *Communication channel consists of a finite channel alphabet $\mathcal{A} = \{a_1, a_2, \cdots, a_q\}$ as well as set of forward channel probabilities $P(a_i \text{ received }, a_i \text{ sent })$, satisfying*

$$\sum_{j=1}^{q} P(a_i \text{ received }, a_i \text{ sent }) = 1$$

For all i, $\sum_{j=1}^{q} P(a_i \text{ received }, a_i \text{ sent })$ is the condition probability received, given a_i sent. See the following image.

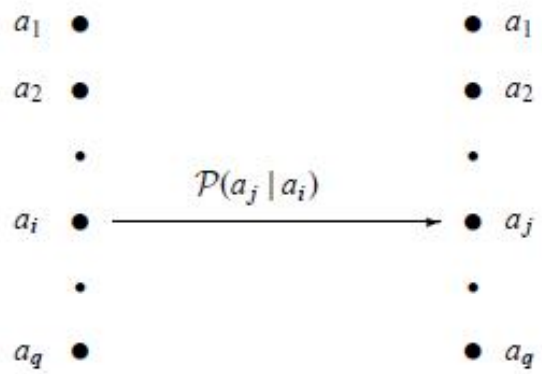

Figure 2.5: Image corresponding of channel communication over the alphabet $\mathcal{A}$.

Definition 2.5. *A communication channel is said to be memory less if the outcome of any one transmission is independent of the outcome of the previous transmissions; i.e., if* $c = c_1, c_2, \cdots, c_n$ *and* $x = x_1, x_2 \cdots, x_n$ *are words of length* n*, then*

$$P(x \text{ received}, c \text{ sent }) = \prod_{i=}^{n} P(x_i \text{received}, c_i \text{ sent })$$

Definition 2.6. *A* q*-ary symmetric channel is a memory less channel which has a channel alphabet of size* q *such that*

1. *each symbol transmitted has the same probability* $p(< 1/2)$ *of being received in error;*

2. *if a symbol is received in error, then each of the* $q-1$ *possible errors is equally likely.*

In particular, the binary symmetric channel (BSC) is a memory less channel which has channel alphabet 0, 1 and channel probabilities

$$P(1 \text{ received }, 0 \text{ sent }) = P(0 \text{ received }, 1 \text{ sent }) = p$$

$$P(0 \text{ received }, 0 \text{ sent }) = P(1 \text{ received }, 1 \text{ sent }) = p - 1$$

Thus, the probability of a bit error in a BSC is p. This is called the crossover probability of the BSC.

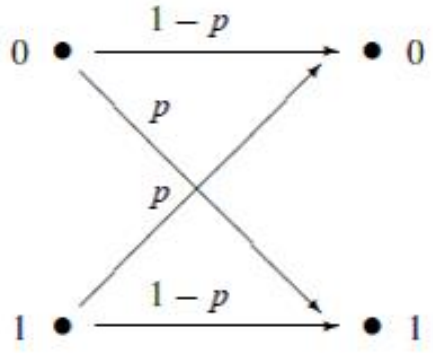

Figure 2.6: Image corresponding of Binary symmetric channel.

Chapter 3

Linear Codes over Finite Chain Ring

In this chapter, we introduce the finite chain ring and the structure of the linear codes and cyclic codes over finite chain ring. Let R be a finite chain ring, K its residue field, γ a fixed generator of the maximal ideal of R and ν nilpotency index of γ. The canonical projection $R[x] \longrightarrow K[x]$ and $R^n \longrightarrow K^n$ will be denote by $-$.

Definition 3.1. *A finite commutative ring with identity $1 \neq 0$ is called a finite chain ring if its ideals are linear ordered by inclusion.*

$$\langle 0 \rangle = \langle \gamma^{\nu} \rangle \subseteq \langle \gamma^{\nu-1} \rangle \subseteq \cdots \subseteq \langle \gamma^{1} \rangle \subseteq \langle \gamma^{0} \rangle = R$$

Lemma 3.2. *[30] Let $V \subseteq R$ be a set of representatives for the equivalence classes of R under congruence modulo γ, we can define V to be maximal subset of R with property that $\overline{r_1} \neq \overline{r_2}$. for all $r_1, r_2 \in R$, $r_1 \neq r_2$. Then:*

1. *for all$r \in R$ there are unique $r_0, \cdots, r_{\nu-1}$ such that $r = \sum_{i=0}^{\nu-1} r_i \gamma^i$*
2. *$|V| = |K|$*
3. *$|\gamma^j R| = |K|^{\nu-j}$ for all $0 \leq j \leq \nu - 1$*

Hensel lifting plays role important in the construction of cyclic code over R. Recall that , $f, g \in R[x]$ are coprime if they generator $R[x]$, i.e. there exist $u, v \in R[x]$ such that $fu + vg = 1$. Any $f \in R[x]$ which is not divisible by γ can be written as $f = uf_1$ where $u \in R[x]$is a unit and f_1 is a monic. The following theorem of Hensel lifting

Theorem 3.3. *[30] Let $g \in R[x]$ be a monic. Assume there are monic, pairwise coprime $f_1 \cdots f_k \in K[x]$ such that $\overline{g} = \prod_{i=1}^{k} f_i$. Then there are monic, pairwise coprime $g_1, \cdots g_k \in R[x]$ such that $g = \prod_{i=1}^{k} g_i$. and $\overline{g_i} = f_i$ for $1 \leq i \leq k$.*

Theorem 3.4. *[30] If $g \in R[x]$ is monic and $\overline{g}$ is square-free, then g factors uniquely into monic, coprime basic irreducible.*

Lemma 3.5. *[30] Let $f, g \in R[x]$. Then f, g are coprime if and only if $\overline{g}, \overline{f}$ are coprime.*

Recall that linear code of the length over finite chain ring R is an R-submodule of R^n. The following definition give the generator matrix of linear code over R.

Definition 3.6. *Let $\mathcal{C}$ be a code over R. A matrix G is called a generator matrix of $\mathcal{C}$ if rows of G span $\mathcal{C}$ and none them can be written as linear combination of the other rows of G. We say that G is a generator matrix in standard form if after a suitable permutation of the coordinates.*

$$G = \begin{pmatrix} I_{k_0} & A_{0,1} & A_{0,2} & A_{0,3} & \cdots & A_{0,\nu-1} & A_{0,\nu} \\ 0 & \gamma I_{k_1} & \gamma A_{1,2} & \gamma A_{1,3} & \cdots & \gamma A_{1,\nu-1} & \gamma A_{1,\nu} \\ 0 & 0 & \gamma^2 I_{k_2} & \gamma^2 A_{2,3} & \cdots & \gamma^2 A_{2,\nu-1} & \gamma^2 A_{2,\nu} \\ \cdot & \cdot & \cdot & \cdot & \cdots & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdots & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdots & \cdot & \cdot \\ 0 & 0 & 0 & 0 & \cdots & \gamma^{\nu-1} I_{k_{\nu-1}} & \gamma^{\nu-1} A_{\nu-1,\nu} \end{pmatrix} = \begin{pmatrix} A_0 \\ \gamma A_1 \\ \gamma^2 A_2 \\ \cdot \\ \cdot \\ \cdot \\ \gamma^{\nu-1} A_{\nu-1} \end{pmatrix} \tag{3.1}$$

say, where the columns are grouped into blocks of the sizes $k_0, k_1, \cdots, k_{\nu-1}, n - \sum_{i=0}^{\nu-1} k_i$ with $k_i \geq 0$. We associate to G the matrix

$$A = \begin{pmatrix} A_0 \\ \cdot \\ \cdot \\ \cdot \\ A_{\nu-1} \end{pmatrix}$$

of course if $k_i = 0$, the matrix $\gamma^i A_i$ and A_i are suppressed in G and A respectively. For given G and $1 \leq 1 \leq \nu - 1$, the matrices A_i are unique modulo $\gamma^{\nu-1}$ and therefore the $\overline{A}$'s are unique.

For any code and any $r \in R$, $(\mathcal{C} : r)$ is the submodule quotient $(\mathcal{C} : r) = \{e \in R^n | re \in \mathcal{C}\}$. We have the following definition.

Definition 3.7. *To any code $\mathcal{C}$ over R we associated the tower of the codes*

$$\mathcal{C} = (\mathcal{C} : \gamma^0) \subseteq \cdots (\mathcal{C} : \gamma^i) \subseteq \cdots \subseteq (\mathcal{C} : \gamma^{\nu-1})$$

over R and its projection to K,

$$\overline{\mathcal{C}} = \overline{(\mathcal{C} : \gamma^0)} \subseteq \cdots \overline{(\mathcal{C} : \gamma^i)} \subseteq \cdots \subseteq \overline{(\mathcal{C} : \gamma^{\nu-1})}$$

Theorem 3.8. *[30] Let $\mathcal{C}$ be a code of the length n. Then*

1. *the parameters $k_0, \cdots, k_{\nu-1}$ are the same for any generator matrix G in the standard form for $\mathcal{C}$,*
2. *any codewords $c \in \mathcal{C}$ can be written uniquely as*

   $$c = (v_0, v_1, \cdots, v_{\nu-1})G$$

   where $v_i \in (R/\gamma^{\nu-i}R)^{k_i} \cong (\gamma^i R)^{k_i}$,
3. *$|\mathcal{C}| = |K|^{\sum_{i=0}^{\nu-1}(\nu-i)k_i}$.*

Definition 3.9. *For any code $\mathcal{C}$ over R we define $k(\mathcal{C})$ to be the number of the rows in a generator matrix in standard form $\mathcal{C}$. We also define $k_i(\mathcal{C})$ to the number of the rows divisible by γ^i but not by γ^{i+1} in a generator matrix in standard form for $\mathcal{C}$ for $i = 0, \cdots, \nu - 1$. Clearly $k(\mathcal{C}) = \sum_{i=0}^{\nu-1} k_i(\mathcal{C})$. (Equivalently we define $k_0(\mathcal{C}) = dim(\overline{\mathcal{C}})$ and $k_i(\mathcal{C}) = dim(\overline{\mathcal{C} : \gamma^i}) - dim(\overline{\mathcal{C} : \gamma^{i-1}})$*

The dual code, denote as usual by $\mathcal{C}^\perp$ The following Theorem give the structure of the dual code $\mathcal{C}$.

Theorem 3.10. *[30] Let $\mathcal{C}$ be a code with generator matrix G in standard form.*

1. *If for $0 \leq i \leq j \leq \nu$, $B_{i,j} = -\sum_{k=i+1}^{j-1} B_{i,k} A^{tr}_{\nu-j,\nu-k} - A^{tr}_{\nu-j,\nu-i}$, then*

$$G = \begin{pmatrix} B_{0,\nu} & B_{0,\nu-1} & \cdots & B_{0,1} & I_{n-k(\mathcal{C})} \\ 0 & \gamma B_{1,\nu-1} & \cdots & \gamma I_{k_{\nu-1}(\mathcal{C})} & 0 \\ \cdot & \cdot & \cdots & \cdot & \cdot \\ \cdot & \cdot & \cdots & \cdot & \cdot \\ \cdot & \cdot & \cdots & \cdot & \cdot \\ \gamma^{\nu-1} B_{\nu-1,a} & \gamma^{\nu-1} I_{k_1(\mathcal{C})} & \cdots & \gamma^{\nu-1} I_{k_{\nu-1}} & \gamma^{\nu-1} A_{\nu-1,\nu} \end{pmatrix} = \begin{pmatrix} B_0 \\ \gamma B_1 \\ \cdot \\ \cdot \\ \cdot \\ \gamma^{\nu-1} B_{\nu-1} \end{pmatrix} \tag{3.2}$$

   is a generator matrix for $\mathcal{C}^\perp$ and a parity check matrix for $\mathcal{C}$.

2. $\overline{(\mathcal{C}^{\perp} : \gamma^i)} = (\overline{(\mathcal{C} : \gamma\nu - 1 - i)})^{\perp}$, $k_0(\mathcal{C}^{\perp}) = n - k(\mathcal{C})$ *and* $k_i(\mathcal{C}^{\perp}) = k_{\nu-i}(\mathcal{C})$ *for* $i = 1, \cdots, \nu - 1$

3. $|\mathcal{C}^{\perp}| = |R^n|/|\mathcal{C}|$ *and* $(\mathcal{C}^{\perp})^{\perp} = \mathcal{C}$.

Cyclic Codes

Recall that a code is cyclic if a cyclic shift of any codewords is a codewords. We assume that n is not divisible by the characteristic of K so that $x^n - 1$ *is square-free and by Theorem 3.4 The polynomial* $(x^n - 1)$ *has unique decomposition into distinct monic basic irreducible factors in* $R[x]$. *We denote* R_n *for the quotient ring of* $R[x]$ *by the ideal generated by* $x^n - 1$, K_n *is similarly defined. As usual, we identify* $(R^n, +)$ *and* $(R_n, +)$. *If* $f \in R[x]$ *has degree* $n - 1$ *or less, we identify* f *and its quotient class in* R_n. *A code over* R *of the length* n *is cyclic if and only if it is an ideal of* R_n. *Clearly if* $\mathcal{C}$ *is cyclic so* $\overline{\mathcal{C}}$. *For a cyclic code* $\mathcal{C}$, *in particular,* $(\mathcal{C} : \gamma^i) = \{e \in R_n | \gamma^i e \in \mathcal{C}\}$ *is the ideal quotient of* $\mathcal{C}$ *by* γ^i *in* R_n, *in particular,* $(\mathcal{C} : \gamma^i)$ *is cyclic for* $0 \leq i \leq \nu - 1$. *Then ideal generated by* $f_1, f_2, \cdots, f_s \in R_n$ *or* $f_1, f_2, \cdots, f_s \in K_n$ *will be denote by* $id(f_1, f_2, \cdots, f_s)$.
The following definition give the generating sets in standard form.

Definition 3.11. *We say that the set* $S = \{\gamma^{a_0} g_{a_0}, \gamma^{a_1} g_{a_1}, \cdots, \gamma^{a_s} g_{a_s}\}$ *is a generating set in standard from for cyclic code* $\mathcal{C} = id(S)$, *if* $0 \leq s \leq \nu$ *and*

1. $0 \leq a_0 < a_2 < \cdots < a_s < \nu$,

2. $g_{a_i} \in R[x]$ *is monic for* $i = 0, \cdots, s$,

3. $deg(g_{a_i}) > deg(g_{a_i+1})$ *for* $i = 0, \cdots, s - 1$,

4. $g_{a_s} | g_{a_{s-1}} | \cdots | x^n - 1$.

The following lemmas for to construct a unique generating set in standard form for a non-zero cyclic code.

Lemma 3.12. *[30] If* $\mathcal{C}$ *is a non-zero code, then* $\overline{(\mathcal{C} : \gamma^{\nu-1})} \neq \{0\}$.

Lemma 3.13. *Let* $S = \{\gamma^{a_0} g_{a_0}, \gamma^{a_1} g_{a_1}, \cdots, \gamma^{a_s} g_{a_s}\}$ *be a generating set in standard form for* $\mathcal{C} = id(S)$. *If* $i < a_0$ *then* $\overline{(:\gamma^i)} = \{0\}$, *otherwise* $\overline{(\mathcal{C} : \gamma^i)} = id(\overline{g}_{a_j})$ *where* j *is maximal with the property* $a_j \leq i$.

For any $f \in K[x]$ such that $f|x^n - 1$, by the Theorem 3.3 and 3.4 we have that the existence and unicity of a polynomial $g \in R[x]$ such that $\overline{g} = f$ and $g|x^n - 1$ since $x^n - 1$ is square-free in $K[x]$. The polynomial g will be called the Hensel lift of f.

Theorem 3.14. *[30] Any non-zero cyclic code $\mathcal{C}$ over R has a unique generating set in standard form.*

The following Theorem give the relation between generating standard form and generator matrix.

Theorem 3.15. *[30] Let $S = \{\gamma^{a_0} g_{a_0}, \gamma^{a_1} g_{a_1}, \cdots, \gamma^{a_s} g_{a_s}\}$ be generating set in standard form for the code $\mathcal{C} = id(S)$. Then*

1. *if $T = \bigcup_{i=0}^{s} \{\gamma^{a_i} g_{a_i} x^{d_{i-1}-d_i-1}, \cdots, \gamma^{a_i} g_{a_i}, \gamma^{a_i} g_{a_i}\}$ where $d_i = deg(g_{a_i})$ for $i = 0, \cdots, s$ and by convention $d_i - 1 = n$, $d_{s+1} = 0$, then defines a generator matrix of $\mathcal{C}$,*
2. *any $c \in \mathcal{C}$ can be uniquely written as $c = \sum_{j=0}^{s} h_j g_{a_j} \gamma^{a_j}$ with $h_j \in (R/R\gamma^{\nu - a_i})[x] \cong (R\gamma^{a_i})[x]$ and $deg(h_j) < d_{j-1} - d_j$,*
3. *$k_i(\mathcal{C}) = d_{j-1} - d_j$ if $i = a_j$ for some j, $k_i(\mathcal{C}) = 0$ otherwise, and $|\mathcal{C}|^{\sum_{j=0}^{s} (\nu - a_j)(d_{j-1} - d_j)}$.*

3.1 Linear Codes Over $\mathbb{Z}_4$

The study of linear codes over finite chain ring has received attention lately and many recent development of coding theory are defined on finite chain ring in particular over ring four elements. For those we propose the ring $\mathbb{Z}_4$. Other family of the cyclic codes over finite chain ring we will study in the following chapter with DNA codes.

Definition 3.16. *Every linear codes over $\mathbb{Z}_4$ contains set of $k_1 + k_2$ codewords $c_1 \cdots, c_{k_1}, c_{k_1+1}, \cdots, c_{k_1+k_2}$ such that every codewords in $\mathcal{C}$ is uniquely expressible in the forme*

$$\sum_{i=1}^{k_1} a_i c_i + \sum_{i=k_1+1}^{k_1+k_2}$$

where $a_i \in \mathbb{Z}_4$ for $1 \leq i \leq k_1$ and $a_i \in \mathbb{Z}_2$ for $k_1 + 1 \leq i \leq k_1 + k_2$. Furthermore, each c_i has at least one component equal to 1 or 3 each c_i has

all components equal to 0 and 2 for $k_1 + 1 \leq i \leq k_1 + k_2$. If $k_2 = 0$, then the code $\mathcal{C}$ is a free $\mathbb{Z}_4$-module.

We can be associated any vector $x \in \mathbb{Z}_4$ a different weight and a different distances, suppose that $n_a(x)$ denote the number of component of x equal to a for all $a \in \mathbb{Z}_4$. The Hamming weight of x is $wt_H(x) = n_1(x) + n_2(x) + n_3(x)$, the Lee weight is $wt_L(x) = n_1(x) + 2n_2(x) + n_3(x)$ and the Euclidean weight of x is $wt_E(x) = n_1(x) + 4n_2(x) + n_3(x)$. Thus components equaling 1 or 3 contribute 1 to each weight while contributes 2 to the Lee weight and 4 to the Euclidean weight. The Hamming, Lee and Euclidean distances between x and y are $d_H(x, y) = wt_H(x-y)$, $d_L(x, y) = wt_L(x-y)$ and $d_E(x, y) = wt_E(x-y)$, respectively.

3.2 Cyclic Codes over $\mathbb{Z}_4$

The cyclic codes over $\mathbb{Z}_4$ from an important family of linear codes over $\mathbb{Z}_4$. They have an algebraic structure (and sometimes combinatorial) interesting. In this part we give algebraic structure of the cyclic code of the odd length over $\mathbb{Z}_4$. Let $R_n = \mathbb{Z}_4[x]/\langle x^n - 1\rangle$. As usual, if $\mathcal{C}$ is the cyclic codes over $\mathbb{Z}_4$ of the length n, an element $c = (c_0, c_1, c_2, \cdots, c_{n-1})$ in $\mathcal{C}$ is defined with polynomial $c_0 + c_1x + c_2x^2 + \cdots, c_{n-1}x^{n-1}$ modulo $x^n - 1$. under this correspondence, a code is a cyclic code over $\mathbb{Z}_4$ if and only if it is an ideal in the ring R_n. The proper context for discussing factorization in $\mathbb{Z}_4[x]$ is not factoring polynomials into a product of irreducible polynomial but into a product of irreducible polynomial but into a product of the polynomials called basic irreducible polynomials. We defined $\mu : \mathbb{Z}_4[x] \to \mathbb{F}_2[x]$ by $\mu(f(x)) = f(x) mod 2$. μ is defined by $\mu(0) = \mu(2)$, $\mu(1) = \mu(3) = 1$, μ is a surjective ring homomorphism with $ker(2) = \{2s(x) | s(x) \in \mathbb{Z}_4[x]\}$. The map μ is called the reduction homomorphism .

Definition 3.17. *1. A polynomial $f(x) \in \mathbb{Z}_4[x]$ is irreducible in $\mathbb{Z}_4$ if whenever $f(x) = g(x)h(x)$ for two polynomials $g(x), h(x) \in \mathbb{Z}_4[x]$, one of $g(x)$ or $h(x)$ is a unit.*

2. *A polynomial $f(x) \in \mathbb{Z}_4[x]$ is basis irreducible in $\mathbb{Z}_4$ if its $\mu(f)$ is irreducible in $\mathbb{Z}_2[x]$.*

3. *An ideal I of a ring $\mathbb{Z}_4$ is called a primary ideal provided $ab \in I$ implies that $a \in I$ or $b^r \in I$ for some positive integer r.*

4. *A polynomial* $f(x) \in \mathbb{Z}_4[x]$ *is primary if the principal ideal*

$$\langle f(x) \rangle = \{f(x)g(x), g(x) \in \mathbb{Z}_4[x]\}$$

is primary ideal.

Lemma 3.18. *[16] Let* $f(x)$ *and* $g(x)$ *be polynomials in* $\mathbb{Z}_4[x]$*. Then* $f(x)$ *and* $g(x)$ *are coprime if and only if* $\mu f(x)$ *and* $\mu g(x)$ *are coprime polynomials in* $\mathbb{F}_2[x]$*.*

Since $\mathbb{F}_2[x]$ *is unique factorization domain,* $\mu(x^n - 1) = (x^n + 1) \in \mathbb{F}_[x]$ *has factorization* $g_1(x), g_2(x) \cdots, g_k(x)$ *into irreducible polynomials. These are pairwise coprime only if* n *is odd. By Hensel's Lemma if* n *is odd then there exist the factorization* $x^n - 1 = f_1(x)f_2(x), \cdots, f_k(x)$ *into pairwise coprime basic irreducible* $f_i(x) \in \mathbb{Z}_4[x]$ *such that* $\mu(f_i(x)) = g_i(x)$*, for get all* f_i *we use the method to Graeffe, see [16].*

Lemma 3.19. *[16] If* $x^n - 1 = f_1, f_2, \cdots, f_k$*, where are basic irreducible and pairwise-coprime polynomials, then this factorization is unique.*

The following theorem give the structure algebraic of the codes cyclic over $\mathbb{Z}_4$*.*

Theorem 3.20. *Suppose* $\mathcal{C}$ *is a cyclic codes over* $\mathbb{Z}_4$ *of the odd length* n*. The there exist unique polynomial* f*,* g *and* h *such that* $x^n - 1 = fgh$ *and* $\mathcal{C} = \langle fh, 2fg \rangle$*. Furthermore,* $\mathcal{C}$ *has type* $4^{degg}2^{degh}$*.*

1. *When* $h = 1$*,* $\mathcal{C} = \langle f \rangle$ *and* $|\mathcal{C}| = 4^{n-degf}$*.*
2. *When* $g = 1$*,* $\mathcal{C} = \langle 2f \rangle$ *and* $|\mathcal{C}| = 2^{n-degf}$*.*

Chapter 4

Greedy Construction of DNA Codes and New Bounds

In this chapter, we introduce the linear lexicode over finite chain ring. We construct linear codes over $\mathbb{Z}_4$ with bounded GC-content. The codes are obtained using a greedy algorithm over $\mathbb{Z}_4$. Further, upper and lower bounds are derived for the maximum size of DNA codes of length n with constant GC-content w and edit distance d are given. This chapter is organized as follows. In Section 2, some preliminary results are presented. Section 3 employs a greedy algorithm to obtain DNA codes with bounded GC-content, and in Section 4 DNA lexicodes are constructed with bounded edit distance. Upper and lower bounds on the edit distance are also presented. In addition, examples of DNA codes with bounded GC-content and edit distance are given.
The ring $\mathbb{Z}_4$ with element $\{0, 1, 2, 3\}$ is considered here with addition and multiplication modulo 4. It is a finite chain ring with maximal ideal $< 2 >$ and nilpotency index 2.
The elements $\{0, 1, 2, 3\}$ of $\mathbb{Z}_4$ are in one to one correspondence with the nucleotide DNA bases $\{A, T, C, G\}$ by the map ϕ such that $0 \to G$, $2 \to C$, $3 \to T$ and $1 \to A$.
We define the reverse of $x = (x_0 x_1 \cdots x_{n-1})$ to be $x^R = (x_{n-1} x_{n-2} \cdots x_1 x_0)$. The complement of the codeword $x = (x_0 x_1 \cdots x_{n-1})$ is the vector $x^C = (\hat{x}_0 \hat{x}_1 \cdots \hat{x}_{n-1})$. The reverse complement (also called the Watson-Crick complement) is $x^{RC} = (\hat{x}_{n-1} \hat{x}_{n-2} \cdots \hat{x}_1 \hat{x}_0)$. For $x \in \mathbb{Z}_4$, $\hat{x}$ is defined to be $\phi(\hat{x})$. A linear code $\mathcal{C}$ is said to satisfy the reverse constraint, respectively the reverse-complement constraint if for all $x \in \mathcal{C}$ we have $x^R \in \mathcal{C}$, respectively $x^{RC} \in \mathcal{C}$.

4.1 Lexicode over Finite Chain Ring R

Let $R = \{\alpha_1, \ldots, \alpha_m\}$ be a finite chain ring with nilpotency index e (defined in the chapiter one) and residue field $\mathbb{F}_{p^r}$. The free module R^n is a linear code over R with basis $B = \{\mathsf{b}_1, \ldots, \mathsf{b}_n\}$. With respect to this basis we give the following B-ordering; we recursively define a lexicographically ordered list $V_i = \mathsf{x}_1, \mathsf{x}_2, \ldots, \mathsf{x}_{p^{rei}}$ as follows

$$\begin{aligned} V_0 &:= 0, \\ V_i &:= V_{i-1}, \quad \alpha_1\mathsf{b}_i + V_{i-1}, \alpha_2\mathsf{b}_i + V_{i-1}, \ldots, \alpha_m\mathsf{b}_i + V_{i-1}, 1 \leq i \leq n. \end{aligned}$$

In this way $|V_i| = m^i$, and R^n is given by V_n. Assume now that we have a property P which can test if a vector $\mathsf{c} \in R$ is selected or not. Recall that a selection property P on V can be seen as a boolean valued function $P : V \to \{True, False\}$, that depends on one variable. Over a finite chain ring R, the property P is called a multiplicative property if $P[\mathsf{x}]$ is true implies $P[\beta\mathsf{x}]$ is true for all $\beta \in R^*$. Suppose that we have a selection property which is multiplicative over a finite chain ring R.

This is modified version of Algorithm [18], where we have two properties instead of one as in [18].

Algorithm 1

1. $\mathcal{C}_0 := 0; i := 1;$
2. *select the first vector $a_i \in V_i \backslash V_{i-1}$ such that $P[a_i + c]$, $P[2a_i + c]$, $d_H(a_i, c) > d$ and $d_H(2a_i, c) > d$ for all $c \in \mathcal{C}_{i-1}$;*
3. *if such an a_i exists, then $\mathcal{C}_i := \mathcal{C}_{i-1}, a_i + \mathcal{C}_{i-1}, 2a_i + \mathcal{C}_{i-1}, 3a_i + \mathcal{C}_{i-1}$; otherwise $\mathcal{C}_i := \mathcal{C}_{i-1}$;*
4. *$i := i + 1$; return to 2.*

For $0 < i \leq n$, the codes C_i are forced to be linear because we take all linear combinations of the selected vectors $\mathsf{a}_{i1}, \ldots, \mathsf{a}_{il}$; $l \leq i$. The codes C_i have a generating set formed by the selected vectors $\mathsf{a}_{i1}, \ldots, \mathsf{a}_{il}$.

Considering the greedy algorithm for finite fields, a natural question that arises is, can a vector $\mathsf{x} \in V_i \setminus V_{i-1}$ exist with $P[\mathsf{x} + \mathsf{c}]$ for all $\mathsf{c} \in C_i$ and $\mathsf{x} \notin C_i$. The following lemma, shows that such a vector does not exist.

Lemma 4.1. *[18] Let R be a finite chain ring with maximal ideal $\langle\gamma\rangle$ and nilpotency index e. Let P be a multiplicative property over R, and let $\mathsf{a_i} \in V_i$ be such that $P[\gamma^j \mathsf{a_i} + \mathsf{c}]$ for all $0 \leq j \leq e-1$ and for all $\mathsf{c} \in C_{i-1}$, $i \geq 1$. Then every $\mathsf{x} \in V_i \setminus V_{i-1}$ satisfying $P[\gamma^j \mathsf{x} + \mathsf{c}]$, for all $0 \leq j \leq e-1$ and all $\mathsf{c} \in C_i$, is in C_i.*

4.1.1 Construction of Lexicodes over $\mathbb{Z}_4$

The construction of lexicodes over $\mathbb{Z}_4$ given in [18] is now reviewed. A linear code $\mathcal{C}$ of length n over $\mathbb{Z}_4$ is an additive code over $\mathbb{Z}_4^n$. Thus $\mathbb{Z}_4^n$ is a linear code over $\mathbb{Z}_4$ with basis $B = \{b_1 \cdots b_n\}$. With respect to this basis, we recursively define a lexicographically ordered list $V_i = x_1, x_2, \cdots, x_{4^i}$ as follows

$$V_0 := 0$$

$$V_i := V_{i-1}, b_i + V_{i-1}, 2b_i + V_{i-1}, 3b_i + V_{i-1}, 1 \leq i \leq n.$$

In this way $|V_i| = 4^i$, and $\mathbb{Z}_4^n$ can be associated with V_n. Assume now that we have a property P which can test if a vector $c \in \mathbb{Z}_4^n$ is selected or not. The selection property P on V can be seen as a boolean valued function

$$P : V \to \{True, False\},$$

that depends on one variable. Over $\mathbb{Z}_4$, the property P is called a multiplicative property if $P[x]$ is true implies $P[3x]$ is true. The following greedy algorithm provides lexicodes over $\mathbb{Z}_4^n$ [18].

Algorithm 1

1. $\mathcal{C}_0 := 0; i := 1;$
2. *select the first vector $a_i \in V_i \backslash V_{i-1}$ such that $P[2a_i + c]$ for all $c \in \mathcal{C}_{i-1}$;*
3. *if such an a_i exists, then $\mathcal{C}_i := \mathcal{C}_{i-1}, a_i + \mathcal{C}_{i-1}, 2a_i + \mathcal{C}_{i-1}, 3a_i + \mathcal{C}_{i-1}$; otherwise $\mathcal{C}_i := \mathcal{C}_{i-1}$;*
4. *$i := i + 1$; return to 2.*

For $0 < i \leq n$, the code $\mathcal{C}_i$ is forced to be linear because all linear combinations of the selected vectors $a_{i1}, \cdots, a_{il}$, $l \leq i$, are taken. The code $\mathcal{C}_i$

has a 'basis' formed from $a_{i1}, \cdots, a_{il}$, so we have a nested sequence of linear codes

$$0 = \mathcal{C}_0 \subseteq \mathcal{C}_1 \subseteq \cdots \subseteq \mathcal{C}_n.$$

$\mathcal{C}_n$ is the lexicode and is denoted $\mathcal{C}_n = \mathcal{C}(B, P)$ where B is the ordering and P is the selection property. We have the following result.

Theorem 4.2. *([18, Theorem 4]) For any basis B of R^n and any multiplicative selection criterion P, the lexicode $\mathcal{C}(B, P)$ is linear and $P[x]$ holds for each codeword $x \neq 0$.*

4.2 A Greedy Algorithm for Bounded GC-Content DNA Codes

In this Section we construct DNA codes with bounded GC-content using Algorithm 1. We begin with the following definition.

Definition 4.3. *Let $\mathcal{C}$ be a linear code over $\mathbb{Z}_4^n$. The GC-content of a codeword $x \in \mathcal{C}$, denoted by $GC(\phi(x))$, is the number of occurrences of G and C in $\phi(x)$*

$$GC(\phi(x)) = |\{1 \leq i \leq n;\, \phi(x)_i \in \{G, C\}\}| = w_{GC}(\phi(x)).$$

We say that a subset $\mathcal{C}$ of $\mathbb{Z}_4^n$ satisfies the bounded GC-content constraint if there exists a positive integer w such that $GC(\phi(x)) \geq w$, $\forall\, x \in \mathcal{C}$.

Remark 4.4. Definition 4.3 differs from the conventional definition [18, 20]. The bounded GC-content constraint ensures that all codewords have a hybridization energy below some threshold, which results in stable DNA strands.

Proposition 4.5. *The property $P_1[x]$ is true if and only if $w_{GC}(\phi(x)) \geq w$ is a multiplicative property over $\mathbb{Z}_4$.*

Proof. *Let $x \in \mathbb{Z}_4^n$ such that $w_{GC}(\phi(x)) \geq w$. Multiplying the vector x by 3 does not change the number of 0's and 2's. This gives that $w_{GC}(\phi(3x)) = w_{GC}(\phi(x)) \geq w$, and the result follows.* □

4.2.1 Construction Results

In this subsection, construction results are presented for linear codes over $\mathbb{Z}_4$ with bounded GC-content. In this case, the verification step for $w_{GC}(\phi(2x)) \geq w$ in Algorithm 1 can be eliminated. This is because for $x \in \mathbb{Z}_4^n$, $w_{GC}(\phi(x)) \geq w$ implies that $w_{GC}(\phi(2x)) \geq w$, and this improves the speed of the algorithm. Some of these codes attain upper bound (5) given in [24, Proposition 1]. Furthermore, the codes obtained are linear as opposed to those in [36]. Table 4.1 gives DNA lexicodes over $\mathbb{Z}_4^n$ obtained using the selection property $P_1[x]$ ($w_{GC}(\phi(x)) \geq w$). The DNA code strands corresponding to the first and second codes in Table 1 are given in Tables 4.2 and 4.3, respectively.

Let C lexicode of the length 10 *generated by*

$$\begin{pmatrix} 2 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 3 & 2 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 3 & 2 & 3 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

Using the programming magma and the map ϕ for obtained the parameter of the DNA code $\phi(C)$ in the following table

number of codeword	$n_G(v)$	$n_{A,T}(v)$	$n_C(v)$	$GC(v)$
1	*10*	*0*	*0*	*10*
7	*6*	*0*	*4*	*10*
42	*5*	*4*	*1*	*6*
14	*3*	*4*	*3*	*6*

4.3 DNA Codes and Edit Distance

The edit distance has been used for biological computation, in particular for two types of genetic mutation. The first is the substitution of nucleotides and consists of two possible mutations:

- *Transition: a purine is replaced by a purine ($A \leftrightarrow G$) or a pyrimidine is replaced by a pyrimidine ($T \leftrightarrow C$).*
  Transversion: a purine is replaced by a pyrimidine or the reverse (eg. $A \leftrightarrow C$).

- *Modification using insertions and deletions.*

Table 4.1: DNA Lexicodes over $\mathbb{Z}_4^n$ Obtained using the Selection Property $P_1[x]$ $(w_{GC}(\phi(x)) \geq w)$

n	w	d_H	Basis of $\mathbb{Z}_4$	Basis of $\mathcal{C}(B, P)$
8	4	4	Canonical basis	21111000
				13210100
				32310010
10	6	4	Canonical basis	2111100000
				1321010000
				3231001000
10	10	1	Canonical basis	2000000000
				0200000000
				0020000000
				0002000000
				0000200000
				0000020000
				0000002000
				0000000200
				0000000020
				0000000002
12	12	1	Canonical basis	200000000000
				020000000000
				002000000000
				000200000000
				000020000000
				000002000000
				000000200000
				000000020000
				000000002000
				000000000200
				000000000020
				000000000002

Table 4.2: DNA Code Strands Corresponding to the Linear Code in the First Row of Table 1

GGGGGGGG	GGGGCCCC	CCCCGGGG	GAAAACCC
GGCCGGCC	CCGGCCGG	CGCGCGCG	GATTTCCC
GGGCCCGC	GGGCCCCG	GGGAAAAC	AACCCGTT
CAAAAGGG	AAAAGGGC	GGGAAACT	TTAACCCG
TGGGAAAC	CTGGGAAA	CTAAAGGG	CCCAAAGT
GGAAACTG	ACTGGGAA	GAAACTGG	TGGGCTTT
AAACTGGG	AACTGGGA	GGGAACTT	CCCGATTT
TTGGGAAC	ACTTGGGA	TGGGAACT	TTCCCGAA
CTTGGGAA	AACTTGGG	GGGCTTAA	AATTCCCG
GGAACTTG	GAACTTGG	AGGGCTTA	TAAACCCG
AAGGGCTT	TTAAGGGC	CTTAAGGG	TTGGGCTT
GCTTAAGG	TAAGGGCT	GGCTTAAG	GCCCTTTT
ATTGGGCA	TTGGGCAA	GGGACTTT	GACCCTTT
TTTGGGAC	TTGGGACT	TGGGACTT	CAATTCCG
CTTTGGGA	TTTACGGG	GACTTTGG	GAACCCTT
CATTTGGG	GGACTTTG	GGGCTTTT	GAAACCCT

Table 4.3: DNA Code Strands Corresponding to the Linear Code in the Second Row of Table 1

GGGGGGGGGG	TCTAGGAGGG	GGCCGGCGGG	ACATGGTGGG
ATCAGAGGGG	GAACGAAGGG	TTGTGACGGG	CATGGATGGG
CCGCGCGGGG	AGTTGCAGGG	GCCGGCCGGG	TGAAGCTGGG
TACTGTGGGG	CTAGGTAGGG	AAGCGTCGGG	GTTCGTTGGG
CAAAAGGGGG	ATGCAGAGGG	GATTAGCGGG	TTCGAGTGGG
TGTGAAGGGG	CCCAAAAGGG	AGACAACGGG	GCGTAATGGG
GTATACGGGG	TAGGACAGGG	CTTAACCGGG	AACCACTGGG
ACTGATGGGG	GGCAATAGGG	TCACATCGGG	CGGTATTGGG
GCCCCGGGGG	TGATCGAGGG	CCGGCGCGGG	AGTACGTGGG
AAGTCAGGGG	GTTGCAAGGG	TACACACGGG	CTACCATGGG
CGCGCCGGGG	ACAACCAGGG	CGGCCCCGGG	TCTTCCTGGG
TTGACTGGGG	CATCCTAGGG	ATCTCTCGGG	GAAGCTTGGG
CTTTTGGGGG	AACGTGAGGG	GTAATGCGGG	TAGCGGTGGG
TCAGTAGGGG	CGGATAAGGG	ACTCTACGGG	GGCTTATGGG
GATATCGGGG	TTCCTCAGGG	CAATTCCGGG	ATGGTCTGGG
AGACTTGGGG	GCGTTTAGGG	TGTGTTCGGG	CCCATTTGGG

In this section, we consider the edit distance in the greedy algorithm in order to find large sets of DNA codewords of length n with given w_{GC} and minimum edit distance d. We begin by providing a definition of edit distance which follows the presentation in [31].

Let $\mathcal{A}$ and $\mathcal{B}$ be finite sets of distinct symbols and let $x^t \in \mathcal{A}^t$ denote an arbitrary string of length t over $\mathcal{A}$. The string edit distance is characterized by a triple $< \mathcal{A}, \mathcal{B}, c >$ consisting of the finite sets $\mathcal{A}$ and $\mathcal{B}$, and the primitive function $c : E \to \mathbb{R}_+$ where $\mathbb{R}_+$ is the set of nonnegative reals, $E = E_s \cup E_d \cup E_i$ is the set of primitive edit operations, $E_s = \mathcal{A} * \mathcal{B}$ is the set of substitutions, $E_d = \mathcal{A} * E$ is the set of deletions, and $E_i = E \times \mathcal{B}$ is the set of insertions. Each triple $< \mathcal{A}, \mathcal{B}, c >$ induces a distance function $d_c : \mathcal{A}^* \times \mathcal{B}^* \to \mathbb{R}_+$ that maps a string x^t to a nonnegative value [31].

Definition 4.6. *The edit distance $d_c(x^t, y^v)$ between two strings $x^t \in \mathcal{A}^t$ and $y^v \in \mathcal{B}^v$ is defined recursively as*

$$d_c(x^t, y^v) = \min \begin{cases} c(x^t, y^v) + d_c(x^{t-1}, y^{v-1}), \\ c(x^t, \epsilon) + d_c(x^{t-1}, y^v), \\ c(\epsilon, y^v) + d_c(x^t, y^{v-1}); \end{cases}$$

where $d_c(\epsilon, \epsilon) = 0$ and ϵ denotes the empty string of length n.

The edit distance constraint for a DNA code $\mathcal{C}$ is $d_c(x, y) \geq d \forall x, y \in \mathcal{C}$, $x \neq y$, for some prescribed minimum edit distance d. The edit distance constraint can reduce non-specific hybridization between distinct codewords, as well as allow for the correction of insertion, deletion and substitution errors in codewords.

Proposition 4.7. *The property $P_2[x]$ is true only if $d_c(\phi(x), \phi(y)) \leq w$ is a multiplicative property over $\mathbb{Z}_4$.*

Proof. *Let $x \in {\mathbb{Z}_4}^n$ and $y \in {\mathbb{Z}_4}^n$. Multiplying x by 3 and y by 3 does not change the number of 0's and 2's. Therefore the number of 1's and 3's also does not change, so*

$$n_1(x) + n_0(x) + n_2(x) + n_3(x) = n_1(3x) + n_0(3x) + n_2(3x) + n_3(3x).$$

This also holds for y and thus $d_c(x, y) = d_c(3x, 3y)$. □

Table 4.4: DNA Lexicodes over $\mathbb{Z}_4^n$ Obtained using the Selection Property $P_2[x]$ $(d_c(\phi(x), \phi(y)) \leq m)$

n	$\phi(x)$	m	w_{GC}	Basis of $\mathbb{Z}_4$	Basis of $\mathcal{C}(B, P)$
4	GGGG	1	4	Canonical basis	2222 2202 2220 2022
4	GCGC	2	4	Canonical basis	2020 0022 0220 2222

Now we use Algorithm 1 to construct linear codes over $\mathbb{Z}_4$ with GC-content bounded by w and edit distance $d_c(\phi(x), \phi(y))$ such that $x \in \mathbb{Z}_4^*$ and $y \in \mathbb{Z}_4^*$. The results are given in Table 4.

We use the programming magma and the map ϕ for obtained the parameter of the DNA code $\phi(\mathcal{C})$ in the following table

number of the codeword	$n_G(v)$	$n_{A,T}(v)$	$n_C(v)$	$GC(v)$
1	*4*	*0*	*0*	*4*
4	*3*	*0*	*1*	
6	*2*	*0*	*2*	
4	*1*	*0*	*3*	
1	*0*	*0*	*4*	

4.3.1 Upper and Lower Bounds

Let $A_4(n, d)$ be the maximum size of a code over $\mathbb{Z}_4$ with length n and minimum edit distance d. Let $A_4^{GC}(n, d, w)$ be the maximum size of a DNA code with length n, minimum edit distance d, and fixed GC weight w. Further, let $A_4^{R,GC}(n, d, w)$, respectively $A_4^{RC,GC}(n, d, w)$ be the maximum size of a DNA code with length n, minimum edit distance d, and fixed GC weight w, that satisfies the reverse constraint, respectively the reverse-complement constraint. The purpose of this section is to give upper and lower bounds on these quantities. We have the following theorem.

Theorem 4.8. *For $n > 0$ with $0 \leq d \leq n$ and $0 \leq w \leq n$, the following results hold.*

$$A_4{}^{GC}(n, d, 0) = A_2(n, d), \tag{4.1}$$

$$A_4{}^{GC}(n, d, w) = A_4^{GC}(n, d, n - w), \tag{4.2}$$

and if $w = n/2$ then

$$A_4^{GC}(n, d, w) = 4. \tag{4.3}$$

Proof. *The analogous result for DNA codes with GC-content and Hamming distance was given in [24]. The corresponding proof is employed here for the edit distance.*
(1): Let $\mathcal{C}$ be a linear code over $\mathbb{Z}_4^n$ with $w_{GC}(\phi(\mathcal{C})) = 0$. Then $\mathcal{C}$ contains only 0's and 1's, so $\mathcal{C}$ can be considered aa binary code which gives $A_4{}^{GC}(n, d, 0) = A_2(n, d)$.
(2): Since $w_{GC}(\phi(\mathcal{C})) = n - w_{AT}(\phi(\mathcal{C}))$, interchanging the A's with C's and T's with G's gives $w_{GC}(\phi(\mathcal{C})) = n - w$, so that $A_4^{GC}(n, d, w) = A_4^{GC}(n, d, n - w)$.
(3): Since $A_4^{RC,GC}(n, d, w) \leq A_4^{GC}(n, d, w)$, by [38, Theorem 5] we have that $A_4^{RC,GC}(n, d, w) = 2$. Then $4 \leq A_4^{GC}(n, d, w)$, and by the pigeonhole principle $A_4^{GC}(n, d, w) \geq 4$, so that $A_4^{GC}(n, d, w) = 4$. □
The following theorem shows the relationship between $A_4^{GC}(n, d, w)$ and $A_4^{rc,GC}(n, d, w)$.

Theorem 4.9. *$1 \leq d \leq n$ and $0 \leq w \leq n$, $A_4^{rc,GC}(n, d, w) \leq \frac{1}{2}.A_4^{GC}(n, d, w)$.*

Proof. *Let $\mathcal{C}$ be a DNA lexicode of $A_4^{GC}(n, d, w)$, that satisfies reverse-complement constraint. Assume that the size of $\mathcal{C}$ is $|\mathcal{C}|$. Let $\mathcal{C}^{rc} = x^{rc} | c \in \mathcal{C}$, it is clear that $\mathcal{C}$ is DNA lexicode of $A_4^{rc,GC}(n, d^{'}, w^{'})$, by Waston-Crick. We have that $w^{'} = w$ and $d^{'} = d$. Then $\mathcal{C}^{rc}$ is DNA lexicode of $A_4^{rc,GC}(n, d, w)$, that satisfies the reverse-complement constraint.*
Let $\mathcal{C}^{'} = \mathcal{C} \cup \mathcal{C}^{rc}$, then we have $\mathcal{C}^{'}$ is (n, d, w) DNA lexicode of length n, minimum edit distance d and fixedGC-content. It is obvious, $\mathcal{C}^{rc} \cap \mathcal{C} = \Phi$, thus we have $|\mathcal{C}^{'}| = 2|\mathcal{C}|$, then $|\mathcal{C}^{'}| \leq A_4^{GC}(n, d, w)$. By consequently $A_4^{rc,GC}(n, d, w) = |\mathcal{C}| = \frac{1}{2}|\mathcal{C}^{'}| \leq \frac{1}{2} A_4^{GC}(n, d, w)$ □
We have the following relationship between the GC-content of a code and the code size over the alphabet $\{A, T, C, G\}$.

Proposition 4.10.

$$A_4^{GC}(n, d, w) \geq A_4^{GC}(n + 1, d + 1, w). \tag{4.4}$$

$$A_4^{GC}(n,d,w) \geq A_4^{GC}(n+1,d,w)/4. \tag{4.5}$$

Proof. *The analogous result for DNA codes with unrestricted GC-content and Hamming distance was given in [28]. The corresponding proof is employed here for the edit distance.*
(4): A $(n, A_4^{GC}(n+1,d+1,w),d,w)$ *code can be obtained from a* $(n+1, A_4^{GC}(n+1,d+1,w),d+1,w)$ *code by removing a symbol from each codeword such that their GC-content is preserved.*
(5): If all the codewords in a $(n+1, A_4^{GC}(n+1,d,w),d,w)$ *code are partitioned into four subsets according to the first symbol, one of the subsets will have size at least* $A_4^{GC}(n+1,d,w)/4$ *and thus is a* $(n+1, A_4^{GC+}(n+1,d,w)/4,d,w)$ *code. By removing the (common) symbol from all codewords in the largest subset, a* $(n, A_4^{GC+}(n+1,d,w)/4,d,w)$ *code is obtained.* □
We have the following relationship between the GC-content of a reverse code and the code size over the alphabet $\{A,T,C,G\}$.

Proposition 4.11.

$$A_4^{GC,R}(n-1,d,w) \leq A_4^{GC,R}(n,d,w) \leq A_4^{GC,R}(n,d-1,w). \tag{4.6}$$

$$A_4^{GC,R}(n-1,d,w) \geq A_4^{GC,R}(n,d,w)/4. \tag{4.7}$$

Proof. *The analogous result for DNA codes with unrestricted GC-content and Hamming distance was given in [28]. The corresponding proof is used here for the edit distance.*
(6): By the construction of codes over $\mathbb{Z}_4$, *we obtain* 4^n *codewords of length* n *and* 4^{n-1} *codewords of length* $n-1$, *and the result follows.*
(7): The codewords of a $\mathcal{C}(n, A_4^{GC,R}(n,d,w),d)-$*code over* $\mathbb{Z}_4$ *can be partitioned into four subsets denoted* C_1,C_2,C_3,C_4 *such that the size of subset* C_1 *is at least* $A_4^{GC,R}(n,d,w)/4$ *and* C_1 *is a* $(n, A_4^{GC,R}(n,d,w)/4,d)$ *code. Removing a symbol from the codewords of* C_1 *such that the distance* d *and weight* w *are maintained, we obtain a* $(n-1, A_4^{GC,R}(n,d,w),d)$ *code, and the result follows.* □

Proposition 4.12. *For* $0 \leq d \leq n$ *and* $0 \leq w \leq n$

$$A_4^{GC,RC}(n,d,w) = A_4^{GC,R}(n,d,w),$$

if n *is even, and*

$$A_4^{GC,R}(n,d+1,w) \leq A_4^{GC,RC}(n,d,w) \leq A_4^{GC,R}(n,d-,w),$$

if n *is odd.*

Proof. *The analogous result for DNA codes with unrestricted GC-content and edit distance was given in [24]. The corresponding proof is employed here for the edit distance. Given a set of codewords of length n, if we replace all entries in any subset of the positions by their complement, the GC-content of these codewords is preserved, as well as the edit distance between any pair of codewords. The edit distance between a codeword and the reverse or reverse-complement of the other codewords is not in general preserved, but if n is even and the first $n/2$ coordinates of each codeword x_i are replaced by their complements to form a new codeword y_i, then $d_c(x_i, x^R) = d_c(y_i, y_j^{RC})$ for all codewords x_i and x_j. Similarly, if n is odd and the first $(n-1)/2$ coordinates of each codeword x_i are replaced by their complements to form y_i, then $|d_c(x_i, x_j^R) - d_c(y_i, y_j^{RC})| \leq 1$.* □

Chapter 5

DNA Codes with Optimal Thermodynamic and Combinatorial Properties

In this chapter, we construct linear codes over $\mathbb{Z}_{16}$ with bounded GC content which have optimal thermodynamic and combinatorial properties. The codes are obtained using a greedy algorithm with selection properties and constrained deletion similarity distance D. Upper and lower bounds are derived for the maximum size of DNA codes over the alphabet $\{A, G, C, T\}$ based on the GC content and D.
The ring $\mathbb{Z}_{16}$ with elements $\{0, 1, 2, \ldots, 15\}$ is considered with addition and multiplication modulo 16. This is a finite chain ring with maximal ideal $< 2 >$ and nilpotency index 4. The rank of a code $\mathcal{C}$, denoted $rank(\mathcal{C})$, is the minimum number of generators of $\mathcal{C}$. As the ring $\mathbb{Z}_{16}$ has cardinality 16, one can construct a one-to-one correspondence between the elements of $\mathbb{Z}_{16}$ and the 16 DNA base pairs over the alphabet $\{A, G, C, T\}^2$ using the map ϕ given in Table 1. The weights of the pairs containing A and T and those containing G and C, denoted w_{AT} and w_{GC}, respectively, can be used to obtain stable DNA codes over $\phi(\mathbb{Z}_{16})$.

The hybridization energy of the duplex can be modeled as a function of the so-called neighborhood energy of the nucleotides. For a pair $a, b \in ADN = \{A, C, G, T\}$, the neighborhood energy is given by

$$w(a, b) = \Delta G(a, b) = \Delta H(a, b) - T\Delta S(a, b),$$

where $\Delta H(a, b)$ and $\Delta S(a, b)$ are the temperature independent enthalpy and

entropy, respectively. The pairs $(a, b) \in ADN^2$ *are also called stacked pairs.*

Table 5.1: Nearest Neighborhood Thermodynamic Value for Stacked Pairs [10]

$w(a,b)$	$b = A$	$b = C$	$b = G$	$b = T$
$a = A$	1.02	1.46	1.29	0.88
$a = C$	1.46	1.83	2.17	1.29
$a = G$	1.32	2.24	1.83	1.46
$a = T$	0.60	1.32	1.46	1.02

$\Delta G(a, b)$ *for these pairs at a temperature of* 310° *is given in Table 2. For* $x = x_0x_1 \dots x_{n-1} \in ADN^n$ *and* $y = y_0y_1 \dots y_{n-1} \in ADN^n$, *define* $S_w(x, y) = \sum_{i=1}^{n-1} s_i^w(x, y)$, *where*

$$s_i^w(x, y) = \min \begin{cases} w(a, b), if x_i = y_i = a, x_{i+1} = y_{i+1} = b \\ 0, otherwise; \end{cases}$$

The quantity $S_w(x, y)$ *is called the additive stem similarity between* x *and* y. *The hybridization energy between* x *and* y *is [15]*

$$E(x, y) = S_w(x, y^{rc}).$$

If the neighborhood energy of the pair base $w(a, b) = 1$, *the DNA hybridization energy* $\varepsilon(x, y)$ *of two DNA strands* x *and* y *can be approximated by the length of the longest subsequence in one strand for which the reverse complement is in the other [14]. The maximum number of Watson-Crick bonds (complementary pairs) which may be formed between two oppositely oriented strands is defined as*

$$\varepsilon(x, x^{rc}) = \max_y \varepsilon(x, y^{rc}) = \max_y \varepsilon(y^{rc}, x) = \varepsilon(x^{rc}, x) = n. \tag{5.1}$$

Example 5.1. If $x = TTCG$ and $x^{rc} = CGAA$, then $\varepsilon(x^{rc}, x) = 4$.

Let x *and* y *be sequences over* $\phi(R^n)$. *The longest subsequence occurring in both sequences is called the deletion similarity between* x *and* y *denoted by* $S^\alpha(x, y)$ *[14]. Thus the deletion similarity* $S^\alpha(x, y)$ *determines the number*

of base pair bonds in the hybridization energy between x and y, i.e. the hybridization energy $\varepsilon(x, y^{rc})$ satisfying (5.1) can be defined as

$$\varepsilon(x, y^{rc}) = \varepsilon(x^{rc}, y) = S^{\alpha}(x, y) = S^{\alpha}(y, x). \tag{5.2}$$

Let D, $1 \leq D \leq n-1$, be an integer. A DNA code $\mathcal{C}$ is called a DNA code of deletion similarity distance D, denoted an (n, D) DNA code, if it satisfies

$$S^{\alpha}(x, y) \leq n - D - 1. \tag{5.3}$$

Example 5.2. The code

$$\mathcal{C} = \{CTTTCTGA, GAAAGACT, GACATTCT, CTGTAAGA\}$$

is an (n, D) DNA code of length $n = 8$ and deletion similarity distance $D = 5$, because the subsequence $Z = CT$ of length $n - D - 1 = 2$ is the longest common subsequence between any pair of strands in $\mathcal{C}$.

Table 5.2: DNA Base Pairs and the Corresponding Elements of $\mathbb{Z}_{16}$

AA	2	GG	0	CT	5	CA	3
AT	6	TG	1	CC	12	GT	15
TT	10	AG	7	TC	9	GC	8
TA	14	GA	13	AC	11	CG	4

Table 6.2 gives a map ϕ which is a one-to one correspondence between the elements of $\mathbb{Z}_{16}$ and the DNA nucleotide base pairs. This map can be used to obtain DNA codes with large GC content and thus high hybridization energy. For this we require the following definitions.

Definition 5.3. *For $x \in \mathbb{Z}_{16}^n$, denote by w_{AT} the number of occurrences of the base pairs AA, AT, TA, TT in $\phi(x)$*

$$w_{AT}(\phi(x)) = |\{1 \leq i \leq n, \phi(x_i) \in \{AA, AT, TA, TT\}\}|.$$

Definition 5.4. *For $x \in \mathbb{Z}_{16}^n$, denote by w_{GC} the number of occurrences of the base pairs GG, CG, GC, CC in $\phi(x)$*

$$w_{GC}(\phi(x)) = |\{1 \leq i \leq n, \phi(x_i) \in \{GG, CG, GC, CC\}\}|.$$

The weight of the base pairs w_{AT} and w_{GC} is used in the greedy algorithm over the ring $\mathbb{Z}_{16}$ given in the next section to obtain DNA codes with large GC content and optimal thermodynamic and combinatorial properties.

Remark 5.5. The number of occurrences of GG, CG, GC, CC and AA, AT, TA, TT can be counted using the bijection ϕ and the complete weight enumerator over $\mathbb{Z}_{16}$

$$cwe_{\phi(\mathcal{C})}(W_0, W_1, \ldots, W_{15}) = \sum_{\phi(c)} \prod_{i=0}^{15} W_i^{n_i(\phi(x))},$$

where $n_i(\phi(x))$ is the number of coordinates with DNA base pair $\phi(x)$.

Definition 5.6. *Let $\mathcal{C}$ be a linear code over $\mathbb{Z}_{16}^n$. The GC content of a codeword $x \in \mathbb{Z}_{16}^n$, denoted $GC(\phi(x))$, is the number of occurrences of G and C in $\phi(x)$*

$$GC(\phi(x)) = |\{1 \leq i \leq n;\, \phi(x)_i \in \{G, C\}\}|.$$

We say that a subset $\mathcal{C}$ of $\mathbb{Z}_{16}^n$ satisfies the constant GC content constraint if

$$GC(\phi(x)) = w,\, \forall x \in C.$$

5.1 Construction of DNA Codes

A linear code $\mathcal{C}$ of length n over $\mathbb{Z}_{16}$ is an additive code over $\mathbb{Z}_{16}^n$ with basis $B = \{b_1, \ldots, b_n\}$. With respect to this basis, we recursively define a lexicographically ordered list $V_i = x_1, x_2, \ldots, x_{16^i}$ as follows

$$V_0 := 0$$

$$V_i := V_{i-1}, b_i + V_{i-1}, 2b_i + V_{i-1}, \ldots, 15b_i + V_{i-1}, 1 \leq i \leq n.$$

In this way $|V_i| = 16^i$, and we can identify $\mathbb{Z}_{16}^n$ by V_n. Assume that we have a property P which can test if a vector $c \in R^n$ is selected or not. The selection property P on V can be seen as a boolean valued function $P : V \to \{True, False\}$ that depends on one variable. Over $\mathbb{Z}_{16}$, the property P is called a multiplicative property if $P[x]$ is true implies $P[\beta x]$ is true for all $\beta \in \mathbb{Z}_{16}^*$.

Lemma 5.7. *[23] Let R be a finite chain ring, with maximal ideal $\langle\gamma\rangle$ and nilpotency index e. Let P be a multiplicative property over R, and let $a_i \in V_i$*

be such that $P[\gamma^j a_i + c]$ *for all* $0 \leq j \leq e-1$ *and for all* $c \in \mathcal{C}_{i-1}$, *for* $i \geq 1$. *Then every* $x \in V_i \backslash V_{i-1}$ *satisfying* $P[\gamma^j a_i + c]$, *for all* $0 \leq j \leq e-1$ *an for all* $c \in \mathcal{C}_i$ *is in* $\mathcal{C}_i$.

Algorithm 1

1. $\mathcal{C}_0 := 0; i := 1$.
2. *Select the first vector* $a_i \in V_i \backslash V_{i-1}$ *such that* $P[2^j a_i + c]$ *for all* $c \in \mathcal{C}_{i-1}$, $0 \leq j \leq 3$.
3. *If such an* a_i *exists, then* $\mathcal{C}_i := \mathcal{C}_{i-1}, a_i + \mathcal{C}_{i-1}, 2a_i + \mathcal{C}_{i-1}, \ldots, 15a_i + \mathcal{C}_{i-1}$; *otherwise* $\mathcal{C}_i := \mathcal{C}_{i-1}$.
4. $i := i + 1$, *return to step 2.*

For $0 < i \leq n$, *the code* $\mathcal{C}_i$ *is forced to be linear because all linear combinations of the selected vectors* $a_{i1}, \ldots, a_{il}, l \leq i$, *are included.* $\mathcal{C}_i$ *has a generating set formed by* $a_{i1}, \ldots, a_{il}$. *Thus, we obtain a nested sequence of codes*

$$0 = \mathcal{C}_0 \subseteq \mathcal{C}_1 \subseteq \cdots \subseteq \mathcal{C}_n.$$

$\mathcal{C}_n$ *is the lexicode denoted by* $\mathcal{C}_n = \mathcal{C}(B, P)$, *where* B *is the ordering and* P *is the selection property.*

Selection Property *The following proposition will be used to minimize the number of base pairs* AT, *i.e. all codewords* c *contain more elements*

$$c_i \in \{GC, CG, GG, CC\} \bigcup \{TG, AG, GA, CT, TC, AC, CA, GT\},$$

so that the DNA codes will have large GC *content. Let* m *be an integer such that* $m < n$.

Proposition 5.8. *The property* $P_1[x]$ *is true if and only if* $w_{AT}(\phi(x)) < m$ *is a multiplicative property over* $\mathbb{Z}_{16}$.

Proof. *Let* $x \in \mathbb{Z}_{16}^n$ *such that* $w_{AT}(\phi(x)) < m$. *Multiplying the vector* x *by* β, $\beta \in \mathbb{Z}_{16}^*$, *does not change the number of occurrences of* 2, 6, 10 *and* 14 *in* x. *This gives that* $w_{AT}(\phi(\beta x)) = w_{AT}(\phi(x)) < m$, *and the result follows.* □

The following proposition will be used to ensure that all codewords c *contain a large number of* $c_i \in \{GC, CG, GG, CC\}$ *and* $w_{AT} = 0$ *to obtain DNA codes with large* GC *content.*

Table 5.3: DNA Lexicodes over $\mathbb{Z}_{16}^n$ Obtained using the Selection Property P

n	P	d_H	Basis of $\mathbb{Z}_{16}$	Basis of $\mathcal{C}(B,P)$
8	$w_{AT} < 3$	2	Canonical basis	22000011 00000022
6	$w_{GC} > 2$	1	Canonical basis	111000 000222 000041 000004
6	$w_{GC} > 3$	3	Canonical basis	111000 000222
8	$w_{GC} > 5$ and $w_{AT} = 0$	3	Canonical basis	1111201200 004038111 000040412

Proposition 5.9. *The property $P_2[x]$ is true if and only if $w_{GC}(\phi(x)) > m$ and $w_{AT}(\phi(x)) = 0$ is a multiplicative property over $\mathbb{Z}_{16}$.*

Proof. *Let $x \in \mathbb{Z}_{16}^n$ such that $w_{GC}(\phi(x)) > m$ and $w_{AT}(\phi(x)) = 0$. Multiplying the vector x by β, $\beta \in \mathbb{Z}_{16}^*$, does not change the number occurrences of 0, 2, 4, 6, 8, 10, 12 and 14 in the vector x. This gives that $w_{GC}(\phi(\beta x)) = w_{GC}(\phi(x)) > m$ and $w_{AT}(\phi(\beta x)) = w_{AT}(\phi(x)) = 0$. Hence the result follows.* □

5.1.1 DNA Codes with Deletion Similarity Distance D

The authors in [28] gave an algorithm of DNA strands with a free energy constraint. In this paper a greedy algorithm with a constraint on the deletion similarity distance D is used to obtain DNA codes satisfying the reverse-complement on the deletion similarity distance D and have high energy hybridization (the stability of the DNA duplex). Table 6.5 gives a DNA codes strands with selection property P_3.

Algorithm 2

1. $\mathcal{C}_0 := 0; i := 1$.

Table 5.4: DNA Code Strands Corresponding to the Linear Code in the First Row of Table 3.1

Codewords	Codewords
GGGGGGGGGGGGGGGG	GGGGGGGGAAAACTCT
GGGGGGGGCGCGTATA	GGGGGGGGGCGCTATA
GGGGGGGGGGGGAAAA	GGGGGGGGAAAAAGAG
GGGGGGGGCTCTATAT	GGGGGGGGTCTCTTTT
GGGGGGGGGGGGCGCG	GGGGGGGGAAAAGCGC
GGGGGGGGCTCTTTTT	GGGGGGGGTCTCTATA
CGCGGGGGGGGGGGGG	GGGGGGGGAAAATCTC
GGGGGGGGCTCTTATA	GGGGGGGGTTTTACAC
GGGGGGGGGGGGATAT	GGGGGGGGAAAAACAC
GGGGGGGGATATAGAG	GGGGGGGGTTTTCCCC
GGGGGGGGGGGGGCGC	GGGGGGGGAAAACCCC
GGGGGGGGCTCTATAT	GGGGGGGGTTTTGAGA
GCGCGGGGGGGGGGGG	GGGGGGGGAAAAGAGA
GGGGGGGGCTCTTTTT	GGGGGGGGTTTTGTGT
GGGGGGGGGGGGTTTT	GGGGGGGGAAAAGTGT
GGGGGGGGCTCTTATA	GGGGGGGGACACTATA
GGGGGGGGGGGGCCCC	GGGGGGGGCACAATAT
GGGGGGGGATATCCCC	GGGGGGGGCCCCCCCC
CCCCGGGGGGGGGGGG	GGGGGGGGCACATTTT
GGGGGGGGATATGAGA	GGGGGGGGCCCCTATA
GGGGGGGGGGGGTATA	GGGGGGGGCACATATA
GGGGGGGGATATGTGT	GGGGGGGGGAGATATA
GGGGGGGGTGTGAAAA	GGGGGGGGCGCGCGCG
GGGGGGGGAGAGTTTT	GGGGGGGGTATAGTGT
GGGGGGGGTGTGATAT	GGGGGGGGCGCGATAT
GGGGGGGGAGAGTATA	GGGGGGGGAAAACGCG
GGGGGGGGTGTGTTTT	GGGGGGGGCGCGGCGC
GGGGGGGGGCGCGCGC	GGGGGGGGCCCCCGCG
GGGGGGGGTGTGTATA	GGGGGGGGGCGCCGCG
GGGGGGGGGCGCTTTT	GGGGGGGGCCCCGCGC
GGGGGGGGAAAACACA	GGGGGGGGCGCGTTTT
GGGGGGGGGCGCCCCC	GGGGGGGGCGCGCCCC

Table 5.5: DNA Code Strands with Selection Property P_3

Codewords	WCC of the Codewordes	Codewords	WCC of the Codewordes
GGGGGGGGGGGG	CCCCCCCCCCCC	GCGCGCCGCGCG	CGCGCGGCGCGC
GGGGGGAAAAAA	CCCCCCTTTTTT	CGCGCGGCGCGC	GCGCGCCGCGCG
AAAAAAGGGGGG	TTTTTTCCCCCC	ATATATCGCGCG	TATATAGCGCGC
GGGGGGCGCGCG	CCCCCCGCGCGC	CGCGCGATATAT	GCGCGCTATATA
CGCGCGGGGGGG	GCGCGCCCCCCC	CGCGCGCGCGCG	GCGCGCGCGCGC
GGGGGGATATAT	CCCCCCTATATA	TATATAAAAAAA	ATATATTTTTTT
ATATATGGGGGG	TATATACCCCCC	AAAAAATATATA	TTTTTTATATAT
GGGGGGGCGCGC	CCCCCCCGCGCG	CCCCCCAAAAAA	GGGGGGTTTTTT
GCGCGCGGGGGG	CGCGCGGGGGGG	AAAAAACCCCCC	TTTTTTGGGGGG
GGGGGGTTTTTT	CCCCCCAAAAAA	AAAAAATTTTTT	TTTTTTAAAAAA
TTTTTTGGGGGG	AAAAAACCCCCC	GCGCGCAAAAAA	CGCGCGTTTTTT
GGGGGGCCCCCC	CCCCCCGGGGGG	AAAAAAGCGCGC	TTTTTTCGCGCG
CCCCCCGGGGGG	GGGGGGCCCCCC	ATATATAAAAAA	TATATATTTTTT
GGGGGGTATATA	CCCCCCATATAT	AAAAAAATATAT	TATATATTTTTT
TATATAGGGGGG	ATATATCCCCCC	CGCGCGAAAAAA	GCGCGCTTTTTT
AAAAAAAAAAAA	TTTTTTTTTTTT	AAAAAACGCGCG	TTTTTTGCGCGC

2. *Select the first vector* $a_i \in V_i \backslash V_{i-1}$ *such that* $S^\alpha(a_i, c) > D$ *and* $S^\alpha(a_i, c^{rc}) > D$ *for all* $c \in \mathcal{C}_{i-1}$.

3. *If such an* a_i *exists, then* $\mathcal{C}_i := \mathcal{C}_{i-1}, a_i + \mathcal{C}_{i-1}, 2a_i + \mathcal{C}_{i-1}, \ldots, 15a_i + \mathcal{C}_{i-1}$; *otherwise* $\mathcal{C}_i := \mathcal{C}_{i-1}$.

4. $i := i + 1$; *return to step 2.*

Proposition 5.10. *The property* $P_3[x]$ *is true if and only if* $S^\alpha(\phi(x), \phi(y)) > D$ *and* $S^\alpha(\phi(x), \phi(x^{rc}) > D$ *is a multiplicative property over* $\mathbb{Z}_{16}$.

Proof. *Let* x *and* y *be two vectors in* $\mathbb{Z}_{16}^n$. *Multiplying* x *and* y *by* β, $\beta \in \mathbb{Z}_{16}^*$, *does not change the number of occurrences of 0, 2, 4, 6, 8, 10, 12 and 14 in* x *and* y. *We have* $\sum_{i=1}^n n_i(x) = \sum_{i=1}^n n_i(\beta x)$ *and* $\sum_{i=1}^n n_i(y) = \sum_{i=1}^n n_i(\beta y)$, *such that* $i \in \{1, 3, 5, 7, 11, 13, 15\}$. *Thus* $S^\alpha(\phi(x), \phi(y)) = S^\alpha(\beta\phi(x), \beta\phi(y))$ *which implies that* $S^\alpha(\phi(\beta x), \phi(\beta y)) > D$ *and* $S^\alpha(\phi(\beta x), \phi(\beta x^{rc}) > D$. □

5.2 Upper and Lower Bounds

Let $A_{16}(n, D)$ *be the maximum size of a code over* $\mathbb{Z}_{16}$ *with length* n *and minimum distance* D. *Let* $A_{16}^{GC}(n, D, w)$ *be the maximum size of a code with length* n, *minimum distance* D *and GC content* w. *Further, let* $A_{16}^{r,GC}(n, D, w)$, *respectively,* $A_{16}^{rc,GC}(n, D, w)$ *be the maximum size of a DNA code with length* n, *minimum distance* D *and GC content* w. *The maximum size* $A_{16}^{GC}(n, D, w)$ *of the code over* $\mathbb{Z}_{16}$ *is greater than the maximum size* $A_4^{GC}(n, D, w)$ *of the code over* $\mathbb{Z}_4$ *given in [6] and [24] because there are* 16^n *codewords over* $\mathbb{Z}_{16}$

but only 4^n codewords over $\mathbb{Z}_4$. The following proposition is analogous to the result for DNA codes with unrestricted GC content and deletion similarity distance given in [28].

Proposition 5.11.

(i)

$$A_{16}^{GC,r}(n-1,D,w) \leq A_{16}^{GC,r}(n,D,w). \tag{5.4}$$

(ii)

$$A_{16}^{GC,r}(n-1,D,w) \geq A_{16}^{GC,r}(n,D,w)/16. \tag{5.5}$$

for odd n.

Proof. *For part (i), we have 16^n codewords of length n and 16^{n-1} codewords of length $n-1$ over $\mathbb{Z}_{16}$, and the result follows. For part (ii), the codewords of $\mathcal{C}(n, A_4^{GC,r}(n,D,w), D)$ over $\mathbb{Z}_{16}$ can be partitioned into 16 subsets denoted $C_1, C_2, \ldots, C_{16}$ such that the size of at least one subset C_i is $A_{16}^{GC,r}(n,D,w)/16$ and so an $(n, A_{16}^{GC,r}(n,D,w)/16, d)$ code exists. Removing a symbol from the codewords of this subset such that the distance D and weight w are maintained, we obtain an $(n-1, A_{16}^{GC,r}(n,D,w), D)$ code, and the result follows.* □

The following proposition is analogous to that for DNA codes with unrestricted GC content and deletion similarity distance given in [24].

Proposition 5.12. *For $0 \leq D \leq n$ and $0 \leq w \leq n$*

$$A_{16}^{GC,rc}(n,D,w) = A_{16}^{GC,r}(n,D,w), \text{ if } n \text{ is even,}$$

and

$$A_{16}^{GC,r}(n,D+1,w) \leq A_{16}^{GC,rc}(n,D,w) \leq A_{16}^{GC,r}(n,D-1,w), \text{ if } n \text{ is odd.}$$

Proof. *The proof is similar to that for Proposition 12 in [24].* □

Proposition 5.13.

$$A_{16}^{GC}(n,D,w) \geq A_{16}^{GC}(n+1,D+1,w). \tag{5.6}$$

$$A_{16}^{GC}(n,D,w) \geq A_{16}^{GC}(n+1,D,w)/16. \tag{5.7}$$

Proof. *The analogous result for DNA codes with unrestricted GC content and Hamming distance was given in [28]. The corresponding proof is given here for the deletion similarity distance. For (5.6), an* $(n, A_{16}^{GC}(n+1, D+1, w), D, w)$ *code can be obtained from an* $(n+1, A_{16}^{GC}(n+1, D+1, w), D+1, w)$ *code by removing a symbol from each codeword. For (5.7), if the codewords in* $(n+1, A_{16}^{GC}(n+1, D, w), D, w)$ *are partitioned into 16 subsets according to the first symbol, one of the subsets will have size at least* $A_{16}^{GC}(n+1, D, w)/16$*, and thus an* $(n+1, A_{16}^{GC}(n+1, D, w)/16, D, w)$ *code exists.* □

Proposition 5.14.

$$A_{16}^{rc,GC}(n, D, w) \leq A_{16}^{GC}(n, D, w)/2.$$

Proof. *Assume that* $M = \{x_i, 0 \leq i \leq |M|\}$ *is a set of* $|M|$ *codewords of length* n *with GC content* w *and minimum distance* D *such that* $S^{\alpha}(s_i, s_j^{rc}) \geq D$ *for all codewords in* M*, and* $x_j^{rc} \in M^{rc} = \{x_j^{rc}, x_j \in M, 0 \leq i \leq |M|\}$*. Then* $M \bigcup M^{rc}$ *is a set of* $2|M|$ *codewords of length* n *with GC content* w *and minimum distance at least* D *provided* $M \bigcap M^{rc} = \emptyset$ *and* $|M \bigcup M^{rc}| = 2|M|$*. Then* $A_{16}^{rc,GC}(n, D, w) = |M| \leq A_{16}^{GC}(n, D, w)/2$. □

Theorem 5.15. *For* $0 \leq D \leq n$ *and* $0 \leq w \leq n$

$$A_{16}^{GC}(n, D, w) \leq A_4^{GC}(n, D, w) A_4^{GC}(n, D, w), \tag{5.8}$$

$$A_{16}^{GC,rc}(n, D, w) \leq A_4^{GC,rc}(n, D, w) A_4^{GC,rc}(n, D, w). \tag{5.9}$$

Proof. *For (5.8), we define a mapping* $\bigodot$ *from DNA codeword over* $\phi(\mathbb{Z}_4)$ *to DNA codeword of the base pair over* $\phi(\mathbb{Z}_{16})$*, denote* M_1 M_2 *as sets of quaternary DNA codewords with length* n*, distance* D *and GC content* w*. Then* $M_1 \bigodot M_2 = \{x \bigodot y, x \in M_1, y \in M_2\}$ *is a set of* $|M_1||M_2|$ *DNA codewords with length* n*, distance* D *and GC content* w*. We have* $S^{\alpha}(x_1, x_2^r) \geq D$ *for all* $x_1, x_2 \in S_1$*, so then* $S^{\alpha}(z_1, z_2^r) \geq D$ *for all* $z_1, z_2 \in S_1 \bigodot S_2$*. Therefore* $S^{\alpha}(x_1 \bigodot y_1, (x_2 \bigodot y_2)^R) = S^{\alpha}(x_1 \bigodot y_1, x_2^r \bigodot y_2^r) = S^{\alpha}(z_1, z_2^R) \geq S^{\alpha}(x_1, x_2^r) \geq D$*. Hence the result. The proof of (5.9) is similar to that of (5.8).* □

Let S *be a set of* V *DNA codewords over* $\phi(\mathbb{Z}_{16})$ *with length* n*, GC content* w *and distance* D*.* S *is defined by* $V(n, w, D) = \sharp\{x \in \mathbb{Z}_{16}^n$*, has GC-content,* w *and* $S^{\alpha}(x, x^{rc}) = D\} = |S|$*. Denote the number of codewords in* S *having distance at most* D *from words in* S *by* $V(s, D)$*. This*

is independent of the choice of $s \in S$, so it can be denoted by $V(D)$ where $V(D) = \sum_{i=0}^{d} \binom{n}{i} (15^i)$. The following sphere-packing upper bound with GC content w and Gilbert-Varshamov lower bound with GC content w are similar to the corresponding results in [28], [19] and [26], and the proofs are similar to those in [19] and [26].

Theorem 5.16. *(Sphere-packing upper bound with GC-content w)*

$$A_{16}^{GC}(n, D, w) \leq \frac{|S|}{V(\lfloor (D-1)/2 \rfloor)}. \tag{5.10}$$

Theorem 5.17. *(Gilbert-Varshamov lower bound with GC-content w)*

$$A_{16}^{GC}(n, D, w) \geq \frac{|S|}{V(D-1)}. \tag{5.11}$$

The following Theorem is analogous to the result for DNA codes with unrestricted GC content w and deletion similarity distance D given in [28].

Theorem 5.18. *For $0 \leq D \leq n$ and $0 \leq w \leq n$*

(i)

$$A_{16}^{GC,rc}(n, D, w) \geq \frac{|S|}{2V^{+}(D-1)}, \tag{5.12}$$

where $V^{+} = \max\{V(s, D) | s \in S\}$.

(ii)

$$A_{16}^{GC,rc}(n, D, w) \leq \frac{|S|}{2V^{-}(\lfloor (D-1)/2 \rfloor)}, \tag{5.13}$$

where $V^{-} = \min\{V(s, D) | s \in S\}$.

Proof. *For part (i), Theorem 5.16 and the construction of DNA codes using the greedy algorithm over $\mathbb{Z}_{16}$ provides a reverse-complement code of size $\frac{|S|}{2V^{+}(D-1)}$. For part (ii), Theorem 5.17 and the construction of DNA code using the greedy algorithm over $\mathbb{Z}_{16}$ provides a reverse-complement code of size $\frac{|S|}{2V^{-}(\lfloor (D-1)/2 \rfloor)}$.* □

Chapter 6

New DNA Cyclic Codes over Rings $R = \mathbb{F}_2[u]/(u^6)$

In this chapter is dealing with DNA cyclic codes which play an important role in DNA computing and have attracted a particular attention in the literature. Firstly, we introduce a new family of DNA cyclic codes over the ring $R = \mathbb{F}_2[u]/(u^6)$. *Such codes have theoretical advantages as well as several applications in DNA computing. A direct link between the elements of such a ring and the* 64 *codons used in the amino acids of the living organisms is established. Such a correspondence allows us to extend the notion of the edit distance to the ring* R *which is useful for the correction of the insertion, deletion and substitution errors. Next, we define the Lee weight, the Gray map over the ring* R *as well as the binary image of the cyclic DNA codes allowing the transfer of studying DNA codes into studying binary codes. Secondly, we introduce another new family of DNA skew cyclic codes constructed over the ring* $\tilde{R} = \mathbb{F}_2 + v\mathbb{F}_2 = \{0, 1, v, v+1\}$ *where* $v^2 = v$ *and study their property of being reverse-complement. We show that the obtained code is derived from the cyclic reverse-complement code over the ring* $\tilde{R}$. *We shall provide the binary images and present some explicit examples of such codes.*

6.1 DNA cyclic codes over $R = \mathbb{F}_2[u]/(u^6)$

The ring considered in this chapter is

$$R = \mathbb{F}_2[u]/(u^6) = \{a_0 + a_1u + a_2u^2 + a_3u^3 + a_4u^4 + a_5u^5; a_i \in \mathbb{F}_2, u^6 = 0\}.$$

It is a commutative ring with 64 elements. It is a principal local ideal ring with maximal ideal $\langle u \rangle$*. The ideals of* R *satisfy the following inclusions*

$$\langle 0 \rangle = \langle u^6 \rangle \subsetneq \langle u^5 \rangle \subsetneq \langle u^4 \rangle \subsetneq \langle u^3 \rangle \subsetneq \langle u^2 \rangle \subsetneq \langle u \rangle \subsetneq \langle R \rangle.$$

Since the ring R *is of the cardinality* 64*, then we can construct a one-to-one correspondence between the elements of* R *and the* 64 *codons over the alphabet* $\{A, G, C, T\}^3$ *by the map* ϕ*, this is given in Table 6.1. A simple verification shows that for all* $x \in R$*, we have*

$$x + \hat{x} = u^5 + u^4 + u^3 + u^2 + u + 1. \tag{6.1}$$

Now, since R^n *is an* R*-module, a linear code over* R *of length* n *is a sub-*

Table 6.1: Identifying codons with the elements of the ring R.

CCC	$u^5+u^4+u^3+u^2+u+1$	GGG	0	ACT	u^3+u^2+1	GTC	u^4+u^2+u+1
GGA	$u^5+u^4+u^3+u^2+u$	CCT	1	ACG	u^3+u^2+u	ACA	u^3+u^2+u+1
GGC	$u^5+u^4+u^3+u^2+1$	CCG	u	TTT	u^4+u^2+1	GAC	$u^5+u^3+u^2+1$
GGT	$u^5+u^4+u^3+u^2$	CCA	$u+1$	TTG	u^4+u^2+u	AGG	u^5+u^3+u+1
AGG	$u^5+u^4+u^3+u+1$	TCC	u^2	CTA	u^4+u+1	GAT	$u^5+u^3+u^2$
CGG	$u^5+u^4+u^2+u+1$	GCC	u^3	GTT	u^4+u^3+1	GTA	u^4+u^3+u+1
GAG	$u^5+u^3+u^2+u+1$	CTC	u^4	GTG	u^4+u^3+u	ATT	$u^4+u^3+u^2+1$
AGA	$u^5+u^4+u^3+u$	TCT	u^2+1	TCA	u^2+u+1	ATA	$u^4+u^3+u^2+u$
AGC	$u^5+u^4+u^3+1$	TCG	u^2+u	CAA	u^5+u^2+u	ATC	$u^4+u^3+u^2$
ATG	$u^4+u^3+u^2+u+1$	TAC	u^5	CAC	u^5+u^2+u	TGA	u^5+u^4+u
AGT	$u^5+u^4+u^3$	TAT	u^5+1	GCA	u^3+u+1	AAT	u^5+u^2+u+1
CGA	$u^5+u^4+u^2+u$	GCT	u^3+1	TTA	u^4+u^3	AAA	u^5+u^3+u
CGC	$u^5+u^4+u^2+1$	GCG	u^3+u	ACC	u^3+u^2	TGC	u^5+u^4+1
CGT	$u^5+u^4+u^2$	TAA	u^5+u	CAT	u^5+u^2	AAC	u^5+u^3+1
TGG	u^5+u^4+u+1	CTG	u^4+u	TGT	u^5+u^4	TCC	u^4+u^2
GAA	$u^5+u^3+u^2+u$	CTT	u^4+1	CAG	u^5+u^3	TAG	u^5+u+1

module $\mathcal{C}$ *of* R^n*. An* (n, k) *linear block code of dimensions* $n = ml$*, is called* quasi-cyclic *if every cyclic shift of a codeword by* l *symbol yields another codeword. For* $x \in R^n$*, denote the number of the component of* x *equal to* a_i *by* $n_{a_i}(x)$*. The Hamming weight of* x *is* $w_H(x) = \sum_{i=0}^{n-1} n_{a_i}(x)$*, where* $a_i \in R^*$*. The Hamming distance* $d_H(x, y)$ *between the vector* x *and* y *equals* $w_H(x-y)$*. Let* $x = x_0x_1 \dots x_{n-1}$ *be a vector in* R^n*. The reverse of* x *is defined as* $x^r = x_{n-1}x_{n-2} \dots x_1x_0$*, the complement of* x *is* $x^c = \hat{x}_0\hat{x}_1 \dots \hat{x}_{n-1}$*, and also called the* Watson-Crick complement *(WCC), the reverse-complement is defined as* $x^{rc} = \hat{x}_{n-1}\hat{x}_{n-2} \dots \hat{x}_1\hat{x}_0$*. A code* $\mathcal{C}$ *is said to be* reversible *if for any* $x \in \mathcal{C}$*, we have* $x^r \in \mathcal{C}$*. Moreover,* $\mathcal{C}$ *is said to be* reverse-complement *if for any* $x \in \mathcal{C}$*, we have* $x^{rc} \in \mathcal{C}$*.*

It is easy to check the following bounds on the edit distance d_c*.*

Proposition 6.1. *Assume that X and Y are two strings in R^n. Then the following holds:*

(i) $d_c(\phi(X), \phi(Y)) \leq n$;

(ii) $d_c(\phi(X), \phi(Y)) \leq d_H(\phi(X), \phi(Y))$;

(iii) $d_c(\phi(X), \phi(\hat{Y})) = d_c(\phi(Y), \phi(\hat{X}))$.

6.1.1 Cyclic Codes over $R = \mathbb{F}_2[u]/(u^6)$

In this subsection we give the algebraic structure of the cyclic code of arbitrary length over R. We start by giving the definition of cyclic code over this ring. Let $\mathcal{C}$ be a code over R of length n. A codeword $(c_0, c_1, \cdots, c_{n-1})$ of $\mathcal{C}$ is viewed as a polynomial $c_0 + c_1 x + \cdots + c_{n-1}x^{n-1}$ in $R[x]$. Let τ be the cyclic shift acting on the codewords of $\mathcal{C}$ in the following way:

$$\tau(c_0, c_1, \cdots, c_{n-1}) = (c_{n-1}, c_0, c_1, \cdot, \cdot, \cdot, c_{n-2}).$$

Recall that linear code $\mathcal{C}$ is cyclic if $\mathcal{C}$ is invariant under permutation τ : $c(x) \mapsto xc(x) \pmod{x^n - 1}$.

The following theorem is a particular result of [12, 21] which gives the structure of the cyclic codes of arbitrary lengths.

Theorem 6.2. *Let $\mathcal{C}$ be a cyclic code of arbitrary length n over the ring R. (i) Assume n is odd. Then there exist polynomials $f_0, f_1, f_2, f_3, f_4, f_5$ over R, such that $f_5|f_4|f_3|f_2|f_1|f_0|x^n - 1$ and $\mathcal{C} = \langle f_0, uf_1, u^2f_2, u^3f_3, u^4f_4, u^5f_5\rangle$; (ii) Assume $n = m2^s$ such that $gcd(m, 2) = 1$. Then the cyclic codes of the length n over R are the ideal generated by $\mathcal{C} = \langle f_0, uf_1, u^2f_2, u^3f_3, u^4f_4, u^5f_5\rangle$, where $f_i|f_0$ and f_0 is divisor of $x^n - 1$ in $\mathbb{F}_2$.*

Let denote by K the field $R/(u)$. We have the following canonical ring morphism

$$- : R[x] \to K[x];\ f \mapsto \overline{f} = f \pmod{u}.$$

The rank of $\mathcal{C}$ is defined as

$$k(\mathcal{C}) = \sum_{i=0}^{5} k_i,$$

where the k_i are such that $|C| = |K|^{\sum_{i=0}^{5}(5-i)k_i}$. The submodule quotient of $\mathcal{C}$ by $v \in R$ is the code

$$(\mathcal{C} : u^i) = \{v \in R^n | u^i v \in \mathcal{C}\}.$$

Thus we have the following tower of linear codes over R.

$$\mathcal{C} = (\mathcal{C} : u) \subseteq (\mathcal{C} : u^2) \subseteq (\mathcal{C} : u^3) \subseteq (\mathcal{C} : u^4) \subseteq (\mathcal{C} : u^5). \tag{6.2}$$

For $i = 0, \cdots, 5$ the projections of $(\mathcal{C} : u^i)$ over the field K are denoted by $Tor_i(\mathcal{C}) = (\overline{\mathcal{C} : u^i})$ and are called the torsion codes associated to the code $\mathcal{C}$ (see [30]).
The following theorem presents some bounds on the edit distance for the cyclic codes defined above.

Theorem 6.3. *Let $\mathcal{C} = \langle f_0, uf_1, u^2f_2, u^3f_3, u^4f_4, u^5f_5 \rangle$ be a cyclic code over R of odd length. Then the minimum edit distance d_c of $\mathcal{C}$ satisfies the following inequalities.*
(i) $d_c(\mathcal{C}) = min\{d_c(Tor_i(\mathcal{C}))\} \leq min\{d_H(Tor_i(\mathcal{C}))\}$, where $i = 0, \cdots, 5$;
(ii) $d_c(\mathcal{C}) \leq min\{deg(f_i)\} + 1$, where $i = 0, \cdots, 5$;
(iii) $d_c(\mathcal{C}) \leq n - rank(\mathcal{C}) + 1$.

Proof. *From [13, Lemma 5.1] and Proposition 6.1, we have $d_c(\mathcal{C}) = min\{d_c(Tor_i(\mathcal{C}))\} \leq min\{d_H(Tor_i(\mathcal{C}))\}$ for every $i \in \{0, \cdots, 5\}$. Assertion (ii) comes from the fact that the code $Tor_i(\mathcal{C})$ and $Tor_0(\mathcal{C})$ are binary cyclic codes satisfying $\langle f_i \rangle \subset \mathcal{C}$. The dimension of $Tor_i(\mathcal{C})$ is $n - deg(f_i)$. By the well-known Singleton bound, we have $d_c(\mathcal{C}) \leq min\{deg(f_i)\} + 1$. Assertion (iii) follows from Proposition 6.1 using again the Singleton bound.* □

6.1.2 DNA Cyclic Codes

Now, we introduce a DNA cyclic code by constructing more precisely, a $[3n, d]$-DNA cyclic code. Set $\mathbb{I}(x) := (x^n - 1)/(x - 1)$ and $\alpha(u) := u^5 + u^4 + u^3 + u^2 + u + 1$.

Definition 6.4. *Let $1 \leq D \leq 3n-1$ be a positive real number. Then a cyclic code $\mathcal{C}$ of length n over R is called an $[n, D]$ DNA cyclic code if the following conditions hold:*

(i) $\mathcal{C}$ is cyclic code, i.e.; $\mathcal{C}$ is an ideal in $R_n = R[x]/(x^n - 1)$;

(ii) for any codeword $x \in C$, we have $(x)^{rc} \neq (x)$ and $(x)^{rc} \in \mathcal{C}$;

(iii) $d_c(x, y) \leq D$ for any $x, y \in C$.

Condition (ii) given in Definition 6.4 shows that the defined DNA cyclic codes are reverse-complement cyclic codes.

Definition 6.5. *Let $f(x) \in R[x]$, denote $f(x)^* = x^{deg(f)} f(\frac{1}{x})$ the reciprocal polynomial of $f(x)$. The polynomial f is said to be* self-reciprocal *if $f(x) = f^*(x)$.*

The following statement can be obtained straightforwardly.

Lemma 6.6. *Let $f(x)$ and $g(x)$ be a polynomials in $R[x]$ with $deg(f(x)) \geq deg(g(x))$. Then the following conditions hold:*

(i) $(f(x)g(x))^* = f(x)^* g(x)^*$;

(ii) $(f(x) + g(x))^* = f(x)^* + x^{deg(f)-deg(g)} g(x)^*$.

Theorem 6.7. *Let $\mathcal{C}$ be a cyclic code of odd length n over R and assume that $\mathcal{C}$ is reverse-complement. Then we have:*

(i) $\mathcal{C}$ contains all the codewords of the form $\alpha(u)\mathbb{I}(x)$;

(ii) $\mathcal{C} = \langle f_0, uf_1, u^2 f_2, u^3 f_3, u^4 f_4, u^5 f_5 \rangle$ where f_i is self-reciprocal for every $i \in \{0, \cdots, 5\}$.

Proof.

(i) Since $\mathcal{C}$ is linear, $(0, \cdots, 0) \in \mathcal{C}$. Also $\mathcal{C}$ is revers-complement, so that $(0, \cdots, 0)^{rc} \in \mathcal{C}$. Then we have $(0, \cdots, 0)^{rc} = (\alpha(u), \cdots, \alpha(u)) = \alpha(u)(x^n - 1)/(x - 1) \in \mathcal{C}$.

(ii) Let us show that $f_i^*(x) = f_i(x)$ for $i \in \{0, \cdots, 5\}$.

Set $f_0(x) = a_0 + a_1 x + \cdots + a_{m-1} x^{m-1} + a_m x^m$ where $f_0/(x^n - 1)$ in $\mathbb{F}_2[x]$. One can assume that $a_0 = a_m = 1$. So that $f_0(x) = 1 + a_1 + \cdots + a_{m-1} x^{m-1} + x^m$. Suppose that $f_0(x)$ corresponds to the vector $(1, a_1, \cdots, 1, 0, 0 \cdots, 0)$ and the reverse-complement of 0 in R is $\alpha(u)$ then

$$f_0^{rc}(x) = \alpha(u)(1+x+\cdots+x^{n-m-2})+(\alpha(u)+1)x^{n-m-1}+\hat{a}_{m-1}x^{n-m}+\cdots+\hat{a}_1 x^{n-2}+(\alpha(u)+1)x^{n-1}) \in \mathcal{C}.$$

Now, since $\mathcal{C}$ is a linear code, we get

$$f_0(x)^{rc} + \alpha(u)(\frac{x^n-1}{x-1}) \in \mathcal{C}.$$

This implies that

$$x^{n-m-1} + (\hat{a}_{m-1} + \alpha(u))x^{n-m} + \cdots + (\hat{a}_1 + \alpha(u))x^{n-2} + x^{n-1}$$

$$= x^{n-m-1}[1 + (\hat{a}_{r-1} + \alpha(u))x + \cdots + (\hat{a}_1 + \alpha(u))x^{m-1} + x^m] \in \mathcal{C}.$$

Multiplying the last polynomial by x^{m+1} in $R[x]/(x^n-1)$, we obtain:

$$1 + (\hat{a}_{m-1} + \alpha(u)) + \cdots + (\hat{a}_1 + \alpha(u))x^{m-1} + x^m \in \mathcal{C}.$$

By Equation (6.5) we see that $\hat{a} + \alpha(u) = a$. Therefore, we obtain:

$$f_0^*(x) = 1 + a_{m-1}x + \cdots + a_1x^{m-1} + x^m \in C.$$

Consequently, we have

$$f_0^*(x) = f_0k_0 + uf_1k_1 + \cdots + u^5f_5k_5,$$

where f_i and k_i are all in $\mathbb{F}_2[x]$. Multiplying both sides of this equality by u^5 gives

$$u^5f_0^*(x) = u^5k_0(x)f_0(x).$$

Now, since $f_0^*(x), f_0(x) \in \mathbb{F}_2[x]$ have the same degree, leading coefficient and constant term, one necessary have $k_0(x) = 1$. Consequently, $f_0(x)$ is self-reciprocal. The same argument can be used for f_1, f_2, f_3, f_4 and f_5 as well. □

In the following, we are interested in providing sufficient conditions for a given code $\mathcal{C}$ to be reverse-complement.

Theorem 6.8. *Assume that $\mathcal{C} = \langle f_0, uf_1, u^2f_2, u^3f_3, u^4f_4, u^5f_5\rangle$ is a cyclic code of odd length n over R with $f_5|f_4|f_3|f_2|f_1|f_0|x^n-1 \in \mathbb{F}_2[x]$. If $\alpha(u)\mathbb{I}(x) \in C$ and $f_i(x)$ are self-reciprocal, then $\mathcal{C}$ is a reverse-complement code.*

Proof. *Let $c(x)$ be a codeword in $\mathcal{C}$, we have to prove that $c(x)^{rc} \in \mathcal{C}$. Since $\mathcal{C} = \langle f_0, uf_1, u^2f_2, u^3f_3, u^4f_4, u^5f_5\rangle$ there exist $\alpha_i(x) \in R[x]$ ($i \in \{1, \cdots, 5\}$) such that*

$$c(x) = f_0\alpha_0 + uf_1\alpha_1 + u^2f_2\alpha_2 + u^3f_3\alpha_3 + u^4f_4\alpha_4 + u^5f_5\alpha_5.$$

Applying the reciprocal and using first Lemma 6.6, and next the fact that $f_0(x), f_1(x), f_2(x),$
$f_3(x), f_4(x)$ *and* $f_5(x)$ *are self-reciprocal, we obtain*

$$c^*(x) = (f_0\alpha_0)^* + (uf_1\alpha_1)^* x^{m_1} + (u^2 f_2\alpha_2)^* x^{m_2} + (u^3 f_3\alpha_3)^* x^{m_3} + (u^4 f_4\alpha_4)^* x^{m_4} + (u^5 f_5\alpha_5)^* x^{m_5},$$

proving that $c(x)^*$ *is in* $\mathcal{C}$. *Since* $\mathcal{C}$ *is cyclic,*

$$x^{n-t-1}c(x) = c_0 x^{n-t-1} + c_1 xn - t + \cdots + c_t x^{n-1} \in \mathcal{C}.$$

It was also assumed that

$$\alpha(u) + \alpha(u)x + \cdots + \alpha(u)x^{n-1} \in \mathcal{C},$$

which leads to

$$\alpha(u) + \alpha(u)x + \cdots + \alpha(u)x^{n-1} + c_0 x^{n-t-1} + c_1 xn - t + \cdots + c_t x^{n-1} \in \mathcal{C}.$$

This is equal to

$$\alpha(u) + \alpha(u)x + \cdots + \alpha(u)x^{n-t-2} + (\alpha(u) + c_0)x^{n-t-1} + \cdots + (\alpha(u) + c_t)x^{n-1}$$

$$= \alpha(u) + \alpha(u)x \cdots + \alpha(u)x^{n-t-1} + \cdots + \hat{c}_0 x^{n-t-1} + \cdots + \hat{c}_t x^{n-1},$$

which is precisely $(c^*(x)^{rc})^* = c(x)^{rc} \in \mathcal{C}$. □

Using similar arguments as in Theorem 6.8, one can prove the following statement.

Theorem 6.9. *Assume that* $\mathcal{C} = \langle f_0, uf_1, u^2 f_2, u^3 f_3, u^4 f_4, u^5 f_5 \rangle$ *is a cyclic code of even length* $n = m2^s$ *over* R *such that* $f_i | f_0$ *in* $\mathbb{F}_2[x]$. *If* $\alpha(u)\mathbb{I}(x) \in C$ *and* $f_i(x)$ *(*$i \in \{1, \cdots, 5\}$*) are self-reciprocal. Then* $\mathcal{C}$ *is a reverse-complement code.*

Corollary 6.10. *Let* C *be a cyclic code of length* $n = m2^s$, $s \geq 0$. *If* $\alpha(u)\mathbb{I}(x) \in \mathcal{C}$ *and there exists an integer* i *such that*

$$2^i \equiv -1 \pmod{m}. \tag{6.3}$$

Then $\mathcal{C}$ *is a reverse-complement code.*

Proof. *The proof is similar to the proof of Corollary 4.13 in [20].* □

Definition 6.11. *For a cyclic code $\mathcal{C} = \langle f_0, uf_1, u^2f_2, u^3f_3, u^4f_4, u^5f_5\rangle$, we define the sub-code $\mathcal{C}_{u^2}$ consisting of all codewords in $\mathcal{C}$ that are a multiple of u^2.*

Lemma 6.12. *Let $\mathcal{C}$ be a cyclic code $\mathcal{C} = \langle f_0, uf_1, u^2f_2, u^3f_3, u^4f_4, u^5f_5\rangle$ of odd length, then we have:*

(i) $\phi(u^2R) = \{GGG, AGT, CGT, TGT, GAT, CAG, TAC, ATC, GTC, TCC, CTC, ACC, CTC, TCC, GGT, CAT\}$,
$\phi(u^3R) = \{GGG, TGT, CAG, TAC, CTC, GCC, AGT, TTA\}$ *and*
$\phi(u^4R) = \{GGG, TAC, CTC, TGT\}$;

(ii) If $\mathcal{C} = \langle f_0, uf_1, u^2f_2, u^3f_3, u^4f_4, u^5f_5\rangle$ is the cyclic code of odd length n over R. Then $\mathcal{C}_{u^2} = \langle u^2f_5\rangle$ and $\phi(\mathcal{C}_{u^2})$ is over the alphabet $\phi(u^2R)$.

Proof. *The part (i) is obtained by a simple calculation.*
For the part (ii), assume that $\mathcal{C} = \langle f_0, uf_1, u^2f_2, u^3f_3, u^4f_4, u^5f_5\rangle$. Since $f_5|f_4|f_3|f_2|f_1|f_0|x^n - 1$, then we obtain $\langle u^2f_5\rangle \subset \mathcal{C}_{u^2}$. Conversely, assume that $c(x) \in C$ such that

$$c(x) = \alpha_0(x)f_0(x) + u\alpha_1(x)f_1(x) + u^2\alpha_2(x)f_2(x) + u^3\alpha_3(x)f_3(x) + u^4f_4 + u^5f_5$$

for all $\alpha_i \in \mathbb{F}_2[x]$. If $c(x)$ is a multiple of u^2 then $x^n - 1$ divides $\alpha_0(x)f_0(x)$ and $x^n - 1$ divides $\alpha_1(x)f_1(x)$. Hence,

$$c(x) = u^2\alpha_2(x)f_2(x) + u^3\alpha_3(x)f_3(x) + u^4\alpha_4(x)f_4(x) + u^5\alpha_5(x)f_5(x).$$

Therefore, $\mathcal{C}_{u^2} \subset \langle u^2f_5\rangle$. Consequently $\mathcal{C}_{u^2} = \langle u^2f_5\rangle$, which completes the proof. □

Remark 6.13. The DNA cyclic codes which are obtained in the Lemma 6.12 are stable across the error in the DNA strands by the usage of the codons, see [5].
Any codeword of sub-code $\phi(\mathcal{C}_{u^2})$ over $\phi(u^2R)$ contains the nucleotide C and G. This is an interesting thermodynamic property of the DNA strand. For its importance, we send the reader to [20].

Example 6.14. Let us consider the following polynomial in $\mathbb{F}_2[x]$,

$$x^7 - 1 = (x - 1)(x^3 + x + 1)(x^3 + x^2 + 1) = f_0f_1f_2. \tag{6.4}$$

In Table 5.2, we present the associate DNA cyclic codes of length 7 given with their corresponding size and their minimal Hamming distance. Table 5.3 and Table 5.4 present all codewords of DNA cyclic code associate to $\mathcal{C} = \langle u^4f_0f_1\rangle$ and to $\mathcal{C} = \langle f_1f_2\rangle$, respectively.

Table 6.2: DNA cyclic codes of length 7

Code $\mathcal{C}$	Size of $\mathcal{C}$	d_H
$\langle u^2 f_0 \rangle$	4096	2
$\langle u^2 f_1 \rangle$	256	3
$\langle u^2 f_2 \rangle$	256	3
$\langle u^2 f_1 f_2 \rangle$	4	7
$\langle u^2 f_0 f_1 \rangle$	64	4
$\langle u^2 f_0 f_2 \rangle$	64	4
$\langle u^4 f_0 f_1 \rangle$	64	4

Example 6.15. We have that $x^{17} - 1 = (x-1)(x^8 + x^5 + x^4 + x^3 + 1)(x^8 + x^7 + x^6 + x^4 + x^2 + x + 1) = f_0 f_1 f_2 f_3$ in $\mathbb{F}_2[x]$. In Table 5.5 we present the DNA cyclic codes associated to $\mathcal{C} = \langle f_0, uf_1, u^2 f_2, u^3 f_3, u^4 f_4, u^5 f_5 \rangle$.

6.1.3 Binary Image of DNA Codes

In this Section we will define a Gray map which allows us to translate the properties of the suitable DNA codes for DNA computing to the binary cases. Table 5.7 gives a binary image of the DNA cyclic code of length 7 given by Table 5.2. Any element $c \in R$ can be expressed as $c = a_0 + a_1 u + a_2 u^2 + a_3 u^3 + a_4 u^4 + a_5 u^5$, where $a_i \in \mathbb{F}_2$, $0 \leq i \leq 5$. The Gray map φ from R to $\mathbb{F}_2$ is defined as follows;

$$\varphi : R^n \to \mathbb{F}_2^{6n}$$

$$\varphi(a_0 + a_1 u + a_2 u^2 + a_3 u^3 + a_4 u^4 + a_5 u^5) = (a_0, a_1, a_2, a_3, a_4, a_5),$$

where $a_i \in \mathbb{F}_2$, $0 \leq i \leq 5$. We have for example $\varphi(1+u) = (1,1,0,0,0,0)$. We define the Lee weight over the ring R by

$$w_{Lee}(a_0 + a_1 u^1 + a_2 u^2 + a_3 u^3 + a_4 u^4 + a_5 u^5) = \sum_{i=0}^{i=5} a_i.$$

The Lee distance $d_L(x,y)$ between the vector x and y is $w_{Lee}(x-y)$. According to the definition of the Gray map, it is easy to check that the image of a linear code over R by φ is a binary linear code. We can obtain the binary image of the DNA code by the map φ and the map ϕ. In Table 6 we give the binary

image of the codons. The binary image of DNA code resolved the problem of constructing DNA codes with some properties, see [29].

The following property of the binary image of the DNA codes comes from the definition.

Lemma 6.16. *The Gray map φ is a linear weight preserving*

$$(R^n, \text{ Lee distance}) \to (\mathbb{F}_2{}^{6n}, \text{ Hamming distance}).$$

Further, if $\mathcal{C}$ is a DNA cyclic code of length n over R, then $\varphi(\mathcal{C})$ is a binary DNA quasi-cyclic code of length $6n$ over $\mathbb{F}_2$ and of index 6.

Proof. *Let $\mathcal{C}$ be a DNA cyclic code of length n over R. Hence $\varphi(\mathcal{C})$ is a set of length $6n$ over the alphabet $\mathbb{F}_2$ which is a quasi-cyclic code of index 6. It is easy to verify that the Gray map is a linear weight preserving.* □

Remark 6.17. The usual Gray map from the ring $\mathcal{R} = \{0, 1, u, u+1\}$ to $\mathbb{F}_2$, have the same isometric properties.

Remark 6.18. The codes of rows 2 and 3 given by Table 5.7 are optimal according to [39].

6.2 DNA Skew Cyclic Codes over $\tilde{R} = \mathbb{F}_2 + v\mathbb{F}_2$

The ring considered in this section is the non-commutative ring $\tilde{R}[x;\theta]$ where θ is an automorphism of $\tilde{R}$. The structure of the latter ring depends on the element of the commutative ring $\tilde{R} = \mathbb{F}_2 + v\mathbb{F}_2 = \{0, 1, v, v+1\}$, where $v^2 = v$ and the automorphism θ on $\tilde{R}$, defined by $\theta(0) = 0$, $\theta(1) = 1$, $\theta(v) = v + 1$, $\theta(v + 1) = v$. Note that $\theta^2(a) = \theta(\theta(a)) = a$ for all $a \in \tilde{R}$. This implies that θ is a ring automorphism of order 2. The skew polynomial ring $\tilde{R}[x;\theta]$ is the set of polynomials $\tilde{R}[x;\theta] = \{a_0 + a_1x + a_2x^2 + \cdots + a_nx^n \mid a_i \in \tilde{R}\}$ endowed with the usual addition of polynomials and the multiplication $*$ (which is not commutative) is defined by the basic rule $(ax^i) * (bx^j) = a\theta^i(b)x^{i+j}$ (the distributive and the associative laws occur).
There is a one-to-one map ψ between the elements of $\tilde{R}$ and the DNA nucleotide base $\{A, T, C, G\}$ given by $0 \mapsto G$, $v \mapsto C$, $v + 1 \mapsto T$ and $1 \mapsto A$. A simple verification shows that for all $x \in \tilde{R}$, we have

$$\theta(x) + \theta(\hat{x}) = v + 1. \tag{6.5}$$

In the following, we only consider codes with even lengths.

Definition 6.19. *Let $\tilde{R} = \mathbb{F}_2 + v\mathbb{F}_2 = \{0, 1, v, v+1\}$ be a ring where $v^2 = v$ and the automorphism θ defined previously. A subset $\tilde{\mathcal{C}}$ of $\tilde{R}^n$ is called a* skew cyclic code *(θ-cyclic code) of length n if the two following conditions hold*

1. *$\tilde{\mathcal{C}}$ is a R-submodule of R^n;*
2. *if $c = (c_0, c_1, \cdots, c_{n-1}) \in \tilde{\mathcal{C}}$ then $(\theta(c_{n-1}), \theta(c_0), \cdots, \theta(c_{n-2})) \in \tilde{\mathcal{C}}$.*

The ring $\tilde{R}_n = \tilde{R}[x;\theta]/(x^n - 1)$ denotes the quotient ring of $\tilde{R}[x;\theta]$ by the (left) ideal $(x^n - 1)$. Let $f(x) \in \tilde{R}_n$ and $r(x) \in \tilde{R}[x;\theta]$, we define the multiplication from the left as follows.

$$r(x) * (f(x) + (x^n - 1)) = r(x) * f(x) + (x^n - 1) \tag{6.6}$$

for any $r(x) \in \tilde{R}[x;\theta]$. Define a map as follows

$$\xi : \tilde{R}^n \to \tilde{R}[x;\theta]/(x^n - 1)$$

$$(c_0, c_1, \cdots, c_{n-1}) \to c_0 + c_1 x + c_2 x^2 + \cdots + c_{n-1} x^{n-1}.$$

Clearly, ξ is an $\tilde{R}$-module isomorphism map which implies that each element $(c_0 + c_1 \cdots + c_{n-1})$ of $\tilde{R}^n$ can be identified with the polynomial $c(x) = c_0 + c_1 x + c_2 x^2 + \cdots + c_{n-1} x^{n-1}$ of $\tilde{R}_n$.

Lemma 6.20. *([1, Lemma 1]) If n is even, and $x^n - 1 = g(x) * f(x)$ in $\tilde{R}[x;\theta]$, then we have:*

$$x^n - 1 = g(x) * f(x) = f(x) * g(x).$$

The following proposition gives the structure of the skew cyclic codes over $\tilde{R}_n$.

Proposition 6.21. *([1, Corollary 5]) Let $\tilde{\mathcal{C}}$ be a skew cyclic code in $\tilde{R}_n$. Then*

1. *If a polynomial $g(x)$ of least degree in $\tilde{\mathcal{C}}$ is a monic then $\tilde{\mathcal{C}} = (g(x))$, where $g(x)$ is (skew) right divisor of $x^n - 1$.*
2. *If $\tilde{\mathcal{C}}$ contains some monic polynomials but no polynomial $f(x)$ of least degree in $\tilde{\mathcal{C}}$ is monic, then $\tilde{\mathcal{C}} = (f(x), g(x))$, where $g(x)$ is a monic polynomial of least degree in $\tilde{\mathcal{C}}$ and $f(x) = vf_1(x)$ or $f(x) = (v+1)f_1(x)$ for some binary polynomial $f_1(x)$.*

3. *If $\tilde{\mathcal{C}}$ does not contain any monic polynomials. Then $\tilde{\mathcal{C}} = (f(x))$ where $f(x) = vf_1(x)$ or $f(x) = (v+1)f_1(x)$ and $f_1(x)$ is a binary polynomial that divides $x^n - 1$.*

Now, we are interesting in constructing of $[n, d]$-DNA skew cyclic codes. To this end, we start by defining such codes.

Definition 6.22. *Let $1 \leq d \leq n-1$ be a positive real number. A skew cyclic code $\tilde{\mathcal{C}}$ over $\tilde{R}$ is said to be a $[n, d]$-DNA skew cyclic code if the following conditions hold.*

1. *$\tilde{\mathcal{C}}$ is a skew cyclic code, that is, $\tilde{\mathcal{C}}$ is a $\tilde{R}$-submodule of $\tilde{R}_n$;*
2. *for any codeword $X \in \tilde{\mathcal{C}}$: $(X)^{rc} \neq (X)$ and $(X)^{rc} \in (C)$;*
3. *$d_H(X, Y) \leq d$ for any $X, Y \in C$.*

6.2.1 The Reverse-Complement DNA Skew Cyclic Codes over $\tilde{R}$

In this subsection, we give conditions on the existence of the reverse-complement cyclic codes of an even length n over the ring $\tilde{R}$. In Table 8 we present all codewords of the DNA skew cyclic code of length 10 and minimal Hamming distance 2.

Let $v = (a_0, a_1, \cdots, a_{n-2}, a_{n-1})$ be a vector in $\tilde{R}_n$, the reverse of the vector v is $v^r = (a_{n-1}, a_{n-2}, \cdots, a_1, a_0)$. Let $f(x)$ be the polynomial corresponding of the vector v such that $f(x) = a_0 + a_1x + \cdots + a_{n-1}x^{n-1}$. To get the polynomial corresponding of the vector v^r in $R[x;\theta]$, we multiply the right of the polynomial $f(x^{-1})$ by x^{n-1} leading to $f(x^{-1})x^{n-1} = a_0x^{n-1} + a_1\theta(1)x^{n-2} + \cdots + a_{n-2}\theta^{n-2}(1)x + a_n\theta^{n-1}(1) = a_{n-1} + a_{n-2}x + \cdots + a_1x^{n-2} + a_0x^{n-1}$. The polynomial corresponding of the vector v^r denoted by $f^*(x)$.

Definition 6.23. *Let $f(x)^* = f(x^{-1}) * x^{deg(f)}$ be the reciprocal polynomial of a given $f(x)$ in $\tilde{R}[x;\theta]$. Then the polynomial f is called* self-reciprocal *if f coincides with f^*.*

Example 6.24. Let f be a polynomial in $\tilde{R}[x;\theta]$ given by $f(x) = x^3 + vx^2 + (v+1)x + v$. The polynomial f represents the DNA sequence $X(ACTC)$.

We get the reverse of the sequence X via $f^*(x)$.

$$\begin{aligned} f^*(x) &= f(x^{-1})x^3 \\ &= \theta^3(1) + v\theta^2(1)x + (v+1)\theta(1)x^2 + v\theta^0(1)x^3 \\ &= vx^3 + (v+1)x^2 + vx + 1. \end{aligned}$$

The reverse of the DNA sequence of X is given by $(CTCA)$.

Notice that the definition of reciprocal polynomial over $\tilde{R}[x,\theta]$ is being different from the one defined over a commutative ring. Indeed, in the non-commutative ring $\tilde{R}[x,\theta]$, we use the right multiplication over the automorphism θ and the multiplication over $\tilde{R}[x,\theta]$.

Lemma 6.25. *Let $f(x)$ and $g(x)$ be polynomials in $\tilde{R}[x,\theta]$ with $deg(f(x)) \geq deg(g(x))$. Then the following assertions hold.*

(i) $(f(x)g(x))^* = f(x)^*g(x)^*$;

(ii) $(f(x)+g(x))^* = f(x)^* + g(x)x^{deg(f)-deg(g)}$.

Proof. *Let us prove assertion (i). One have $f(x) = \sum_{i=0}^{n} a_i x^i$ and $g(x) = \sum_{j=0}^{p} b_j x^j$, with $deg(f) \geq deg(g)$. Therefore,*

$$f(x)g(x) = \sum_{k=0}^{n+p}\sum_{i=0}^{k} a_i\theta^i(b_{k-i})x^k.$$

From Definition 6.2.1,

$$(f(x)g(x))^* = (\sum_{k=0}^{n+p}\sum_{i=0}^{k} a_i\theta^i(b_{k-i})x^{-k})x^{n+p}.$$

Thus

$$(f(x)g(x))^* = \sum_{k=0}^{n+p}\sum_{i=0}^{k} a_i\theta^i(b_{k-i})x^{n+p-k}.$$

Again from Definition 6.2.1 we have

$$f(x)^* = \sum_{i=0}^{n} a_i\theta^i(1)x^{n-i} = \sum_{i=0}^{n} a_i x^{n-i}$$

and

$$g(x)^* = \sum_{j=0}^{p} b_j \theta^j(1) x^{p-j} = \sum_{j=0}^{p} b_j x^{p-j}.$$

Consequently we have

$$f(x)^* g(x)^* = \sum_{k=0}^{n+p} \sum_{i=0}^{k} a_i \theta^i(b_{k-i}) x^{n+p-k}.$$

The result follows.
Now let us prove assertion (ii). *From the Definition 6.2.1, we have*

$$\begin{aligned}(f(x)+g(x))^* &= (f+g)^*(x) = ((f+g)(x^{-1}))x^{deg(f)} \\ &= (f(x^{-1})+g(x^{-1}))x^{deg(f)} = (f(x^{-1})x^{deg(f)} + g(x^{-1})x^{deg(f)}) \\ &= (f^*(x) + g(x^{-1})x^{deg(f)}) = f^*(x) + g(x^{-1})x^{deg(g)}x^{deg(f)-deg(g)} \\ &= f^*(x) + g^*(x)x^{deg(f)-deg(g)},\end{aligned}$$

which completes the proof. □

In the following we are interested in providing necessary conditions for $\tilde{\mathcal{C}}$ *to be a reverse-complement code.*

Theorem 6.26. *Let* $\tilde{\mathcal{C}} = (f(x))$ *be a skew cyclic code in* $\tilde{R}_n$, *where* $f(x)$ *is monic polynomial of minimal degree. If* $\tilde{\mathcal{C}}$ *is reverse-complement then the polynomial* $f(x)$ *is self-reciprocal and* $v(x^n-1)/(x-1) \in \tilde{\mathcal{C}}$.

Proof. *Let* $\tilde{\mathcal{C}} = (f(x))$ *be a skew cyclic code over* $\tilde{R}$, *where* $f(x)$ *is monic polynomial of minimal degree in* $\tilde{\mathcal{C}}$. *We know that*

$$(0, 0, \cdots, 0) \in \tilde{\mathcal{C}},$$

since $\tilde{\mathcal{C}}$ *is reverse-complement then*

$$(0, 0, \cdots, 0)^{rc} \in \tilde{\mathcal{C}}$$

i,e.;

$$(\hat{0}, \hat{0}, \cdots, \hat{0}) = (v, v, \cdots, v) \in \tilde{\mathcal{C}},$$

this vector correspond of the polynomial

$$v + vx + \cdots + vx^{n-1} = v(x^n-1)/(x-1) \in \tilde{\mathcal{C}}.$$

We have that $f(x)$ is monic polynomial of minimal degree in $\tilde{\mathcal{C}}$, where $f(x) = 1 + a_1x + \cdots + x^t$, the vector correspond to the polynomial $f(x)$ is $(1, a_1, \cdots, 0, 0, \cdots, 0)$, since $\tilde{\mathcal{C}}$ is reverse-complement and linear, then

$$(1, a_1, \cdots, 0, 0, \cdots, 0)^{rc} \in \tilde{\mathcal{C}},$$

i.e.,

$$f^{rc}(x) = v + vx + \cdots + vx^{n-t-2} + (v+1)x^{n-t-1} + a_{t-1}x^{n-t} + \cdots + a_1x^{n-2} + vx^{n-1}$$

$$= f^{rc}(x) + v(x^n - 1)/(x - 1) \in \tilde{\mathcal{C}}.$$

This implies that

$$x^{n-t-1} + (\hat{a_{t-1}} + v)x^{n-t} + \cdots + (\hat{a}_1 + v)x^{n-2} + x^{n-1} \in \tilde{\mathcal{C}}.$$

Multiplying on the right by x^{t+1-n}, we obtain,
$(1 + (\hat{a}_{t-1} + v)\theta(1)x + \cdots + (\hat{a}_1 + v)\theta^{t-1}(1)x^{t-1} + \theta^t(1)x^t)x^{t-n-1} \in \tilde{\mathcal{C}}$.
Hence,

$$(1 + (\hat{a}_{t-1} + v)x + \cdots + (\hat{a}_1 + v)x^{t-1} + x^t) \in \tilde{\mathcal{C}},$$

which implies (thanks to Equation (6.5)) that

$$f^*(x) = 1 + a_{t-1}x + \cdots + x^t \in \tilde{\mathcal{C}}.$$

Since $\tilde{\mathcal{C}} = (f(x))$, there exists $q(x) \in R[x, \theta]$ such that $f^*(x) = q(x)f(x)$, one necessary have $q(x) = 1$, that is $f^*(x) = f(x)$. □

Theorem 6.27. *Let $\tilde{\mathcal{C}} = (vf_1(x))$ be a skew cyclic code in $\tilde{R}_n$, where $f_1(x)$ is a monic binary polynomial of lowest degree with $f_1(x)|(x^n - 1)$. If $\tilde{\mathcal{C}}$ is a reverse-complement code then $f_1(x)$ is self-reciprocal.*

Proof. *Let $f_1(x) = 1 + a_1x + a_2x + \cdots + x^r$ be a binary polynomial. The vector corresponds to $f_1(x)$ is*

$$v = (1, a_1, \cdots, a_{r-1}, 1, 0, 0, 0, \cdots, 0, 0).$$

Hence

$$v^{rc} = (\hat{0}, \hat{0}, \hat{0}, \cdots, \hat{0}, \hat{1}, \hat{a}_{r-1}, \cdots, \hat{a}_1, \hat{1}).$$

These vectors correspond of the polynomial

$$f_1^{rc}(x) = v + vx + \cdots + vx^{n-r-2} + (v+1)x^{n-r-1} + \hat{a}_{n-r}x^{n-r} + \cdots + \hat{a}_1x^{n-2} + (v+1)x^{n-1}$$

$$= f_1^{rc} + v(x^n - 1)(x - 1).$$

Since $\tilde{\mathcal{C}}$ *is a linear code, then* $f_1^{rc} + v(x^n - 1)(x - 1) \in \tilde{\mathcal{C}}$. *Therefore*

$$x^{n-r-1} + (\hat{a}_{r-1} + v)x^{n-1} + \cdots + (\hat{a}_1 + v)x^{n-2} + x^{n-1} \in \tilde{\mathcal{C}},$$

multiplying by x^{-n+r+1}, *we obtain*

$$1 + (\hat{a}_{r-1} + v)\theta(1)x^1 + \cdots + (\hat{a}_1 + v)\theta^{r-1}(1)x^{r-1} + 1\theta^r(1)x^r \in \tilde{\mathcal{C}}.$$

Then

$$1 + (\hat{a}_{r-1} + v)x^1 + \cdots + (\hat{a}_1 + v)x^{r-1} + x^r \in \tilde{\mathcal{C}}.$$

By Equation (6.5), we obtain

$$f_1^*(x) = 1 + a_{r-1}x^1 + \cdots + a_1x^{r-1} + x^r \in \tilde{\mathcal{C}},$$

hence

$$vf_1^*(x) = 1 + v(a_{r-1}x^1 + \cdots + a_1x^{r-1} + x^r) \in \tilde{\mathcal{C}}$$

by Corollary 6.21, we have $vf_1^*(x) = vf_1(x)q(x)$, *one necessary have* $q(x) = 1$. *Then* $f_1^*(x) = f_1(x)$. □

Theorem 6.28. *Let* $\tilde{\mathcal{C}} = (f(x), g(x))$ *be a skew cyclic code in* $\tilde{R}_n$, *where* $f(x)$ *is a polynomial of minimal degree in* $\tilde{\mathcal{C}}$ *and is not a monic polynomial,* $g(x)$ *is a polynomial of least degree among the monic polynomials in* $\tilde{\mathcal{C}}$. *If* $\tilde{\mathcal{C}}$ *is a reverse-complement code then* $f(x)$ *and* $g(x)$ *are self-reciprocal.*

Proof. *The proof is similar to the proof of the Theorem 6.26 and of Theorem 6.27.* □

In the following, we provide sufficient conditions for $\tilde{\mathcal{C}}$ *being reverse-complement.*

Theorem 6.29. *Let* $\tilde{\mathcal{C}} = (f(x))$ *be a skew cyclic codes in* $\tilde{R}_n$, *where* $f(x)$ *is monic polynomial of the degree minimal in* $\tilde{\mathcal{C}}$. *If* $v(x^n - 1)/(x - 1) \in \tilde{\mathcal{C}}$ *and* $f(x)$ *is self-reciprocal then* $\tilde{\mathcal{C}}$ *is reverse-complement.*

Proof. *Let* $f(x) = 1 + a_1x + a_2x^2 + \cdots + a_{r-1}x^{r-1} + x^r$ *be a monic polynomial of the degree minimal in* $\tilde{\mathcal{C}}$ *and* $c(x) \in \tilde{\mathcal{C}}$, *we ha ve* $c(x) = q(x)f(x)$ *where* $q(x) \in R[x, \theta]$.
$c(x)^* = (q(x)f(x))^*$, *by the Lemma 6.25 we have* $c(x)^* = q(x)^*f(x)^*$, *since*

$f(x)$ is self-reciprocal then $c(x)^* = q(x)^* f(x) \in \tilde{\mathcal{C}}$ for all $c(x) \in \tilde{\mathcal{C}}$. Recall that we have

$$v + vx + \cdots + vx^{n-1} \in \tilde{\mathcal{C}}. \tag{6.7}$$

Now, let $c(x) = c_0 + c_1 x + c_2 x^+ \cdots + c_t x^t$, we multiply the right polynomial $c(x)$ by x^{n-t-1} we obtain $c(x) * x^{n-t-1} = c_0 + c_1\theta(1)x + c_2\theta^2(1)x^2 + \cdots + c_t\theta^t(1)x^t$, then

$$c(x) * x^{n-t-1} = c_0 x^{n-t-1} + c_1 x^{n-t} + \cdots + c_t x^{n-1} \in \tilde{\mathcal{C}}. \tag{6.8}$$

Combining (6.7) and (6.8) we obtain

$$(v + vx + \cdots + vx^{n-t-2} + (c_0 + v)x^{n-t-1} + \cdots + (c_t + v)x^{n-1}) \in \tilde{\mathcal{C}}, \tag{6.9}$$

leading to the following equality (using Equation (6.5) we have that $c_i + v = \hat{c}_i$). Then we obtain

$$v + vx + \cdots + vx^{n-t-2} + \hat{c}_0 x^{n-t-1} + \hat{c}_1 x^{n-t} + \cdots + \hat{c}_{t-1} x^{n-2} + \hat{c}_t x^{n-1} = (c(x))^{rc}.$$

Therefore, $(c^*(x)^{rc})^* = c(x)^{rc} \in \tilde{\mathcal{C}}$. □

Using similar arguments as those used in the proof of Theorem 6.29, we can prove the two following statements.

Theorem 6.30. *Let $C = (vf_1(x))$ be a skew cyclic code in $\tilde{R}_n$, where $f_1(x)$ is a monic binary polynomial of lowest degree with $f_1(x)|(x^n - 1)$. If $v(x^n - 1)/(x - 1) \in \tilde{\mathcal{C}}$ and $f_1(x)$ is self-reciprocal, then $\tilde{\mathcal{C}}$ is reverse-complement.*

Theorem 6.31. *Let $C = (f(x), g(x))$ be a skew cyclic codes in $\tilde{R}_n$, where $f(x)$ is a polynomial of degree minimal in $\tilde{\mathcal{C}}$ and is not monic polynomial. Let $g(x)$ be a polynomial of least degree among monic polynomial in $\tilde{\mathcal{C}}$. If $v(x^n - 1)/(x - 1) \in \tilde{\mathcal{C}}$ and $f(x)$ and $g(x)$ are self-reciprocal, then $\tilde{\mathcal{C}}$ is reverse-complement.*

6.2.2 Binary Image of DNA Skew Cyclic Codes

Recall that the Gray map φ from $\mathbb{F}_2 + v\mathbb{F}_2$ to $\mathbb{F}_2$, is defined as follows: for each element of $\mathbb{F}_2 + v\mathbb{F}_2$ expressed as $a + vb$, where $a, b \in \mathbb{F}_2$ maps $\varphi(a + vb) = (a + b, a)$, that is, $0 \mapsto (0, 0)$, $1 \mapsto (1, 1)$, $v + 1 \mapsto (0, 1)$, $v \mapsto (1, 0)$.
The linearity of φ comes straightforwardly from the definition of the Gray map.

We can obtain the binary image of the DNA code from the maps φ and ψ as well as the DNA alphabet onto the set of length 2 binary word given by $G \mapsto (0, 0)$, $A \mapsto (1, 1)$, $T \mapsto (0, 1)$, $C \mapsto (1, 0)$. We have the following property of binary image of the DNA skew cyclic code.

Corollary 6.32. *The map $R \to \mathbb{F}_2^{2n}$ is distance preserving linear isometry, hence if $\tilde{\mathcal{C}}$ is DNA skew cyclic code over R, then $\varphi(\tilde{\mathcal{C}})$ is a DNA skew quasi-cyclic code of length $2n$ and of index 2.*

Proof. *The proof is similar to the one of Lemma 6.16* □

Table 6.3: A DNA Cyclic Code associated to $\mathcal{C} = \langle u^4 f_0 f_1 \rangle$ given in (6.4)

GGGGGGGGGGGGGGGGGGGG	CCCCCCCCCCCCCCCCCCCC
CTCGGGCTCCTCCTCGGGGGG	GAGCCCGAGGAGGAGCCCCCC
GGGCTCGGGCTCTGTTGTTGT	CCCGAGCCCGAGACAATAACA
TGTGGGCTCGGGCTCTGTTGT	ACACCCGAGCCCGAGACAACA
TGTTGTGGGCTCGGGCTCTGT	ACAACACCCGAGCCCGAGACA
TGTTGTTGTGGGCTCGGGCTC	ACAACAACACCCGAGCCCGAG
CTCTGTTGTTGTGGGCTCGGG	GAGACAACAACACCCGAGCCC
GGGCTCTGTTGTTGTGGGCTC	CCCGAGACAACAACACCCGAG
TATGGGTATTATTATGGGGGG	ATACCCATAATAATACCCCCC
GGGTATGGGTATTATTATGGG	CCCATACCCATAATAATACCC
GGGGGGTATGGGTATTATTAT	CCCCCCATACCCATAATAATA
TATGGGGGGTATGGGTATTAT	ATACCCCCCATACCCATAATA
TATTATGGGGGGTATGGGTAT	ATAATACCCCCCATACCCATA
TATTATTATGGGGGGTATGGG	ATAATAATACCCCCCATACCC
GGGTATTATTATGGGGGGTAT	CCCATAATAATACCCCCCATA
TGTGGGTGTTGTTGTGGGGGG	ACACCCACAACAACACCCCCC
GGGTGTGGGTGTTGTTGTGGG	CCCACACCCACAACAACACCC
GGGGGGTGTGGGTGTTGTTGT	CCCCCCACACCCACAACAACA
TGTGGGGGGTGTGGGTGTTGT	ACACCCCCCACACCCACAACA
TGTTGTGGGGGGTGTGGGTGT	ACAACACCCCCCACACCCACA
TGTTGTTGTGGGGGGTGTGGG	ACAACAACACCCCCCACACCC
GGGTGTTGTTGTGGGGGGTGT	CCCACAACAACACCCCCCACA
CTCGGGCTCTGTTGTTGTGGG	GAGCCCGAGACAACAACACCC
GGGCTCGGGCTCTGTTGTTGT	CCCGAGCCCGAGACAACAACA
TGTGGGCTCGGGCTCTGTTGT	ACACCCGAGCCCGAGACAACA
TGTTGTGGGCTCGGGCTCTGT	ACAACACCCGAGCCCGAGACA
TGTTGTTGTGGGCTCGGGCTC	ACAACAACACCCGAGCCCGAG
CTCTGTTGTTGTGGGCTCGGG	GAGACAACAACACCCGAGCCC
GGGCTCTGTTGTTGTGGGCTC	CCCGAGACAACAACACCCGAG
GGGGGGCTCGGGCTCCTCCTC	CCCCCCGAGCCCGAGGAGGAG
CTCGGGGGGCTCGGGCTCCTC	GAGCCCCCCGAGCCCGAGGAG
CTCCTCGGGGGGCTCGGGCTC	GAGGAGCCCCCCGAGCCCGAG

Table 6.4: A DNA Cyclic associated to $\mathcal{C} =< f_1 f_2 >$ given in (6.4)

GGGGGGGGGGGGGGGGGGGG	CCCCCCCCCCCCCCCCCCCC
GGAGGAGGAGGAGGAGGAGA	CCTCCTCCTCCTCCTCCTCT
GGCGGCGGCGGCGGCGGCGC	CCGCCGCCGCCGCCGCCGCG
GGTGGTGGTGGTGGTGGTGT	CCACCACCACCACCACCACA
AGGAGGAGGAGGAGGAGGAG	TCCTCCTCCTCCTCCTCCTC
AGAAGAAGAAGAAGAAGAAGA	TCTTCTTCTTCTTCTTCTTCT
AGCAGCAGCAGCAGCAGCAGC	TCGTCGTCGTCGTCGTCGTCG
AGTAGTAGTAGTAGTAGTAGT	TCATCATCATCATCATCATCA
CGGCGGCGGCGGCGGCGGCGG	GCCGCCGCCGCCGCCGCCGCC
CGACGACGACGACGACGACGA	GCTGCTGCTGCTGCTGCTGCT
CGCCGCCGCCGCCGCCGCCGC	GCGGCGGCGGCGGCGGCGGCG
CGTCGTCGTCGTCGTCGTCGT	GCAGCAGCAGCAGCAGCAGCA
TGGTGGTGGTGGTGGTGGTGG	ACCACCACCACCACCACCACC
TGATGATGATGATGATGATGA	ACTACTACTACTACTACTACT
TGCTGCTGCTGCTGCTGCTGC	ACGACGACGACGACGACGACG
TGTTGTTGTTGTTGTTGTTGT	ACAACAACAACAACAACAACA
GAGGAGGAGGAGGAGGAGGAG	CTCCTCCTCCTCCTCCTCCTC
GAAGAAGAAGAAGAAGAAGAA	CTTCTTCTTCTTCTTCTTCTT
GACGACGACGACGACGACGAC	CTGCTGCTGCTGCTGCTGCTG
GATGATGATGATGATGATGAT	CTACTACTACTACTACTACTA
AGGAGGAGGAGGAGGAGGAGG	TCCTCCTCCTCCTCCTCCTCC
AAAAAAAAAAAAAAAAAAAAA	TTTTTTTTTTTTTTTTTTTTT
AACAACAACAACAACAACAAC	TTGTTGTTGTTGTTGTTGTTG
AATAATAATAATAATAATAAT	TTATTATTATTATTATTATTA
CAGCAGCAGCAGCAGCAGCAG	GTCGTCGTCGTCGTCGTCGTC
CAACAACAACAACAACAACAA	GTTGTTGTTGTTGTTGTTGTT
CACCACCACCACCACCACCAC	GTGGTGGTGGTGGTGGTGGTG
CATCATCATCATCATCATCAT	GTAGTAGTAGTAGTAGTAGTA
TAGTAGTAGTAGTAGTAGTAG	ATCATCATCATCATCATCATC
TAATAATAATAATAATAATAA	ATTATTATTATTATTATTATT
TACTACTACTACTACTACTAC	ATGATGATGATGATGATGATG
TATTATTATTATTATTATTAT	ATAATAATAATAATAATAATA

Table 6.5: DNA cyclic codes associate to $\mathcal{C} = \langle f_0, uf_1, u^2f_2, u^3f_3, u^4f_4, u^5f_5\rangle$

The Code $\mathcal{C}$	Size of the code $\mathcal{C}$
$\langle u^3f_1, u^4f_2, u^5f_3\rangle$	1125899906842624
$\langle u^5f_2\rangle$	512
$\langle f_3, u^5f_2\rangle$	4611686018427387904
$\langle u^4f_1, u^5f_3\rangle$	8589934592

Table 6.6: Binary image of the codons given by Table 5.1

GGG	000000	CCC	111111	TAT	000001	ATA	111110
GGA	011111	CCT	100000	TAC	100001	ATG	011110
GGC	101111	CCG	010000	TAA	010001	ATT	101110
GGT	001111	CCA	110000	TAG	110001	ATC	001110
AGG	110111	TCC	001000	CAT	001001	GTA	110110
AGA	010111	TCT	101000	CAC	011001	GTG	100110
AGC	100111	TCG	011000	CAA	011001	GTT	100110
AGT	000111	TCA	111000	CAG	111001	GTC	000110
CGG	111011	GCC	000100	AAT	000101	TTA	111010
CGA	011011	GCT	100100	AAC	100101	TTG	011010
CGC	101011	GCG	010100	AAA	010101	TTT	101010
CGT	001011	GCA	110100	AGG	110101	TCC	001010
TGG	110011	ACC	001100	GAT	001101	CTA	110010
TGA	010011	ACT	101100	GAC	101101	CTG	010010
TGC	100011	ACG	011100	GAA	011101	CTT	100010
TGT	000011	ACA	111100	GAG	111101	CTC	000010

Table 6.7: A binary image of DNA cyclic codes of length 7 given Table 5.2

The code $\mathcal{C}$	Length of $\varphi(\mathcal{C})$	$d_H(\varphi(\mathcal{C}))$	Size of the Code $\varphi(\mathcal{C})$
$\langle u^2f_0\rangle$	42	12	4096
$\langle u^2f_1\rangle$	42	18	256
$\langle u^2f_2\rangle$	42	18	256

Table 6.8: DNA skew cyclic code of length 10 and minimal distance 2

GGGGGGGGGG	CCCCCCCCCC	CCCCCGGGGG	GGGGGCCCCC
GGGGCCCCCG	CCCCGGGGGC	CCCCGCCCCG	GGGGCGGGGC
GGGTTTTTGG	CCCAAAAACC	CCCATAAACG	GGGTATTTGC
GGGTAAAACG	CCCATTTTGC	CCGGGCCGGG	GGCCCGGCCC
GGCCCCCGGG	CCGGGGGCCC	CCGGCGGCCG	GGCCGCCGGC
GGCCGGGCCG	CCGGCCCGGC	CCGTATTACG	GGCATAATGC
GGCAAAATCG	CCGTTTTAGC	CCGTTAATTG	GGCAATTAAC
GGCAAAATGG	CCGTTTTACC	CATTAACGGG	GTAATTGCCC
GTAAAACGGG	CATTTTGCCC	CAGTATGCCG	GTCATACGGC
GTAACCATTG	CATTGGTAAC	CAAGCGTACG	GTTCGCATGC
GTACGGTACG	CATGCCATGC	CAATTACCGG	GTTAATGGCC
GTACCCATGG	CTAGGGTACC	CAAAATGGGG	GTTTTACCCC
GATTTTGGGG	CTAAAACCCC	CAAATACCCG	GTTTATGGGC
GTTTAACCCG	CAAATTGGGC	CAACGCAACG	GTTGCGTTGC
GTTGCCAACG	CAACGGTTGC	CAACCGTTGG	GTTGGCAACC
GTTGGGTTGG	CAACCCAACC	CCACGCAACG	GGTGCGTTGC

Bibliography

[1] T. Abualrub, N. Aydin and P. Seneviratne, On Θ-cyclic codes over $\mathbb{F}_2 + v\mathbb{F}_2$, Australian Journal of Combinatorics 54, 115-126, 2012.

[2] T. Abualrub, A.Ghrayeb, X. N. Zeng, Construction of cyclic codes over $\mathbb{F}_4$ for DNA computing, Journal of the Franklin Ins. 343, 488-457, 2006.

[3] L. Adleman, Molecular computation of the solution to cobinatorial problems, Science 266-1021-1024, 1994.

[4] C. Alf-Steinberger, The genetic code and error transmission. Proc. Natl. Acad. Sci. USA, 64, 584-591, 1969.

[5] M. B. Bechet. Bias de codons et régulation de la traduction chez les bactéries et le phages. PhD thesis, University of Paris 7, 2007.

[6] N. Bennenni, K. Guenda and T.A. Gulliver, Greedy construction of DNA codes and new bounds, conf ACA2014, New York, 2014.

[7] N. Bennenni, K. Guenda and S. Mesnager, DNA cyclic codes over rings. AMC. To appear. 2016.

[8] N. Bennenni, K. Guenda and T.A. Gulliver, Constructing DNA Codes with Optimal Thermodynamic and Combinatorial Properties, conf, ICCC 2015, algiers, 2015.

[9] N. Bennenni, K. Guenda and T.A. Gulliver, Greedy construction for DNA computing and New Bounds, AAECC , To appear. 2016.

[10] M.A. Bishop, A.G. D'Yachkov, A.J. Macula, T.E. Renz and V.V. Rykov. Free energy gap and statistical thermodynamic fidelity of DNA codes. J. Comp. Biol. **14***(8), 1088–1104 (2007).*

[11] *Y.M. Chee and S. Ling. Improved lower bounds for constant GC-content DNA codes. IEEE Trans. Inform. Theory.* **54***(1), 391–394 (2008).*

[12] *H. Dinh and S.R. Lopez-Permouth, Cyclic and negacyclic codes over finite chain rings, IEEE Trans. Infrom. Theory, 50: 1728-1744, 2004.*

[13] *S. T. Dougherty, J. Lark Kim and H. Kulosman. MDS code over finit principal ideal rings. Des. Codes Cryptogr.50:77-92. 2009*

[14] *A. D'yachkov, Lectures on DNA codes, http://arxiv.org/pdf/1401.7492v1.pdf. 2014.*

[15] *A. D'yachkove and A. N. Vorinina, DNA code over additive stem similarity, Prob. Inform. Transmission, 45(2), 124-144, 2009.*

[16] *W. C. Huffman and V. Pless, Fundamentals of error-correcting codes, Cambridge, 2003.*

[17] *N. Goldman, P. Bertone, S. Chen, C. Dessimoz, E. M. LeProust, B.Sipos, and E. Birney. Towards practical, high-capacity, low-maintenance information storage in synthesized DNA. Nature, 2013.*

[18] *K. Guenda, T.A. Gulliver and S.A. Sheikholeslam. Lexicodes over rings. Des. Codes Cryptogr.* **72***(3), 749–763 (2014).*

[19] *T. Ericson, Bounds on the size of a code, Topics in Coding Theory VII, Lecture Notes in Control and Inform. Sci, Springer-Verlag, 128, 45–78, (1989).*

[20] *K. Guenda, T.A. Gulliver and P. Solé. On cyclic DNA codes. Proc. IEEE Int. Symp. Inform. Theory, 121-125, Istanbul, Jul. 2013.*

[21] *K. Guenda and T.A. Gulliver, Repeated Root Constacyclic Codes of the Length mp^s over $\mathbb{F}_p^r + u\mathbb{F}_p^r + \cdots + u^{e-1}\mathbb{F}_p^r$, J. Alg. App. 14(1), Feb. 2015.*

[22] *K. Guenda and T.A. Gulliver Construction of cyclic codes over $\mathbb{F}_2 + u\mathbb{F}_2$ for DNA computing. Appl. Algebra Eng. Commun. Comput.* **24***(6), 445-459, 2013.*

[23] *K. Guenda, T.A. Gulliver and S.A. Sheikholeslam, Lexicodes over rings, Des. Codes Cryptogr., 72(3), 749–763 (2014).*

[24] O.D. King. Bounds for DNA codes with constant GC-content. Electron. J. Combin. **10**, *#R33 (2003).*

[25] J.Y. Lee, S.-Y. Shin, T.H. Park and B.-T. Zhang, Solving traveling salesman problems with DNA molecules encoding numerical values, Biosystem 78(1-3), 39-47 (2004).

[26] F.J. MacWilliams and N.J.A. Sloane, The Theory of Error-Correcting Codes, North-Holland, 1977.

[27] M. Mansuripur,P.K. Khulbe, S.M. Kuebler, J.W. Perry, M.S. Giridhar and N. Peyghambarian: Information storage and retrievel usin macromolecules as storage media. University Arizona Technical Report. 2003.

[28] A. Marathe, A.E. Condon and R.M. Corn. On combinatorial DNA word design. J. Comp. Biol. **8***(3), 201–219 (2001).*

[29] O. Milenkovic and N. Kashyap, On the design of codes for DNA computing. IEEE Proceding Internationnal. (ISIT05), 2006.

[30] G. H. Notron and A. Salagean, on the Structure of Linear and Cyclic Codes over Finite Chain Ring. AAECC 10,489-506, 2000.

[31] E.S. Ristad and P.N. Yianilos. Learning string-edit distance. IEEE Trans. Anal. Mach. Intell. **20***(5), 522-532, 1998.*

[32] V. Rykov, A.J. Macula, D. Torny and P. White, DNA sequence and quaternary cyclic codes, IEEE International Syposium on iformation Theory (ISIT 2001), DC, p.248, 2001.

[33] R. Sanchez, E. Morgado and R. Grau, Gene Algebra from a Genetic Code Algebraic Structure, J. Math. Biol. 51,431-475, 2005.

[34] D.D. Shoemaker, D.A. Lashkari, D. Morris, M. Mittman and R.W. Davis. Quantitative phenotypic analysis of yeast deletion mutant using a highly parallel molecular bar-coding strategy. Nat. Genet. **14**, *450–456 (1996).*

[35] I. Siap, T. Abualrub and A. Ghrayeb, Cyclic DNA codes over ring $\mathbb{F}_2[u]/(u^2-1)$ *based on the deletion distance, Franklin Instute, (36), 731-740, 2009.*

[36] D.H. Smith, N. Aboluion, H. Montemanni and S. Perkins. Linear and nonlinear constructions of DNA codes with Hamming distance d and constant GC-content. Discr. Math. **311***(13), 1207–1219 (2011).*

[37] R. Soni and G. Prajapati, A Moderne review on DNA cryptographic techniques. IJARCSSE, Vol3, Issue 7, ISSN: 2277128X (2013).

[38] J. Sun. Bounds on edit metric codes with combinatorial DNA constraints. Master's Thesis, Brock University, (2009).

[39] http://www.codetables.de

[40] http://www.extremetech.com/extreme/199414-scientists-create-million-year-data-storage-with-dna