5-2008

A Study of Genetic Code by Combinatorics and Linear Algebra Approaches

Tanner Jennings Crowder
College of William and Mary

A Study of Genetic Code by Combinatorics and Linear Algebra Approaches

A thesis submitted in partial fulfillment of the
requirements for the degree of Bachelor of Arts with Honors in
Mathematics from the College of William and Mary in Virginia,
by

Tanner Jennings Crowder

Accepted for: Highest Honors

____

Adviser: Chi-Kwong Li

____

Sara Day

____

Jianjun Tian

____

Margaret Saha

Williamsburg, Virginia
April 2008

Abstract

The genetic code-based matrices constructed in this work and the corresponding hamming distance matrices are studied using combinatorics and linear algebra approaches. Recursive schemes for generating the matrices are obtained. Algebraic properties such as ranks, eigenvalues, and eigenvectors of the Hamming distance matrices are examined. The results lead to an easy calculation of the powers of the Hamming distance matrices. Moreover, a decomposition of the Hamming Distance matrices in terms of permutation matrices is obtained. The decomposition gives rise to hypercube structures to the genetic code based matrices. A new scheme is given to generate matrices where each entry is a 4-tuple, which counts the number of each nucleotide in the entries of the genetic code matrix. Connections and potential applications of the results will be discussed.

Keywords: Hamming Distance, Permutation Matrices, Gray Code, Genetic Code Eigenstructure

Acknowledgements

The author wishes extend gratitude to the members of the Honors Thesis Examining Committee, namely Professors Sarah Day, Margaret Saha, and Jianjun Paul Tian. The author would also like to thank his advisor, Dr. Chi-Kwong Li, for his support and dedication to this Honors Thesis. Also, thanks to Martin Saunders for proofreading this thesis. The thesis was also possible because of the support of the College of William and Mary NSF CSUMS Grant.

Contents

Chapter 1

Introduction

1.1 DNA and RNA

Genetic Code is the set of rules by which information encoded in RNA/DNA is translated into amino acid sequences in living cells. The bases for the encoded information are nucleotides. There are four nucleotide bases for RNA: Adenine, Uracil, Guanine, and Cytosine, which are labeled by the basis $\{A, U, G, C\}$ respectively. (In DNA Uracil is replaced by Thymine (T)). Canonical genetic code is a mapping between codons and the amino acids. Codons are tri-nucleotide sequences such that each triplet relates to an amino acid. For example, the codon CAG encodes the amino acid Glutamine. Amino acids are the basic building blocks of proteins. Canonical genetic code is the type that we will be studying.

The genetic code map is $g : C' \to A'$, where $C' = \{\{x_1x_2x_3\} : x_i \in \{A, C, G, U\}\}$. C' is the set of codons and A' is the set of amino acids and termination codons. The function, $g : C' \to A'$, is interesting because $g(x_1x_2x_3)$ is a surjection but not an injection. This is because there are 4^3 tri-nucleotide sequences and only 20 amino acids, plus the start and stop codons. More than one codon can represent the same amino acid; however, two different amino acids cannot be represented by one codon.

In general, genetic sequences can be long, so it is difficult to extract information or to observe patterns. The focus of this study is building matrices which will contain all length n nucleotide sequences and can efficiently represent the genetic sequences. Many studies have been devoted to examining how genetic code has evolved, more specifically: Is genetic code random, or is there a reason genetic information is encoded the way it is? Patterns that arise in genetic code suggest that the code is not random. One theory is that genetic code evolved to minimize the effects of mutations. Specifically there is a theory that nucleotide alphabet has grown in complexity since the origin of genetic code. One current aspect of research is examining the redundancy of genetic code and it's effect on the dynamic of evolution [1]. In Chapter 4, a graph $G = (V, E)$ will be considered, where V is the set of all length n genetic sequences, and E is the edge set where two vertices are adjacent if they differ by one nucleotide base. A *Hamilton circuit* will be given for that graph which may help analyze mutations in genetic code. A Hamilton circuit of a graph G is a circuit containing all vertices of the graph G, such that each vertex only appears once in the circuit [7].

1.2 Gray Code

Gray code is an encoding scheme with the property that two consecutive sequences only differ by one position [7]. For example, the classical binary representations for three and four are 011 and 100 respectively, but a Gray code representation for three and four is 011 and 010, respectively. In classical binary, 011 and 100 differ in all three positions, but in the Gray code representation 011 and 010 differ in only one position, namely the last position. A Gray code representation of nucleotides was proposed by Swanson [6]. He et al. [2, 3] studied the idea and built matrices out of the Gray code to investigate the symmetries and structure when looking at the distances between nucleotide sequences.

Define G_n to be all the Gray code sequences of length n, which can be generated by a recursive algorithm. G_n is constructed by taking the sequences from G_{n-1} and prepending a 0 to them then taking the sequences of G_{n-1} in reverse order and prepending a 1 to them; therefore $G_n = \{0||a_0, 0||a_1, \ldots, 0||a_{n-1}, 1||a_{n-1}, 1||a_{n-2}, \ldots, 1||a_0\}$, where $a_i \in G_{n-1}$. Note $a||b$ is the symbol for a concatenate b. To illustrate this process take $G_1 = \{0, 1\}$. Then by construction $G_2 = \{0||0, 0||1, 1||1, 1||0\} = \{00, 01, 11, 10\}$. To construct G_3 copy the entries and prepend a 0 to every string, (e.g. $000, 001, 011, 010$), and then copy the entries in reverse order and prepend a 1 to every string (e.g $110, 111, 101, 100$), so $G_3 = \{000, 001, 011, 010, 110, 111, 101, 100\}$.

Since $|G_n|$ doubles in size from $|G_{n-1}|$ and G_1 only has 2 entries, $|G_n| = 2^n$. It is well known that the graph of G_n has a Hamilton circuit, where two sequences are adjacent if and only if they differ in only one position. For example a Hamilton circuit for G_2 is $00 - 01 - 11 - 10 - 00$. Since Gray code has that property, it will a graph $G = (V, E)$, as described in section 1.1, will have the property.

Initially Gray code was intended for transmitting information where a change in one bit would distort the information less than if the information was encoded using the standard binary representation [7]. It is natural to represent genetic code in this manner because Gray code is designed to minimize the mismatches between the digit encoding adjacent bases and therefore minimizing the mismatches between nearby chromosome segments; thus the degree of mutation will be reduced [2, 3].

1.3 Genetic Code Matrices and Hamming Distance Matrices

Following He et al. [2, 3], the following correspondence for the nucleotides and two-bit Gray codes will be used: $C \sim \binom{0}{0}$, $U \sim \binom{1}{0}$, $G \sim \binom{1}{1}$, and $A \sim \binom{0}{1}$. The code-based matrix, which will contain all nucleotide strings of length n is defined as C_n. The Gray code sequences represented by C_n will be denoted by a $2^n \times 2^n$ matrix. Here are C_1, C_2, C_3 and their corresponding Gray code representations.

$$C_1 \sim \begin{array}{c} \\ 0 \\ 1 \end{array} \begin{array}{c} \begin{array}{cc} 0 & 1 \end{array} \\ \begin{pmatrix} \binom{0}{0} & \binom{1}{0} \\ \binom{0}{1} & \binom{1}{1} \end{pmatrix} \end{array} \quad \text{so} \quad C_1 = \begin{pmatrix} C & U \\ A & G \end{pmatrix}.$$

$$
C_2 \sim \begin{array}{c} \\ 00 \\ 01 \\ 11 \\ 10 \end{array} \begin{array}{c} \begin{array}{cccc} 00 & 01 & 11 & 10 \end{array} \\ \begin{pmatrix} \binom{00}{00} & \binom{01}{00} & \binom{11}{00} & \binom{10}{00} \\ \binom{00}{01} & \binom{01}{01} & \binom{11}{01} & \binom{10}{01} \\ \binom{00}{11} & \binom{01}{11} & \binom{11}{11} & \binom{10}{11} \\ \binom{00}{10} & \binom{01}{10} & \binom{11}{10} & \binom{10}{10} \end{pmatrix} \end{array}
$$

so

$$
C_2 = \begin{pmatrix} CC & CU & UU & UC \\ CA & CG & UG & UA \\ AA & AG & GG & GA \\ AC & AU & GU & GC \end{pmatrix}.
$$

And

$$
C_3 \sim \begin{array}{c} \\ 000 \\ 001 \\ 011 \\ 010 \\ 110 \\ 111 \\ 101 \\ 100 \end{array} \begin{array}{c} \begin{array}{cccccccc} 000 & 001 & 011 & 010 & 110 & 111 & 101 & 100 \end{array} \\ \begin{pmatrix} \binom{000}{000} & \binom{001}{000} & \binom{011}{000} & \binom{010}{000} & \binom{110}{000} & \binom{111}{000} & \binom{101}{000} & \binom{100}{000} \\ \binom{000}{001} & \binom{001}{001} & \binom{011}{001} & \binom{010}{001} & \binom{110}{001} & \binom{111}{001} & \binom{101}{001} & \binom{100}{001} \\ \binom{000}{011} & \binom{001}{011} & \binom{011}{011} & \binom{010}{011} & \binom{110}{011} & \binom{111}{011} & \binom{101}{011} & \binom{001}{011} \\ \binom{000}{010} & \binom{001}{010} & \binom{011}{010} & \binom{010}{010} & \binom{110}{010} & \binom{111}{010} & \binom{101}{010} & \binom{001}{010} \\ \binom{000}{110} & \binom{001}{110} & \binom{011}{110} & \binom{010}{110} & \binom{110}{110} & \binom{111}{110} & \binom{101}{110} & \binom{001}{110} \\ \binom{000}{111} & \binom{001}{111} & \binom{011}{111} & \binom{010}{111} & \binom{110}{111} & \binom{111}{111} & \binom{101}{111} & \binom{001}{111} \\ \binom{000}{101} & \binom{001}{101} & \binom{011}{101} & \binom{010}{101} & \binom{110}{101} & \binom{111}{101} & \binom{101}{101} & \binom{001}{101} \\ \binom{000}{100} & \binom{001}{100} & \binom{011}{100} & \binom{010}{100} & \binom{110}{100} & \binom{111}{100} & \binom{101}{100} & \binom{001}{100} \end{pmatrix} \end{array}
$$

so

$$
C_3 = \begin{pmatrix} CCC & CCU & CUU & CUC & UUC & UUU & UCU & UCC \\ CCA & CCG & CUG & CUA & UUA & UUG & UCG & UCA \\ CAA & CAG & CGG & CGA & UGA & UGG & UAG & UAA \\ CAC & CAU & CGU & CGC & UGC & UGU & UAU & UAC \\ AAC & AAU & AGU & AGC & GGC & GGU & GAU & GAC \\ AAA & AAG & AGG & AGA & GGA & GGG & GAG & GAA \\ ACA & ACG & AUG & AUA & GUA & GUG & GCG & GCA \\ ACC & ACU & AUU & AUC & GUC & GUU & GCU & GCC \end{pmatrix}.
$$

When $n = 3$, or is a multiple of 3, C_n contains nucleotide triplets, which are codons. Therefore interesting biological structure starts to appear in C_3.

The Hamming distance is a measure of how two strings of the same length differ. For example, the binary strings 001 and 011 have a Hamming distance 1, since there is only one difference in the second position. This is precisely the Hamming distance of the two strings giving the codon CAG because $CAG \sim \binom{001}{011}$–by construction. The Hamming distance is not exclusive to binary strings; the words "math" and "bath" have a Hamming distance 1 because they differ in the first position. The binary strings 010101 and 011110 have a Hamming distance 3. To get a better understanding of the Genetic code matrix and the recursion, the Hamming distance matrices, D_n, associated with C_n will be studied. D_1, D_2, D_3 are as follows:

$$D_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad D_2 = \begin{pmatrix} 0 & 1 & 2 & 1 \\ 1 & 0 & 1 & 2 \\ 2 & 1 & 0 & 1 \\ 1 & 2 & 1 & 0 \end{pmatrix},$$

$$D_3 = \begin{pmatrix} 0 & 1 & 2 & 1 & 2 & 3 & 2 & 1 \\ 1 & 0 & 1 & 2 & 3 & 2 & 1 & 2 \\ 2 & 1 & 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 2 & 1 & 0 & 1 & 2 & 3 & 2 \\ 2 & 3 & 2 & 1 & 0 & 1 & 2 & 1 \\ 3 & 2 & 1 & 2 & 1 & 0 & 1 & 2 \\ 2 & 1 & 2 & 3 & 2 & 1 & 0 & 1 \\ 1 & 2 & 3 & 2 & 1 & 2 & 1 & 0 \end{pmatrix}.$$

Each entry of D_n is the Hamming distance between the Gray code sequences that represent the nucleotides of C_n. The Hamming distance matrix gives substantial information about genetic code and yet requires less storage. Specifically it gives information about the composition of each entry in C_n. It shows how many possible U or A and C or G nucleotides are contained in each entry. However, it only shows how many total U's and A's (and therefore C's and G's) appear combined, not how many of each individual nucleotide appear. Also the order of the nucleotides is lost with this reduction.

As a primer to the proofs done in the later sections, an exercise in finding the Hamming distance of an arbitrary entry in D_n will be done.

1.4 Computing the Hamming Distance of an (i, j) entry of D_n

This section is devoted to showing an algorithm for finding the Hamming distance of an arbitrary (i, j) entry of D_n. It is useful to illustrate this technique because it is frequently used in the proofs of this study. This can be done with a few simple steps. First, one must translate i and j into Gray code. Take $i, j \in \mathbb{N}$ and convert i and j it to standard binary. To obtain the standard Gray code, check the right most digit in the binary string; if its neighbor to the left is a 1 then change the digit; if it is a zero leave the digit unaltered. Once that has been done to all the positions, i and j are in the standard Gray code representation. Finally, compute $i \oplus j$ and add up the number of 1's in the resulting string, which results in the Hamming distance. The symbol $\oplus$ is addition mod 2, or the exclusive or.

This algorithm yields the Hamming distance, because each entry in the Genetic code matrix is represented by the column number over the row number, when both are represented in Gray code. Furthermore when the exclusive or is computed on two binary string, two like digits go to zero and two different digits go to one; for example $1 \oplus 1 = 0$, $0 \oplus 0 = 0$, but $1 \oplus 0 = 1$. So the hamming distance is exactly the amount of times that $1 \oplus 0$ or $0 \oplus 1$ occurs in the Gray code representation for a given nucleotide string. Note that $1 \oplus 0$ and $0 \oplus 1$ happens precisely at each occurrence of U or A in a nucleotide string, respectively.

Chapter 2

Basic Results of D_n

In this chapter a recursive definition of D_n is presented, and its basic properties are discussed. Take D_1 and D_2 to be defined as the Hamming distance matrices for $n = 1, 2$ then

$$D_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \text{and} \quad D_2 = \begin{pmatrix} 0 & 1 & 2 & 1 \\ 1 & 0 & 1 & 2 \\ 2 & 1 & 0 & 1 \\ 1 & 2 & 1 & 0 \end{pmatrix}.$$

There is a clear recursive structure in these two matrices. Take D_1 as a building block of D_2. Notice that D_1 appears on the diagonal of D_2 and on the anti-diagonal where the 0's are replaced with 2's and the 1's are unaltered. Notice also that D_2 is symmetric and persymmetric, i.e. symmetric about its anti-diagonal. Therefore, $D_n = F_n D_n^t F_n$, where F_n is the anti-diagonal matrix. D_3 will exhibit a similar structure using D_2 as its building block, and also by definition of D_2, D_3 uses D_1 as well. This is the recursive structure that this thesis will exploit.

$$\begin{aligned} D_2 &= \begin{pmatrix} D_1 & D_1 \\ D_1 & D_1 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \\ 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \end{pmatrix} \\ &= \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \\ 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \end{pmatrix} \end{aligned}$$

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

Being able to recursively generate D_n is computationally valuable because, as discussed earlier, D_n physically stores less information than C_n and yet still gives insight to the structure of C_n. For example, when examining D_3, an entry having a Hamming distance of 3 implies that the nucleotide string is either AAA, UUU, or U's and A's in combination; U and A are the only nucleotides that yield a Hamming distance of 1. So if the Hamming distance is 2, it is known that the codons must contain either two A's, two U's, or a one A and one U. This extends to all D_n. The value of an entry in D_n is exactly how many A's and U's–and clearly C and G can be counted as well– are in the corresponding entry of C_n. Storing less information is important because these matrices not only grow exponentially in dimension, $2^n \times 2^n$, but also each entry of the matrix grows with the size of n. The Hamming distance matrix significantly decreases the amount of information per entry.

2.1 Recursive Structure of D_n

Theorem 2.1 *Suppose D_n is a matrix defined as in Section 1 and suppose that D_n has the form,*

$$D_n = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix},$$

where B_{ij} is a $2^{n-1} \times 2^{n-1}$ sub matrix. Then $B_{11} = B_{22}$ and $B_{12} = B_{21}$. Moreover

$$D_{n+1} = \begin{bmatrix} B_{11} & B_{12} & 2J_{n-1} + B_{11} & B_{12} \\ B_{12} & B_{11} & B_{12} & 2J_{n-1} + B_{11} \\ 2J_{n-1} + B_{11} & B_{12} & B_{11} & B_{12} \\ B_{12} & 2J_{n-1} + B_{11} & B_{12} & B_{11} \end{bmatrix},$$

where $J_{n-1} \in M_{2^{n-1}}$ with all entries equal to one.

Proof. It will first be shown that $B_{11} = B_{22}$ and $B_{12} = B_{21}$, and then the recursive structure of D_n will be shown inductively.

Let G_n be the ordered binary sequences of length n using the Gray code construction. Define $H(a, b)$ to be the Hamming distance between a and b. Then an (a, b) entry of D_n is defined by $H(a, b)$. For $a^{n-1} = (a_1, \ldots, a_{n-1})$ and $b^{n-1} = (b_1, \ldots, b_{n-1})$ in G_{n-1}, let $a = 0||a_1a_2 \ldots a_{n-1}, b = 0||b_1b_2 \ldots b_{n-1}, \tilde{a} = 1||a_{n-1} \ldots a_1, \tilde{b} = 1||b_{n-1} \ldots b_1 \in G_n$.

Assume

$$D_n = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}.$$

Then an (a, b) entry of B_{11} is the Hamming distance

$$H(a, b) = H(0a_1a_2 \ldots a_{n-1}, 0b_1b_2 \ldots b_{n-1}).$$

The $(\tilde{a}, \tilde{b})$ entry of B_{22} is $H(\tilde{a}, \tilde{b}) = H(1||a_{n-1} \ldots a_2a_1, 1||b_{n-1} \ldots b_2b_1) = H(0||a_{n-1} \ldots a_2a_1,$ $0||b_{n-1} \ldots b_2b_1) = H(0||a_1 \ldots a_{n-1}, 0||b_1 \ldots b_{n-1}) = H(a, b)$. Thus the $(\tilde{a}, \tilde{b})$ entry of B_{12} is equal to the (a, b) entry of B_{11}. Similarly the $(a, \tilde{b})$ entry of B_{12} is $H(a, \tilde{b}) = H(0||a_1a_2 \ldots a_{n-1},$ $1||b_{n-1}b_{n-2} \ldots b_1)$, which is the same as the $(\tilde{a}, b)$ entry of B_{21} equal to $H(1||a_{n-1}a_{n-2} \ldots a_1,$ $0||b_1b_2 \ldots b_{n-1})$. So $B_{11} = B_{22}$ and $B_{12} = B_{21}$. This implies that D_n is symmetric and persymmetric.

To show the recursive structure, recall

$$D_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix},$$

and

$$D_2 = \begin{bmatrix} 0 & 1 & 2 & 1 \\ 1 & 0 & 1 & 2 \\ 2 & 1 & 0 & 1 \\ 1 & 2 & 1 & 0 \end{bmatrix} = \begin{bmatrix} B_{11} & B_{12} & 2J_{n-1} + B_{11} & B_{12} \\ B_{12} & B_{11} & B_{12} & 2J_{n-1} + B_{12} \\ 2J_{n-1} + B_{11} & B_{12} & B_{11} & B_{12} \\ B_{12} & 2J_{n-1} + B_{11} & B_{12} & B_{11} \end{bmatrix}$$

Clearly the construction is true for D_1 and D_2. So assume that the construction is true for $D_1, D_2, \ldots, D_n$. Now consider $n \geq 3$, and

$$D_{n+1} = \begin{bmatrix} X_{11} & X_{12} & X_{13} & X_{14} \\ X_{21} & X_{22} & X_{23} & X_{24} \\ X_{31} & X_{32} & X_{33} & X_{34} \\ X_{41} & X_{42} & X_{43} & X_{44} \end{bmatrix} \quad \text{where } X_{ij} \in M_{2^{n-1}}.$$

Take any $(\acute{a}, \acute{b})$ entry of the X_{11} sub matrix, the distance will equal $H(00||a, 00||b) = H(a_1 \ldots a_{n-1}, b_1 \ldots b_{n-1})$ which is the (a, b) entry of the B_{11} because the first two positions have a Hamming distance of 0. Now take any $(\grave{a}, \grave{b})$ entry in X_{12}. The distance will be $H(00||a_1 \cdots a_{n-1}, 01||b_{n-1} \cdots b_1) = H(a, \tilde{b})$, which is equal to the distance of the $(a, \tilde{b})$ entry of B_{12}. Note that the same logic follows for $X_{22} = B_{11}$ and $X_{21} = B_{12}$ respectively. Therefore the first 2×2 block of D_{n+1} is D_n, because the zeros that are generated by Gray code construction for the first 2^n entries have no effect on the Hamming distance. Also by induction the bottom right 2×2 matrix is also equal to D_n.

Now consider any $(\grave{a}, \grave{b})$ entry of X_{13}. It must be shown that the entry will have the same distance as the (a, b) entry of $B_{11} + 2J_{n-1}$. Note that $\grave{b} = (11||b_1 b_2 \ldots b_{n-1})$, for X_{13}. So compute $H(\grave{a}, \grave{b}) = H(00||a_1 \ldots a_{n-1}, 11||b_1 \ldots b_{n-1}) = H(00||a, 11||b) = 2 + H(a, b)$. Thus it is the distance of the (a, b) entry of $B_{11} + 2J_{n-1}$. So the sub-matrix $X_{13} = 2J_{n-1} + B_{11}$, with J_{n-1} as defined in the theorem. Also, for any $(\grave{a}, \grave{b})$ entry of X_{24}, the distance will be $H(\grave{a}, \grave{b}) = H((01||a_{n-1} \ldots a_1), (10||b_{n-1} \ldots b_1)) = H(01, 10) + H(a, b)$ which is the distance of the (a, b) entry of $B_{11} + 2J_{n-1}$ as well.

Next it must be shown that $X_{14} = B_{12}$. Choose any $(\acute{a}, \acute{b})$ entry of the X_{14} matrix. Compute the entry by computing the distance $H(\acute{a}, \acute{b}) = H(00||a_1 \ldots a_{n-1}, 10||b_{n-1} \ldots b_1) = H(00||a_1 \ldots a_{n-1}, 10||b_{n-1} \ldots b_1) = H(0||a, 1||\tilde{b})$ which is the $(a, \tilde{b})$ entry of B_{12}. Also choose any $(\grave{a}, \grave{b})$ entry of X_{23}. The distance will be $H(01||a_{n-1} \ldots a_1, 11||b_1 \ldots b_{n-1}) = H(1||\tilde{a}, 0||b)$ which is equal to the entry $(\tilde{a}, b)$ of B_{12}. By induction the top right 2×2 sub matrix will equal the bottom left 2×2 matrix, since D_n is persymmetric.

$$D_{n+1} = \begin{array}{r} \\ 00(a_1 \cdots a_{n-2}) \\ 01(a_{n-2} \cdots a_1) \\ 11(a_1 \cdots a_{n-2}) \\ 10(a_{n-2} \cdots a_1) \end{array} \begin{pmatrix} 00(b_1 \cdots b_{n-2}) & 01(b_{n-2} \cdots b_1) & 11(b_1 \cdots b_{n-2}) & 10(b_{n-2} \cdots b_1) \\ B_{11} & B_{12} & 2J_n + B_{11} & B_{12} \\ B_{12} & B_{11} & B_{12} & 2J_n + B_{11} \\ 2J_n + B_{11} & B_{12} & B_{11} & B_{12} \\ B_{12} & 2J_n + B_{11} & B_{12} & B_{11} \end{pmatrix}$$

Define, $B_{11}^{n+1} = D_n = \begin{bmatrix} B_{11}^n & B_{12}^n \\ B_{12}^n & B_{11}^n \end{bmatrix}$ and also $B_{12}^{n+1} = \begin{bmatrix} 2J_{n-1} + B_{11}^n & B_{12}^n \\ B_{12}^n & 2J_{n-1} + B_{11}^n \end{bmatrix}$. The definition produces $D_{n+1} = \begin{bmatrix} B_{11}^{n+1} & B_{12}^{n+1} \\ B_{12}^{n+1} & B_{11}^{n+1} \end{bmatrix}$, so by induction, D_{n+1} can be used to build D_{n+2} in the same way D_n was used to construct D_{n+1}. □

Notice if D_{n+1} is written as a 4×4 block matrix as in the Theorem 2.1, then D_n appears centrally embedded as a 2×2 block. Since the amount of divisions is arbitrary, one can divide a $2^n \times 2^n$ by any power of two subdivisions and recreate the same structure.

Repeating this argument one sees that every D_k will be centrally embedded in D_n for $k = 1, \ldots, n-1$. In addition not only is every D_k centrally embedded in D_n, but a recursive D_k structure can be found in D_n for any $k \leq n$. Thus the matrix

$$\begin{pmatrix} 0 & k \\ k & 0 \end{pmatrix}, \qquad k = 1, \ldots, n,$$

can be found depending which value of k one wants.

2.2 Properties of D_n

In this section, some basic properties of D_n are obtained. Some of the results proven are stated in He [2, 3], without a detailed proof. Parts (1) and (2), lead to the conclusion that the

Hamming distance matrices are multiples of *doubly stochastic* matrices, which is item (3). A doubly stochastic matrix is a matrix such that all of the column and row sums are 1. Since the Hamming distance matrices are multiples of doubly stochastic matrices with integer entries, there exists a decomposition of D_n into an integral combination of permutation matrices (permutations of the identity). Assertions (4) and (5) in Theorem 2.2, are consequences of Theorem 2.1.

Theorem 2.2 *Let n be the length of the RNA sequences and G_n be the n-bit Gray code, defined in Section 1. Then*

(1) The genetic code-based matrix C_n is a $2^n \times 2^n$ matrix with RNA bases of length n. Each two neighboring entries of genetic code in both directions differ by exactly one base. Also, each neighboring entry of D_n has a difference of 1.

(2) The Hamming distance-based matrix D_n is a $2^n \times 2^n$ matrix. The common row/column sum of the matrix D_n equals $n2^{n-1}$. The total summation of the entries of the matrix D_n is $n2^{2n-1}$.

(3) D_n is symmetric and persymmetric. Furthermore $\frac{D_n}{n2^{n-1}}$ is doubly stochastic.

(4) D_n contains exactly $n+1$ distinct entries, namely $0, 1, \ldots, n$.

(5) The previous matrix D_{n-1} is embedded inside the matrix D_n.

Proof.

(1) By Gray code construction each binary sequence in G_n differs by one from a neighboring binary string. Fix a row in C_n. The entries will be represented by $\binom{x}{y}$ where $x, y \in G_n$. If the row is fixed, y will stay constant for all the columns. However, x will be allowed to vary but only by one position when moving from one column to another. Thus the nucleotide sequence will only be changed by one bit when moving along a row. The

same is true if the column is fixed. So by the way the nucleotides are defined, and by the definition of Gray code, each neighboring nucleotide only differs by one bit. This also implies that the Hamming distance differs by one when moving across a column or row. This is again because as one Gray code sequence is fixed, the other can differ by one bit from each of its neighbors.

(2) For $n = 1$, a Gray code construction is $G_1 = \{0, 1\}$, and $G_2 = \{00, 01, 11, 10\}$, by definition. Fix the row (or column) to be 1. The Hamming distance compares the Gray code representation of 1 and the Gray code representation of each column, j. So the Hamming distances for that row will be $H(\underbrace{00\cdots0}_{n}, j)$ which is $\underbrace{00\cdots0}_{n} \oplus j = j$. Thus the row sum for row 1 is going to be the summation of 1's in all length n binary sequences which is $n2^{n-1}$. This is because half of the length n binary sequences are 1, and there are 2^n of them. Clearly the total sum of the entries in D_n is $2^n \cdot 2^{n-1} = n2^{2n=1}$.

(3) Since the D_n has common row and column sum, the matrix $\frac{D_n}{n2^{n-1}}$ is doubly stochastic. Also by the recursive structure given in Theorem 2.1, $D_n^t = D_n$, so D_n is symmetric.

(4) Take D_n defined as in Section 1. Without loss of generality, fix the first row of D_n, since the rows/columns are permutations of each other. Take the sequence $(00\cdots0)$, by construction, some entry of D_n will be defined by $H(00\cdots0, 00\cdots0) = 0$. It is also known by construction that $(11\cdots1) \in G_n$, thus an entry in D_n will be defined by $H(00\cdots0, 11\cdots1) = n$, because there are n ones. This is clearly the largest Hamming distance when computed with a string of all zeros. Since it is the largest Hamming distance for one row it must be the largest for all rows. Also it was shown previously that each neighboring entry must vary by a Hamming distance of 1, so each integer $i : 0 < i < n$ must be represented in the first row of D_n.

(5) Follows from Theorem 2.1.

□

Chapter 3

The Eigenstructure of D_n

3.1 Preliminary Linear Algebra

Since a focus of this study is to store information in the most efficient way, we will study the eigenstructure. To study D_n, we use some preliminaries from basic linear algebra.

Here are some well known facts about real matrices [4]:

(a) Every real $n \times n$ symmetric matrix is diagonalizable by an orthogonal matrix, i.e. there is an orthonormal basis of eigenvectors.

(b) The rank of a matrix is the number of nonzero eigenvalues.

(c) Let $A_1, \ldots A_n$ be square matrices. Their direct sum is

$$\oplus \sum_{i=1}^{k} A_i = A_1 \oplus \ldots \oplus A_n = \begin{pmatrix} A_1 & & 0 \\ & \ddots & \\ 0 & & A_n \end{pmatrix}$$

and $\text{rank}(\oplus \sum_{i=1}^{k} A_k) = \sum_{i=1}^{k} \text{rank}(A_i)$

3.2 Eigenvectors and Eigenvalues of D_n

First we determine the eigenvalues of D_n:

Theorem 3.1 *The matrix $D_n \in M_{2^n}$ has $n+1$ nonzero eigenvalues equal to*

$$n2^{n-1}, \overbrace{-2^{n-1}, -2^{n-1}, \ldots, -2^{n-1}}^{n}.$$

Proof. We will prove the theorem by induction. The result for $n = 1$ is clear. So assume the result is true for D_n. Note that D_n has two eigenvectors of the form

$$\begin{aligned} x &= a[1, 1, \ldots.., 1]^t \\ y &= a[\underbrace{1, \ldots, 1}_{2^{n-1}}, \underbrace{-1, \ldots, -1}_{2^{n-1}}]^t \end{aligned}$$

with $a = 2^{-n/2}$ for the eigenvalues $n2^{n-1}$ and -2^{n-1}. So, by induction assumption, there is an orthogonal matrix P, with x and y as the first two columns such that

$$A_n = P^t D_n P = [n2^{n-1}] \oplus (-2^{n-1})I_n \oplus 0_{2^n-n-1}.$$

Now let $Q = P \oplus P$. Then

$$\begin{aligned} Q^t D_{n+1} Q &= Q^t \begin{pmatrix} D_n & D_n \\ D_n & D_n \end{pmatrix} Q + Q^t \begin{pmatrix} 0 & 0 & 2J_{n-1} & 0 \\ 0 & 0 & 0 & 2J_{n-1} \\ 2J_{n-1} & 0 & 0 & 0 \\ 0 & 2J_{n-1} & 0 & 0 \end{pmatrix} Q \\ &= \begin{pmatrix} A_n & A_n \\ A_n & A_n \end{pmatrix} + \begin{pmatrix} 0 & C_n \\ C_n & 0 \end{pmatrix}, \end{aligned}$$

where $C_n = \mathrm{diag}\,(2^n, 2^n, 0, \ldots, 0)$.

Up to a permutation similarity, $Q^t D_{n+1} Q$ is a direct sum: $R_1 \oplus R_2 \oplus R_3 \oplus 0_{2^{n+1}-2n-2}$, where

$$R_1 = 2^{n-1} \begin{pmatrix} n & n+2 \\ n+2 & n \end{pmatrix}, \quad R_2 = 2^{n-1} \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix}$$

and R_3 is a direct sum of $(n-1)$ copies of the matrix

$$-2^{n-1} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}.$$

Notice $R_1 \oplus R_2$ has eigenvalues $(n+1)2^n, -2^n, -2^n, 0$, and all the $n-1$ nonzero eigenvalues of R_3 are equal to -2^n. By an inductive argument, the assertion follows.

□

Next we obtain an orthonormal set of eigenvectors of D_n which correspond to the nonzero eigenvalues.

Theorem 3.2 *An orthonormal set of eigenvectors of D_n corresponding to the nonzero eigenvalues $n2^{n-1}, -2^{n-1}, \ldots, -2^{n-1}$ can be constructed as follows. For D_1, the orthonormal eigenvectors are $\frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 1 \end{pmatrix}$ and $\frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ -1 \end{pmatrix}$. Suppose $v_0, v_1, \ldots, v_n$ is constructed for D_n. Then*

$$\tilde{v}_j = \frac{1}{\sqrt{2}}\begin{pmatrix} v_j \\ v_j \end{pmatrix} \text{ for } j = 0, \ldots, n \quad \text{and } \tilde{v}_{n+1} = \frac{1}{\sqrt{2}}\begin{pmatrix} v_0 \\ -v_0 \end{pmatrix},$$

form an orthonormal set of eigenvectors of D_{n+1} corresponding to the nonzero eigenvalues.

Proof. The results can be verified for $n = 1, 2$. Suppose $n > 2$, and the result is true for D_m with $m \leq n$. Clearly, $D_{n+1}\tilde{v}_0 = (n+1)2^n\tilde{v}_0$, since in Chapter 2 it was shown that $(n+1)2^n$ is the common row sum of D_{n+1}.

Let $J_{n-1} \in M_{2^{n-1}}$ be the matrix with all entries equal to one, and let $K_n = J_{n-1} \oplus J_{n-1} \in M_{2^n}$. By induction assumption, $v_0, \ldots, v_n$ form an orthonormal set of eigenvectors for D_n. It can be seen that $K_n v_j = 0$ for all $j = 1, \ldots, n$. Thus,

$$D_{n+1}\tilde{v}_j = \frac{1}{\sqrt{2}}\begin{pmatrix} 2D_n v_j \\ 2D_n v_j \end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix} 2 \cdot -2^{n-1} v_j \\ 2 \cdot -2^{n-1} v_j \end{pmatrix} = -2^n \tilde{v}_j \qquad j = 1, \ldots, n.$$

Moreover,

$$D_{n+1}\tilde{v}_{n+1} = \frac{1}{\sqrt{2}}\left[\begin{pmatrix} (D_n - D_n)v_0 \\ -(D_n - D_n)v_0 \end{pmatrix} + \begin{pmatrix} -2J_{n-1}v_0 \\ 2J_{n-1}v_0 \end{pmatrix}\right] = \frac{1}{\sqrt{2}}\begin{pmatrix} -2 \cdot 2^{n-1} v_0 \\ 2 \cdot 2^{n-1} v_0 \end{pmatrix} = -2^n \tilde{v}_{n+1}.$$

By construction, $\langle \tilde{v}_j, \tilde{v}_j \rangle = 1$ for $j = 0, \ldots, n+1$, and since $\langle v_j, v_k \rangle = 0$ for any $j \neq k$, $\tilde{v}_0, \ldots, \tilde{v}_{n+1}$ are orthoginal. By the principle of induction, the assertion is true. □

By Theorem 3.1 and Theorem 3.2,

$$D_n = n2^{n-1}v_0v_0^t - 2^{n-1}(v_1v_1^t + \cdots + v_nv_n^t).$$

This result provides a far more efficient way to generate D_n. D_n can be generated by the $n+1$ eigenvectors, meaning that now only $n+1$ vectors of size 2^n have to be stored. In Section 2.1, D_n was generated recursively by D_{n-1}, meaning that to generate D_n, 2^{n-1} vectors of size 2^{n-1} had to be stored.

Next we study the powers of D_n.

Theorem 3.3 *Let k be a positive integer. Then*

$$D_n^k = \alpha(n,k)v_0v_0^t + \beta(n,k)D_n,$$

where

$$\alpha(n,k) = (2^{n-1})^k(n^k + (-1)^k n) \quad \textit{and} \quad \beta(n,k) = (-2^{n-1})^{k-1}.$$

Proof. By Theorem 3.1 and Theorem 3.2,

$$D_n = n2^{n-1}v_0v_0^t - 2^{n-1}(v_1v_1^t + \cdots + v_nv_n^t).$$

Define $L_n = v_1v_1^t + \cdots + v_nv_n^t$ then $D_n = n2^{n-1}v_0v_0^t - 2^{n-1}L_n$. So $2^{n-1}L_n = n2^{n-1}v_0v_0^t - D_n$, therefore $L_n = nv_0v_0^t - \frac{D_n}{2^{n-1}}$. Recall,

$$D_n^k = (n2^{n-1})^k v_0v_0^t + (-2^{n-1})^k L_n.$$

Making the substitution for L_n, yields

$$D_n^k = (n2^{n-1})^k v_0v_0^t + (-2^{n-1})^k[nv_0v_0^t - \frac{D_n}{2^{n-1}}].$$

Regrouping the terms

$$D_n^k = [(n2^{n-1})^k + (-2^{n-1})^k n]v_0v_0^t + (2^{n-1})^{k-1}D_n$$

$$= 2^{k(n-1)}[n^k + (-1)^k n]v_0 v_0^t + (-2^{n-1})^{k-1} D_n.$$

Taking

$$\alpha(n,k) = (2^{n-1})^k (n^k + (-1)^k n)$$

and

$$\beta(n,k) = (-2^{n-1})^{k-1},$$

$$D_n^k = \alpha(n,k) v_0 v_0^t + \beta(n,k) D_n.$$

□

As a direct result of Theorem 3.3, no matter what power k, D_n^k will only have as many distinct values as D_n.

Corollary 3.4 *For every positive integer k, D_n^k has $n+1$ distinct values.*

Chapter 4

Decomposition of D_n and Hypercube Structure of C_n

4.1 Decomposition of D_n

Since D_n is a multiple of doubly stochastic matrix with integer entries, it can be decomposed into an integral combination of permutation matrices. We will show that the sum involves only 2^n permutation matrices, which can be defined recursively. The decomposition for $n = 3$ was first recognized by He et al [2, 3]. We have the following general result.

Theorem 4.1 *Let D_n be the Hamming distance matrix defined in Section 1. Then $D_n = \sum_{i=1}^{2^n} a_i^n P_i^n$, where $a = (a_1^n, a_2^n, \ldots, a_{2^n}^n)$ with $a_i \in \{0, 1, \ldots, n\}$, and P_i^n are permutation matrices determined as follows:*
For $n = 1$,

$$a = (a_1^1, a_2^1) = (0, 1) \quad \text{and} \quad P_1^1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \text{and} \quad P_2^1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

For $n \geq 1$

$$P_j^{n+1} = \begin{pmatrix} P_j^n & 0 \\ 0 & P_j^n \end{pmatrix} \quad \text{and} \quad P_{j+2^n}^{n+1} = \begin{pmatrix} 0 & P_j^n \\ P_j^n & 0 \end{pmatrix}$$

and

$$a = (a_1^{n+1}, a_2^{n+1}, \ldots, a_{2^{n+1}}^{n+1}) = (a_1^n, \ldots, a_{2^n}^n, a_1^n, \ldots, a_{2^n}^n) + (\underbrace{0, \ldots, 0}_{2^n}, \underbrace{2, \ldots, 2}_{2^{n-1}}, \underbrace{0, \ldots, 0}_{2^{n-1}}).$$

Moreover, $P_1^n + \ldots + P_{2^n-1}^n = J_{n-1}$*, and each* P_i^n *is symmetric and per symmetric.*

Proof. We prove the result by induction on n, including the additional property that $P_1^n + \ldots + P_{2^n-1}^n = J_{n-1}$ and P_i^n is symmetric and persymmetric. Take

$$D_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = 0 \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + 1 \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},$$

and

$$D_2 = \begin{pmatrix} 0 & 1 & 2 & 1 \\ 1 & 0 & 1 & 2 \\ 2 & 1 & 0 & 1 \\ 1 & 2 & 1 & 0 \end{pmatrix}$$

$$= 0 \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} + 1 \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} + 2 \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} + 1 \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}.$$

So assume that the scheme is true for n, it will be shown that this is true for $n+1$. Assume $D_n = \sum_{i=1}^{2^n} a_i^n P_i^n$ then it will be shown that $D_{n+1} = \sum_{j=1}^{2^n} a_j^{n+1} P_j^{n+1} + a_{j+2^n}^{n+1} P_{j+2^n}^{n+1}$, where a_i^{n+1} and P_i^{n+1} are described as before.

Clearly D_{n+1} will yield 2^{n+1} distinct permutation matrices, by definition. Now it will be shown that with the coefficients it will sum to all of D_{n+1}. As proven in Section 2.1 if

$$D_n = \begin{pmatrix} B_1 & B_2 \\ B_2 & B_1 \end{pmatrix}$$

then

$$\begin{aligned} D_{n+1} &= \begin{pmatrix} B_1 & B_2 & 2J_{n-1} + B_1 & B_2 \\ B_2 & B_1 & B_2 & 2J_{n-1} + B_1 \\ 2J_{n-1} + B_1 & B_2 & B_1 & B_2 \\ B_2 & 2J_{n-1} + B_1 & B_2 & B_1 \end{pmatrix} \\ &= \begin{pmatrix} B_1 & B_2 & B_1 & B_2 \\ B_2 & B_1 & B_2 & B_1 \\ B_1 & B_2 & B_1 & B_2 \\ B_2 & B_1 & B_2 & B_1 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 2J_{n-1} & 0 \\ 0 & 0 & 0 & 2J_{n-1} \\ 2J_{n-1} & 0 & 0 & 0 \\ 0 & 2J_{n-1} & 0 & 0 \end{pmatrix}. \end{aligned}$$

Let the first matrix of D_{n+1} represent how the permutation matrices will change from D_n to D_{n+1}, and the second matrix will represent how the coefficients will change. Recall that $a = (a_1^n, \ldots, a_{2^n}^n, a_1^n, \ldots, a_{2^n}^n) + (\underbrace{0, \ldots, 0}_{2^n}, \underbrace{2, \ldots, 2}_{2^{n-1}}, \underbrace{0, \ldots, 0}_{2^{n-1}})$. When P_i^{n+1} is multiplied by the corresponding a_i^{n+1}, the first 2^n coefficients will remain unaltered which by induction generates

$$\sum_{j=1}^{2^n} a_j^{n+1} \begin{pmatrix} P_j^n & 0 \\ 0 & P_j^n \end{pmatrix} = \begin{pmatrix} B_1 & B_2 & 0 & 0 \\ B_2 & B_1 & 0 & 0 \\ 0 & 0 & B_1 & B_2 \\ 0 & 0 & B_2 & B_1 \end{pmatrix}.$$

Also, the last 2^{n-1} coefficients will also remain unaltered. However, by this scheme the $\{2^n + 1 \quad \text{to} \quad 2^n + 2^{n-1}\}$ permutation matrices are the first 2^{n-1} permutation matrices, but the coefficients have a two added to them. This will generate all the entries of B_1, except they will increase by two. So,

$$\sum_{j=1}^{2^n} a_{j+2^n}^{n+1} \begin{pmatrix} 0 & P_j^n \\ P_j^n & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 2J_{n-1} + B_1 & B_2 \\ 0 & 0 & B_2 & 2J_{n-1} + B_1 \\ 2J_{n-1} + B_1 & B_2 & 0 & 0 \\ B_2 & 2J_{n-1} + B_1 & 0 & 0 \end{pmatrix}.$$

Which together generates all of D_{n+1}.

The first 2^n and the last 2^{n-1} coefficients of D_{n+1} are unaltered. So, $a_j^{n+1} = a_j^n$ and $a_{j'+2^n}^{n+1} = a_{j'}^n$, where $j = 1, 2, \ldots, 2^n$ and $j' = 2^{n-1} + 1, 2^{n-1} + 2, \ldots, 2^n$. Here the fact that $\sum_{j=1}^{2^{n-1}} P_j^n = J_{n-1}$ is used, so the $\{2^n + 1, 2^n + 2, \ldots 2^n + 2^{n-1}\}$ are the entries of B_1 with a 2 added to every entry. Thus the coefficients will have the same imposition. Therefore $a_{\tilde{j}+2^n}^{n+1} = a_{\tilde{j}}^n + 2$, where $\tilde{j} = 1, 2, \ldots 2^{n-1}$. $\square$

4.2 The Graph and Hamilton Circuits of C_n

Consider Gray code represented in a graph $G = (V, E)$, where each vertex is an n-bit binary sequence, and there is an edge between two vertices if the binary sequences differ by one position. Clearly for G_n there will be 2^n vertices in this graph, and each vertex will have degree n. It is known that for every Gray code G_n, there is a Hamilton circuit corresponding to the entries [7].

Let $G_n^* = (V_n^*, E_n^*)$ be the graph such that V_n consists of the entries of C_n, and two entries are adjacent if they differ in one position (as genentic sequences).

In [2, 3], He showed that the matrices P_i^3 in Theorem 4.1 corresponds to certain hypercube structures in $\mathbf{R}^3$. For example, when $n = 2$ the permutation matrices P_i^2 correspond to the entries in C_2 as follows.

$$P_1^2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \sim \begin{pmatrix} CC & 0 & 0 & 0 \\ 0 & CG & 0 & 0 \\ 0 & 0 & GG & 0 \\ 0 & 0 & 0 & GC \end{pmatrix}$$

$$P_2^2 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \sim \begin{pmatrix} 0 & CU & 0 & 0 \\ CA & 0 & 0 & 0 \\ 0 & 0 & 0 & GA \\ 0 & 0 & GU & 0 \end{pmatrix}$$

$$P_3^2 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \sim \begin{pmatrix} 0 & 0 & UU & 0 \\ 0 & 0 & 0 & UA \\ AA & 0 & 0 & 0 \\ 0 & AU & 0 & 0 \end{pmatrix}$$

$$P_4^2 = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \sim \begin{pmatrix} 0 & 0 & 0 & Uc \\ 0 & 0 & UG & 0 \\ 0 & AG & 0 & 0 \\ AC & 0 & 0 & 0 \end{pmatrix}$$

Then each four entries of C_2 correspond to P_i^2 form a cycle in G_2, and can be arranged as the vertex of a graph in $\mathbf{R}^2$ as depicted in Figure 4.1

Note that the four circuits $CC - CG - GG - GC - CC$, $CU - CA - GA - GU - CU$, $UU - UA - AA - AU - UU$, and $UC - UG - AG - AC - UC$ correspond to the matrices

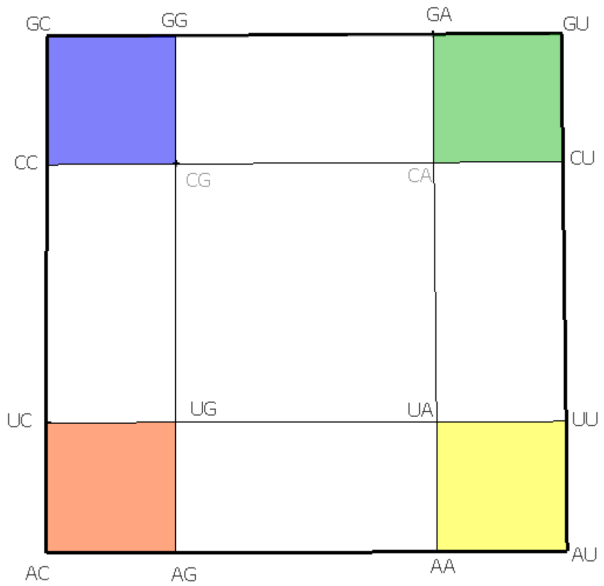

Figure 4.1: Graph Corresponding to C_2

$P_1^1, P_2^2, P_3^2, P_4^2$, respectively. If the first edge in every circuit is deleted, an edge can be drawn between $CG - CA$, $CU - UU$, $UA - UG$, and $UC - CC$. For $n = 2$ a Hamilton circuit of G_2 can be constructed as follows:

$$CC{-}GC{-}GG{-}CG{-}CA{-}GA{-}GU{-}CU{-}UU{-}AU{-}AA{-}UA{-}UG{-}AG{-}AC{-}UC{-}CC.$$

The pattern to observe is that the first circuit starts by going backwards, and the next circuit runs forwards. This pattern repeats itself until the Hamilton circuit is completed.

For $n = 3$:

$$P_1^3 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \sim \begin{pmatrix} CCC & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & CCG & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & CGG & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & CGC & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & GGC & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & GGG & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & GCG & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & GCC \end{pmatrix}$$

$$P_2^3 = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \sim \begin{pmatrix} 0 & CCU & 0 & 0 & 0 & 0 & 0 & 0 \\ CCA & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & CGA & 0 & 0 & 0 & 0 \\ 0 & 0 & CGU & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & GGU & 0 & 0 \\ 0 & 0 & 0 & 0 & GGA & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & GCA \\ 0 & 0 & 0 & 0 & 0 & 0 & GCU & 0 \end{pmatrix}$$

$$P_3^3 = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \sim \begin{pmatrix} 0 & 0 & CUU & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & CUA & 0 & 0 & 0 & 0 \\ CAA & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & CAU & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & GAU & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & GAA \\ 0 & 0 & 0 & 0 & GUA & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & GUU & 0 & 0 \end{pmatrix}$$

$$P_4^3 = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \sim \begin{pmatrix} 0 & 0 & 0 & CUC & 0 & 0 & 0 & 0 \\ 0 & 0 & CUG & 0 & 0 & 0 & 0 & 0 \\ 0 & CAG & 0 & 0 & 0 & 0 & 0 & 0 \\ CAC & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & GAC \\ 0 & 0 & 0 & 0 & 0 & 0 & GAG & 0 \\ 0 & 0 & 0 & 0 & 0 & GUG & 0 & 0 \\ 0 & 0 & 0 & 0 & GUC & 0 & 0 & 0 \end{pmatrix}$$

$$P_5^3 = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix} \sim \begin{pmatrix} 0 & 0 & 0 & 0 & UUC & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & UUG & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & UAG & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & UAC \\ AAC & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & AAG & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & AUG & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & AUC & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$P_6^3 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \sim \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & UUU & 0 & 0 \\ 0 & 0 & 0 & 0 & UUA & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & UAA \\ 0 & 0 & 0 & 0 & 0 & 0 & UAU & 0 \\ 0 & AAU & 0 & 0 & 0 & 0 & 0 & 0 \\ AAA & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & AUA & 0 & 0 & 0 & 0 \\ 0 & 0 & AUU & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$P_7^3 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \sim \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & UCU & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & UCA \\ 0 & 0 & 0 & 0 & UGA & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & UGU & 0 & 0 \\ 0 & 0 & AGU & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & AGA & 0 & 0 & 0 & 0 \\ ACA & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & ACU & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$P_8^3 = \begin{pmatrix} 0&0&0&0&0&0&0&1\\ 0&0&0&0&0&0&1&0\\ 0&0&0&0&0&1&0&0\\ 0&0&0&0&1&0&0&0\\ 0&0&0&1&0&0&0&0\\ 0&0&1&0&0&0&0&0\\ 0&1&0&0&0&0&0&0\\ 1&0&0&0&0&0&0&0 \end{pmatrix} \sim \begin{pmatrix} 0&0&0&0&0&0&0&UCC\\ 0&0&0&0&0&0&UCG&0\\ 0&0&0&0&0&UGG&0&0\\ 0&0&0&0&UGC&0&0&0\\ 0&0&0&AGC&0&0&0&0\\ 0&0&AGG&0&0&0&0&0\\ 0&ACG&0&0&0&0&0&0\\ ACC&0&0&0&0&0&0&0 \end{pmatrix}$$

A similar construction can be done for $n = 3$, where the edges between the first two entries of each permutation matrix are deleted, and then an edge is drawn between $CCG - CCA$, $CCU - CUU$, repeating the pattern for every other permutation matrix ending with $UCC - CCC$. For $n = 3$, this will create a Hamilton circuit for G_3^* in a similar way to $n = 2$. So the subgraph of the nucleotides corresponding to the nonzero entries of each permutation matrix P_i^n is a circuit in G_n^*. Also, every permutation matrix is connected to two other permutations at 2^{n-1} positions. So there exists a circuit in the graph where all the permutation matrices are considered vertices of a hypercube. We will show that one can combining the circuits correspond of P_i^n to get a Hamilton circuit in G_n^* for general n in the following.

We begin with the following lemma, which follows from symmetry and per symmetry of P_i^n. We give a different proof.

Lemma 4.2 *Assume P_i^n is the permutation matrix as defined in Theorem 4.1. If P_i^n has a nonzero entry at position $(1, q_1)$, then the $(2^n, 2^n - q_1 + 1)$ and the $(2^{n-1}+1, 2^{n-1} - q_{2^{n-1}} + 1)$ entry of P_i^n will also be nonzero.*

Proof. By the decomposition given for $n = 2, 3$, this assertion holds. It must be shown for $n \geq 4$. So assume the assertion is true for n, it will be shown for $n+1$. As proven in Theorem 4.1, $P_i^{n+1} = \begin{pmatrix} P_i^n & 0 \\ 0 & P_i^n \end{pmatrix}$ and $P_{i+2^n}^{n+1} = \begin{pmatrix} 0 & P_i^n \\ P_i^n & 0 \end{pmatrix}$ for $1 \leq i \leq 2^n$. Let $(1, q_1)$ denote the position of the row-one-nonzero entry of P_i^n. By induction, since the $(2^n, 2^n - q_1 + 1)$ entry of P_i^n is nonzero, so is the $(2^n, 2^n - q_1 + 1)$ entry of P_i^{n+1}. This is because the first $2^n \times 2^n$

block of P_i^n is the same as P_i^{n+1}. By the construction of P_i^{n+1}, $\forall(r,s) : 1 \leq r, s \leq 2^n$, if (r,s) is a non-zero entry of P_i^n, then $(r+2^n, s+2^n)$ is a nonzero entry of P_n^{n+1}. By the induction hypothesis, P_i^{n+1} has a nonzero entry at $(2^n, 2^n - q_1 + 1) + (2^n, 2^n) = (2^{n+1}, 2^{n+1} - q_1 + 1)$.

Also, if P_i^n has a nonzero entry at $(1, q_1)$, by construction, then $P_{i+2^n}^{n+1}$ will have an nonzero entry at $(1, q_1^*)$, where $q_1^* = 2^n + q_1$. By induction P_i^n has a nonzero entry at $(2^n, 2^n - q_1 + 1)$ so, by construction the $(2^{n+1}, 2^n - q_1 + 1)$ position of $P_{i+2^n}^{n+1}$, will be nonzero. This is because for $P_{i+2^n}^{n+1}$, the first 2^n rows have nonzero entries precisely where P_i^n has nonzero entries, however the columns are shifted by 2^n. The last 2^n rows have nonzero entries in the same columns as P_i^n. But notice that $2^n - q_1 + 1 = 2^n - (q_1^* - 2^n) + 1 = 2^{n+1} - q_1^* + 1$. Thus if either P_i^{n+1} or $P_{i+2^n}^{n+1}$ have a nonzero entry at position $(1, q)$, then also the $2^{n+1} - q + 1$ entry is nonzero. The first part of the assertion holds by the induction hypothesis.

If the $(1, q_1)$ entry of P_i^n is nonzero then the $(2^n, 2^n - q_1 + 1)$ entry is nonzero as well. Clearly by the decomposition given for $n = 2, 3$ the assertion holds, so assume the assertion is true for n. It will be shown that the assertion is true for $n+1$. Since the $(2^n, 2^n - q_1 + 1)$ of P_i^n is nonzero then by construction so is the $(2^n, 2^n - q_1 + 1)$ entry of P_i^{n+1}, because the first $2^n \times 2^n$ block of P_i^{n+1} is P_i^n. Since P_i^n is copied onto the lower right $2^n \times 2^n$ block of P_i^{n+1}, the $(2^n + 1, 2^n + q_1)$ entry is also nonzero. But by the previous part of this proof, $q_1 = 2^n - q_{2^n} + 1$ so $(2^n + 1, 2^n + q_1) = (2^n + 1, 2^{n+1} - q_{2^n} + 1)$.

Furthermore, by construction, if (s,t) is a nonzero entry of P_i^n, then $(s,t) + (0, +2^n)$ entry of $P_{i+2^n}^{n+1}$ is nonzero. So, since the $(2^n, 2^n - q_1 + 1)$ entry of P_i^n is nonzero, the $(2^n, 2^n - q_1 + 1) + (0, 2^n) = (2^n, 2^{n+1} - q_1 + 1) = (2^n, q_{2^n})$ entry of $P_{i+2^n}^{n+1}$ is nonzero. By construction the $(2^n + 1, q_1)$ entry of $P_{i+2^n}^{n+1}$ is nonzero but $q_1 = 2^{n+1} - q_{2^n} + 1$. Therefore the $(2^n + 1, q_1) = 2^n + 1, 2^{n+1} - q_{2^n} + 1)$. Thus by induction the assertion holds. □

Theorem 4.3 *Let P_i^n be defined as in Theorem 4.1. Then there is a circuit of length 2^n in G_n^* connecting the entries in C_n corresponding to the nonzero position of P_i^n.*

Proof. We will prove this by induction. Take the graph G, where the vertices are the nucleotides corresponding to the nonzero entries of P_i^n, and two vertices are adjacent if their Hamming distance is 1. The assertion is clearly true for $n = 2, 3$, so assume that G has a circuit for n. It will be shown that there is a circuit in

$$\begin{pmatrix} P_i^n & 0 \\ 0 & P_i^n \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} 0 & P_i^n \\ P_i^n & 0 \end{pmatrix},$$

when G' is the graph whose vertices are the nonzero entries of P_j^{n+1}. Note that these two matrices are just P_i^{n+1} and $P_{i+2^n}^{n+1}$ respectively, for an integer i.

By induction the nonzero entries of P_i^n have a circuit denoted as $x_1 - x_2 - \cdots - x_{2^n} - x_1$, where the position of x_1 is $(1, q_1)$, x_2 is $(2, q_2), \ldots,$ x_{2_n} is $(2^n, q_{2_n})$. In other words, the circuit is connected in consecutive order according to the rows. By the recursive structure in P_i^n, the nucleotides corresponding to P_i^{n+1} have two disjoint circuits, because P_i^n appears as two sub-matrices of P_i^{n+1}. Let the two circuits of P_i^{n+1} be $x_1 - x_2 - \ldots - x_{2^n} - x_1$ and $y_1 - y_2 - \ldots - y_{2^n} - y_1$, respective to the nucleotide sequences. Note that the circuits corresponding positions in the matrix are $(1, q_1) - (2, q_2) - \cdots - (2^n, q_{2^n}) - (1, q_1)$ and $(2^n + 1, r_1) - (2^n + 2, r_2) - \cdots - (2^{n+1}, r_{2_n}) - (2^n + 1, r_1)$, respectively.

But by Lemma 4.2, $r_1 = 2^{n+1} - q_{2^n} + 1$, so r_1 and q_{2_n} are equidistant from the vertical center because $r_1 + q_1 = 2^{n+1} + 1$. Thus since $G_n = \{0||a_0, 0||a_1, \ldots, 0||a_{n-1}, 1||a_{n-1}, 1||a_{n-2}, \ldots, 1||a_0\}$, and since x_{2^n} and y_1 are equidistant from the center the only change made will be to the first bit. By Gray code construction, a pair of Gray code sequences equidistant from the center only differ in one bit. Every change made as the Gray codes move closer to the center will be reversed as the Gray code goes away from the center, except for the first bit. The first bit will change from a 0 to a 1 or vise versa. By Lemma 4.2, if P_i^{n+1} has a nonzero entry at $(1, q_1)$, then it also has a nonzero entry at $(2^{n+1}, 2^{n+1} - q_1 + 1)$. So since the position corresponding to y_{2^n} is $(2^{n+1}, r_{2^n})$, and $r_{2^n} = 2^{n+1} - q_1 + 1$, x_1 and y_{2^n} are also equidistant from the center. Therefore x_{2^n} and y_1 are adjacent, and y_{2^n} and x_1 are adjacent.

So, delete the edges (x_{2^n}, x_1) and (y_{2^n}, y_1), and then connect (x_{2^n}, y_1) and (y_{2^n}, x_1); that will be a circuit for the graph G'. Furthermore, it should be noted that since the change only occurs in the first bit, and it occurs as the Gray code changes horizontally and vertically with respect to the matrix, if the first bit of the nucleotide string is C, then it will change to G, and if the first bit of the nucleotide string is U then it will go to A, and vice versa. □

The next three theorems will provide a way to construct a Hamilton cycle in G_n^*. More specifically, the Hamilton cycle will only be constructed by the paths generated from the non-zero entries of the permutation matrices. The proofs will use Lemma 4.2 and Theorem 4.1. The Hamilton cycle gives a pathway for mutations in genetic code.

Theorem 4.4 *Consider the permutation matrices P_j^n and P_{j+1}^n defined in Theorem 4.1. Let $g_1, \ldots, g_{2^n}$ and $\hat{g}_1, \ldots, \hat{g}_{2^n}$, be the genetic sequences defined in Theorem 4.3, corresponding to the nonzero positions of the matrices respectively. Then consecutive sequences in $g_1 - g_{2^n} - \cdots - g_2 - \hat{g}_2 - \hat{g}_3 - \cdots \hat{g}_{2^n} - \hat{g}_1 - g_1$, differ by one nucleotide. In other words the sequence provides a circuit in a graph G_n^* with the vertices being $g_1, g_2, \ldots g_{2^n}$ and $\hat{g}_1, \hat{g}_2, \ldots, \hat{g}_{2^n}$.*

Proof. For every permutation matrix P_i^n, fix the row (or column), to be one. A unique matrix will represent a nonzero entry in each column; precisely the first row of P_i^n will have a nonzero entry in column i. This is true because columns of the nonzero entries are preserved by construction. $P_{i+2^n}^{n+1}$, by construction, has a nonzero entry at the same position as the nonzero entry of P_i^n, but is shifted by 2^n, thus the nonzero entry is $i+2^n$ by induction.

Thus, P_i^n and P_{i+1}^n correspond to nonzero entries in column i and $i+1$ in row 1. Call the genetic sequences corresponding to those positions g_1 and $\hat{g}_1$ respectively. In chapter 2 it was shown that two neighboring nucleotide strings differ by only one position. Therefore draw an edge between the nucleotides corresponding to g_1 and $\hat{g}_1$. Since there is a unique P_j^n, $j = 1, \ldots, 2^n$, that has a nonzero entry for every cell in the first row, there will be a path that visits all P_j^n, $j = 1, \ldots, 2^{n-1}$ exactly once. It is also known that the first entry in the Gray code sequence G_n is $0\underbrace{00\cdots0}_{n-1}$ and by construction the last entry is $1\underbrace{00\cdots0}_{n-1}$. So there is an edge between P_1^n and $P_{2^n}^n$. Thus there is at least one edge between the nucleotides corresponding to the nonzero entries of P_i^n and P_{i+1}^n. Also there is an edge between P_1^n and $P_{2^n}^n$.

Take g_2 and $\hat{g_2}$, to be the nucleotide sequences corresponding to the nonzero entries of P_i^n and P_{i+1}^n in row 2. There is an edge between g_2 and $\hat{g_2}$ by symmetry, since every row is just a permutation of the first row. Thus the first two rows contain a circuit that visits every entry in that row exactly once. □

Remark 4.5 *If each permutation matrix is viewed as vertex of a graph, then there is a Hamilton circuit between all of the permutation matrices, i.e.,* $P_1^n - P_2^n - \cdots P_{2^n}^n - P_1^n$*, where two permutation matrices are adjacent if and only if they contain a nucleotide strings that differ in only one position to the next. This idea is abstract but will be used in the next proof. Furthermore since every row j is just a permutation of row 1, a Hamilton circuit can be found for row j that disjoint from the one found in row 1, by the same process. This is because for any row j a unique permutation matrix will represent a nonzero entry for a specific column. Thus there can be an edge connected between the nucleotides represented by those nonzero entries. Thus there are* 2^n *disjoint ways to connect all of the* P_i^n *because there is a different way to connect the Hamilton cycles for each row.*

Theorem 4.6 *Consider the graph* $G_n^* = (V_n^*, E_n^*)$ *such that the length n genetic sequences are the vertices and two vertices are adjacent if they differ by one position. Then one can combine the circuits corresponding to* $P_1^n, P_2^n, \ldots P_{2^n}^n$*, in Theorem 4.3, to form a Hamilton circuit of the graph.*

Proof. Let G be the graph, where the vertex set is all length n nucleotide sequences, and two vertices are adjacent if they have a Hamming distance of 1. By the Theorem 4.3 and Theorem 4.4 it is known that there is a circuit corresponding to the nonzero entries *within* each permutation matrix and *between* the permutation matrices. Take P_1^n and P_2^n and their corresponding circuits, constructed in the same manner as Theorem 4.3, $x_1^1 - x_2^1 - \ldots - x_{2^n}^1 - x_1^1$

and $x_1^2 - x_2^2 - \ldots - x_{2^n}^2 - x_1^2$, where x_s^i represents the nucleotide sequence corresponding to the nonzero entry of the s^{th} row of P_i^n. Delete the edges (x_1^1, x_2^1) and (x_1^2, x_2^2), then connect the edges (x_1^1, x_1^2) and (x_2^1, x_2^2). This can be done because x_1^1 and x_1^2 are neighboring cells which, as proven in Theorem 2.2, differ by only one nucleotide base; similarly x_2^1 and x_1^2 are also neighboring cells. This will yield a circuit consisting of all nucleotides represented by the nonzero entries of P_1^n and P_2^n. Similarly do the same with the circuits corresponding to P_3^n and P_4^n. Delete the edges (x_1^3, x_2^3) and (x_1^4, x_2^4). Then connect the edges (x_1^3, x_1^4) and (x_2^3, x_2^4), which will yield a circuit between nucleotide sequences represented by the nonzero entries of P_3^n and P_4^n. If this is done with P_i^n and P_{i+1}^n, where i is odd and $0 \leq i \leq 2^n$ there will be 2^{n-1} disjoint circuits since all permutation matrices are disjoint. So $\forall i : 0 \leq i \leq 2^n, i$ is odd, delete the edges (x_1^i, x_2^i) and (x_1^{i+1}, x_2^{i+1}), then connect edges (x_1^i, x_1^{i+1}) and (x_2^i, x_2^{i+1}). Finally for all $0 \leq j \leq 2^n$ and j odd, delete the edge (x_1^j, x_1^{j+1}) and then connect (x_1^{j+1}, x_1^{j+2}), which will result in a Hamilton circuit for all nucleotides of length n. After the first iteration the circuits will be connected as follows:

$$\begin{pmatrix} x_1^1- & x_1^2 & x_1^3- & x_1^4 & \ldots & x_1^{2^n-1}- & x_1^{2^n} \\ x_2^2- & x_2^1 & x_2^4- & x_2^3 & \ldots & x_2^{2^n}- & x_2^{2^n-1} \\ \vdots & \vdots & \vdots & \vdots & \ldots & \vdots & \vdots \end{pmatrix},$$

where a dash after an entry indicates its neighbor (to the right) and it are adjacent. This makes 2^{n-1} disjoint circuits of the graph G containing P_i^n and P_{i+1}^n. Then for the next iteration the Hamilton circuit for G will be completed as follows:

$$\begin{pmatrix} x_1^1 & x_1^2- & x_1^3 & x_1^4- & \ldots & x_1^{2^n-1} & x_1^{2^n}- \\ x_2^2- & x_2^1 & x_2^4- & x_2^3 & \ldots & x_2^{2^n}- & x_2^{2^n-1} \\ \vdots & \vdots & \vdots & \vdots & \ldots & \vdots & \vdots \end{pmatrix}.$$

Note that there is an edge between $x_1^{2^n}$ and x_1^1. This clearly creates a Hamilton circuit which visits all the nucleotides corresponding to the nonzero entries of P_i^n before traveling to a nucleotide represented by the nonzero entries of P_j^n. Also, if the circuit is started at x_1^1, then the circuit will traverse backwards; for example $x_1^1 - x_{2^n} - \ldots$, but when the circuit moves to x_2^2 it will traverse forward. This pattern of alternating between traversing ascending and descending the rows will continue. The Hamilton ircuit will traverse as follows.

$$x_1^1 - x_{2^n}^1 - \ldots - x_2^1 - x_2^2 - x_3^2 \ldots - x_1^2 - x_1^3 - x_{2^n}^3 - \ldots - x_2^{2^n} - x_3^{2^n} - \ldots x_1^{2^n} - x_1^1.$$

□

Chapter 5

The Genetic Code Matrix C_n

5.1 An Introduction into Genetic Code

James Watson and Francis Crick solved one of the many quandaries in the scientific world when they "cracked" the genetic code in 1953. It was then necessary for other researchers to study how genetic code was translated into amino acids. It was known that there are 20 different amino acids (plus start and stop codons), and since there are four nucleotide bases, $\{A, U, C, G\}$, there are 4^n different combinations of bases, for a string of length n. Therefore, $n = 3$ is the smallest number of bases that could be used to represent the 20 different codons. There is either degeneracy between the codons or some just do not occur in nature. There happens to be degeneracy between the codons, meaning they represent the same amino acid; however, there is no ambiguity, so two different amino acids cannot be represented by the same codon.

As presented in Chapter 1, C_n is the genetic code matrix with each cell represented by n-distinct nucleotides. Along with the recursive structure in D_n, it can be shown that there is a recursive way to generate C_n. It will be shown that given C_n, C_{n+1} can be generated. Also, there is a MatLab program that can be found in Appendix 1, which will generate C_n.

5.2 Constructing Genetic Code Recursively

Theorem 5.1 *Suppose C_n is the genetic matrix defined in Chapter 1. Then*

$$C_{n+1} = \begin{pmatrix} C||C_n & U||C_nF_n \\ A||F_nC_n & G||F_nC_nF_n \end{pmatrix},$$

where F_n is the anti-diagonal matrix.

Proof. It is known

$$C_1 = \begin{pmatrix} C & U \\ A & G \end{pmatrix} \quad \text{and} \quad C_2 = \begin{pmatrix} CC & CU & UU & UC \\ CA & CG & UG & UA \\ AA & AG & GG & GA \\ AC & AU & GU & GC \end{pmatrix}.$$

Therefore, it must be shown that the formula works for $n \geq 3$. Assume the construction is valid for n, prove that the construction is true for $n+1$. It is known that $G_n = \{0||a_0, 0||a_1, \ldots, 0||a_{n-1}, 1||a_{n-1}, 1||a_{n-1}, \ldots, 1||a_0\}$, where $a_i \in G_{n-1}$. Take $\alpha, \beta, \alpha', \beta' \in G_n$ where $\alpha = b_1b_2\ldots b_n$ and $\alpha' = b'_1b'_2\ldots b'_n$; and $\beta = b_nb_{n-1}\ldots b_1$ and $\beta' = b'_nb'_{n-1}\ldots b'_1$ such that $b_i, b'_i \in \{0,1\}$. It is also known that

$$C_n = \begin{pmatrix} B_1 & B_2 \\ B_3 & B_4 \end{pmatrix}.$$

Where B_1, B_2, B_3, B_4 are defined by a certain $\binom{\alpha'}{\alpha}$.

Take

$$C_{n+1} = \begin{pmatrix} X_1 & X_2 \\ X_3 & X_4 \end{pmatrix}.$$

By definition of Gray code, the (α, α') entry of X_1 is defined by $\binom{0\alpha'}{0\alpha}$. Since $\binom{0}{0} \sim C$, $\binom{0\alpha'}{0\alpha} \sim C||\binom{\alpha'}{\alpha}$, so X_1 can be represented by concatenating C to the (α, α') entry of C_n. Thus $X_1 = C||C_n$

Now take the (α, β') entry of X_2, defined by $\binom{1||\beta'}{0||\alpha} = \binom{1}{0}||\binom{\beta'}{\alpha} \sim U||\binom{\beta'}{\alpha}$, since $U \sim \binom{1}{0}$. But C_nF_n switches the columns $k-j$ and j for $j = 0, 1, \ldots, 2^n$, leaving the rows unchanged. So an (α, α') entry of C_n is defined by $\binom{\beta'}{\alpha}$ in C_nF_n, because an the (α, α') entry of C_n is $\binom{\alpha'}{\alpha} = \binom{b'_1b'_2 \ldots b'_n}{b_1b_2 \ldots b_n}$. So when taking the corresponding (α, α') entry of C_nF_n, the Gray code for the columns will be reversed so it will be equivalent to $\binom{b'_nb'_{n-1} \ldots b'_1}{b_1b_2 \ldots b_n} = \binom{\beta'}{\alpha}$. Therefore $U||C_nF_n = X_2$. The result for X_3 is similar, however F_nC_n switches the rows and leaves the columns unchanged.

Take the corresponding entry of (β, β') in X_4, which is defined by $\binom{1||\beta'}{1||\beta}$. The operation $F_nC_nF_n$ reverses the columns and the rows of C_n. Since $\binom{1}{1} \sim G$ and C_n is defined by $\binom{\alpha'}{\alpha}$, when multiplying $F_nC_nF_n$, both the column and row Gray codes will be reversed. So the $\binom{\alpha'}{\alpha} = \binom{b'_1b'_2 \ldots b'_n}{b_1b_2 \ldots b_n}$, entry of C_n changes to $\binom{b'_nb'_{n-1} \ldots b'_1}{b_nb_{n-1} \ldots b_1} = \binom{\beta'}{\beta}$ which is, by definition, $F_nC_nF_n$. So $X_4 = G||F_nC_nF_n$. $\square$

5.3 Counting Nucleotides

5.3.1 Usefulness

An important aspect of this study is to generate as much information as possible about all length n nucleotide sequences but also to store it in a more manageable fashion. Determining exactly how many of each nucleotide are represented by a cell of C_n would store more information than D_n but still would not be much more computationally expensive. Since it is known that $C \sim \binom{0}{0}, U \sim \binom{1}{0}, A \sim \binom{0}{1}, G \sim \binom{1}{1}$, and by definition of Hamming distance, if a cell in D_n has a Hamming distance of i, there must be, i so many A's and U's represented in the corresponding C_n sequence. This section will show exactly how many times (C, U, A, G) are represented in each C_n cell. S_n will be defined as the matrix in which all the entries are a 4-tuple, (x_C, x_U, x_A, x_G), where x_i is how many times that nucleotide occurs in the corresponding cell of C_n.

5.3.2 Counting the Occurrences of Nucleotides Per Cell

Theorem 5.2 *Define S_n to be a matrix of size $2^n \times 2^n$, where each cell of S_n is represented by a numerical sequence, (x_C, x_U, x_A, x_G), where x_i is the number of times the i^{th} nucleotide is represented in C_n. Then*

$$S_{n+1} = \begin{pmatrix} (1000)J_n + S_n & (0100)J_n + S_nF_n \\ (0010)J_n + F_nS_n & (0001)J_n + F_nS_nF_n \end{pmatrix}$$

where F_n is the anti-diagonal matrix, and J_n is a $2^n \times 2^n$ matrix of all 1's.

Proof. As previously defined

$$C_1 = \begin{pmatrix} C & U \\ A & G \end{pmatrix} \quad \text{so by definition} \quad S_1 = \begin{pmatrix} (1000) & (0100) \\ (0010) & (0001) \end{pmatrix}$$

Then by the construction of S_n

$$S_2 = \begin{pmatrix} (1000)J_1 + S_1 & (0100)J_1 + S_1F_n \\ (0010)J_1 + F_nS_1 & (0001)J_1 + F_nS_1F_n \end{pmatrix}$$

$$= \begin{pmatrix} (1000)J_1 + \begin{pmatrix} (1000) & (0100) \\ (0010) & (0001) \end{pmatrix} & (0100)J_1 + \begin{pmatrix} (0100) & (1000) \\ (0001) & (0010) \end{pmatrix} \\ (0010)J_1 + \begin{pmatrix} (0010) & (0001) \\ (1000) & (0100) \end{pmatrix} & (0001)J_1 + \begin{pmatrix} (0001) & (0010) \\ (0010) & (1000) \end{pmatrix} \end{pmatrix}$$

$$= \begin{pmatrix} (2000) & (1100) & (0200) & (1100) \\ (1010) & (1001) & (0101) & (0110) \\ (0020) & (0011) & (0002) & (0011) \\ (1010) & (0110) & (0101) & (1001) \end{pmatrix}.$$

The definition is correct for $n = 1, 2$, it must be shown that it is true for $n \geq 3$. So assume the construction for n, it must be shown for $n + 1$. We showed in the previous section that:

$$C_{n+1} = \begin{pmatrix} C||C_n & U||C_nF_n \\ A||F_nC_n & G||F_nC_nF_n \end{pmatrix}.$$

This is obviously equivalent to adding the corresponding nucleotide to each sub-matrix. We know, according to the induction hypothesis, how many of each nucleotide are contained in C_n, so clearly adding a nucleotide to C_n is modeled by the definition of S_{n+1}. $\square$.

5.4 Amino Acids

5.4.1 Definitions

There are redundancies in the codons of genetic code, but there is no ambiguity. For example, CCU and CCC both represent Prolic (Pro) acid, but there is no ambiguity so that no codon represents more than one amino acid. There are also start and stop codons. The translation section of genetic code starts with an initiation chain which is called a start codon. Stop codons are identified by the name of a color, and they signal release factors, so there is a mapping that maps the Genetic codons to their amino acids. There are 20 amino acids and 1 start codon, so there is obviously going to be some overlap, which is modeled in this matrix. Note that this is only for $n = 3$ and any multiple of three, since codons are tri-nucleotide sequences. C_n can be mapped from codons to amino acids.

5.4.2 Amino Acid Matrix

For n=3

$$C_3 = \begin{pmatrix} CCC & CCU & CUU & CUC & UUC & UUU & UCU & UCC \\ CCA & CCG & CUG & CUA & UUA & UUG & UCG & UCA \\ CAA & CAG & CGG & CGA & UGA & UGG & UAG & UAA \\ CAC & CAU & CGU & CGC & UGC & UGU & UAU & UAC \\ AAC & AAU & AGU & AGC & GGC & GGU & GAU & GAC \\ AAA & AAG & AGG & AGA & GGA & GGG & GAG & GAA \\ ACA & ACG & AUG & AUA & GUA & GUG & GCG & GCA \\ ACC & ACU & AUU & AUC & GUC & GUU & GCU & GCC \end{pmatrix}$$

But our Amino Acid Matrix (denoted by A_n, where n is a multiple of 3) is as follows

$$A_3 = \begin{pmatrix} Pro & Pro & Leu & Leu & Phe & Phe & Ser & Ser \\ Pro & Pro & Leu & Leu & Leu & Leu & Ser & Ser \\ Gln & Gln & Arg & Arg & OPAL & Trp & AMBER & OCHRE \\ His & His & Arg & Arg & Cys & Cys & Tyr & Tyr \\ Asn & Asn & Ser & Ser & Gly & Gly & Asp & Asp \\ Lys & Lys & Arg & Arg & Gly & Gly & Gly & Gly \\ Thr & Thr & MET(START) & Ile & Val & Val & Ala & Ala \\ Thr & Thr & Ile & Ile & Val & Val & Ala & Ala \end{pmatrix}$$

Note that with A_3, $MET, OPAL, AMBER, \quad \text{and} \quad OCHREA$, are the start and stop codons as mentioned in the previous paragraph.

Chapter 6

Further Research

Through this thesis, there has been a lot information on genetic code and the Hamming distances that are related to nucleotide strings. This information has been presented in a structurally recursive manner that is easy to generate. An important issue that can be addressed is how to apply the recursive schemes to current biological problems.

Some further points of research include, but are not limited to, finding real applications for the recursive structure of the Hamming distance matrix, the Nucleotide Counting Matrix, and/or the Genetic Code Matrix. Also, since the matrices grow exponentially, generating these matrices for large n is classically inefficient. A more efficient way to store the data could also be an avenue of research.

There may be interesting implications of the hypercube structure and Hamilton circuit that could be useful in genetic mutation. Since the two vertices of the hypercube are adjacent if and only if the codons differ in one position, what effect would changing a codon during RNA transcription have on the corresponding amino acid? For example, if one wanted to compute how many mutations it would take for GCU to mutate into CUC, once could examine all of the pertinent Hamilton paths between the two codons.

Lastly, this paper has described a clever way to generate D_n^k, which was a result of the Eigenstructure. However, during the study it was not obvious what implications this actually had. A study on how this corresponds to the overarching problem could prove to be useful.

Appendix A

MatLab Code

The Following are two MatLab Programs that generate D_n and C_n respectively.

For D_n:

```
clear all; close all; clc;

D1 = [0 1; 1 0];

D2 = [0 1 2 1; 1 0 1 2; 2 1 0 1; 1 2 1 0];

for n=3:k                       %k is to which D to generate

    for i=1:2^(n-2)

        for s=1:2^(n-2)

            J(i, s+2^(n-1))=2;

        end

    end

    for i=2^(n-2)+1:2^(n-1)

        for s= 2^(n-2)+2^(n-1)+1:2^n

            J(i,s)=2;

        end

    end

    for i=1:2^(n-2)

        for s=1:2^(n-2)

            J(i+2^(n-1),s)=2;
```

```
        end
    end
    for i=2^(n-2)+1:2^(n-1)
        for s= 2^(n-2)+2^(n-1)+1:2^n
            J(s,i)=2;
        end
    end
D3=[D2 D2; D2 D2]+J;
D2=D3
J=0;
end
```

For C_n:

```
clear all; close all; clc;

syms C U G A CC CU UU UC CA CG UG UA AA AG GG GA AC AU GU GC
C2 = [CC CU UU UC; CA CG UG UA; AA AG GG GA; AC AU GU GC];

for n=3:k                          %k is how large C is
    for i= 1:1:2^(n-1)
        for j=1:1:2^(n-1)
            CC1(j,i) = C*C2(j,i);
        end
    end

    for i= 0:1:2^(n-1)-1
        for j=1:1:2^(n-1)
            CA1(j,2^(n-1)-i) = A*C2(j,i+1);
        end
    end
```

```
    for i= 1:2^(n-1)
        for j=0:2^(n-1)-1
            CU1(2^(n-1)-j, i) = U*C2(j+1,i);
        end
    end

    for i= 0:2^(n-1)-1
        for j=0:2^(n-1)-1
            CG1(2^(n-1)-j,2^(n-1)-i) = G*C2(j+1,i+1);
        end
    end
    C2= [CC1 CU1; CA1 CG1];
    C2
end
```

Appendix B

List of Notation

$\oplus$	Direct Sum
(i, j)	Row i, Column j of a Matrix
G_n	n-bit Gray Code Sequence
D_n	Hamming Distance Matrix
C_n	Genetic Code matrix
F_n	The Anti-Diagonal Matrix
$a\|\|b$	a Concatenate b
P_i^n	The Permutation Matrices for D_n
$H(a, b)$	Hamming Distance of a and b

Bibliography

[1] Freeland S.J., Wu T. and Keulmann N. The Case for an Error Minimizing Genetic Code. Orig Life Evol Biosph. 33(4-5): 457-77.(2003)

[2] He, M. Genetic code, Hamming Distance and Stochastic Matrices, Bull Math. Biology 66:1405-1421.(2004)

[3] He, M. Genetic Code, Attributive Mappings and Stochastic Mtrices, Bull Math. Biology 66:965-973. (2004)

[4] Horn, R. and Johnson, J. Matrix Analysis,Cambridge University Press, New York. (1985)

[5] Jimenéz-Monteno, M., Mora-Basenez, C.R. and Poechel T. The Hypercube Structure of Genetic Code, BioSystems, 39:117125. (1996)

[6] Swanson, R. A Unifying Concept for The Amino Acid Code. Bull. Math. Biology. 46:187-203 (1984).

[7] Tucker, A. Applied Combinatorics, John Wiley & Sons, 5^{th} ed. (2007)