AN ABSTRACT OF THE DISSERTATION OF

Bader Albader for the degree of Doctor of Philosophy in

Computer Science presented on June 1, 2009.

Title: Some Communication Algorithms for Gaussian and Eisenstein-Jacobi Networks

Abstract approved: _____
Bella Bose, Mary Flahive

Interconnection networks play an important role in designing high performance computers. Recently two new classes of interconnection networks based on the concept of Gaussian and Eisenstein-Jacobi integers were introduced. In this research, efficient routing and broadcasting algorithms for these networks are developed. Furthermore, constructing edge disjoint Hamiltonian cycles in Gaussian networks is also investigated. Some resource placement methods for Eisenstein-Jacobi networks are also studied.

Some Communication Algorithms for Gaussian and Eisenstein-Jacobi Networks

by

Bader Albader

A DISSERTATION

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Doctor of Philosophy

Presented June 1, 2009
Commencement June 2009

Doctor of Philosophy dissertation of Bader Albader presented on June 1, 2009

APPROVED:

_____

Major Professor, representing Computer Science

_____

Director of the School of Electrical Engineering and Computer Science

_____

Dean of the Graduate School

I understand that my dissertation will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my dissertation to any reader upon request.

_____

Bader Albader, Author

ACKNOWLEDGMENTS

In the beginning, I would like to thanks Allah for giving me the strength and the power to be able to finish my degree. Without his generosity and mercy, none of this would have been possible.

There are no one to whom I own as much as I do to my wife for her unconditional love and support. Her continuous encouragement made my achievement possible.

I would also like to record my sincere thanks to the invaluable and able help and guidance of my wife's family. Their support played an important role in helping me during my studies.

I would like to express my sincere and deep appreciation to my major professor, Dr. Bose, for his valuable ideas, which put me in the right direction. His wide knowledge and his logical way of thinking have been of great value for me. In addition, despite his busy schedule he was available anytime to help me out. He was also instrumental in showing me how to analyze and work with a research problem.

I would like to express sincere gratitude to Dr. Mary Flahive for her valuable guidance and her constructive comments to improve the quality of my thesis and for serving in my committee.

I would also like to express my sincere gratitude to Dr. Jack F. Higginbotham, Dr. Timothy A. Budd, and Dr. Thinh Nguyen for serving in my committee.

I am thankful to Kuwait University for sponsoring my graduate studies and to Oregon State University for providing me with excellent education and research environments.

I would also like to express my deep gratitude for Dr. Bader Albdaiwi for his guidance and support.

I want to thank all the helpful staff members of the School of E.E.C.S, particularly Ferne Simendinger for her invaluable assistance.

Last but not the least, I would like to thank the people who made my stay at Corvallis all the more pleasant and enjoyable with their delightful company.

TABLE OF CONTENTS

Page

TABLE OF CONTENTS (Continued)

Page

LIST OF FIGURES

Figure Page

LIST OF FIGURES (Continued)

Figure Page

LIST OF TABLES

Table Page

DEDICATION

To My Wife and Children With Love.

Some Communication Algorithms for Gaussian and Eisenstein-Jacobi networks

Chapter 1

Introduction

Parallel computers are classified into two categories: shared memory multiprocessors and distributed multi computers. A shared-memory multiprocessor consists of a number of processors that communicate with each other through one or more shared memory modules. In a distributed multiprocessor, however, the processing elements communicate with each other over an interconnection network. The interconnection network plays important role in the design of high performance computers. In the past many machines have been designed based on the toroidal [12][35] or the De Bruijn topology [36]. Recently two new networks, Gaussian and Eisenstein, were introduced [15][27]. They are based on quotient rings of Gaussian and Eisenstein integers. In this thesis, some topological properties and communication algorithms for these networks are described.

Gaussian networks were recently proposed [15][27]. These networks are based on quotient rings of Gaussian integers. Each node in these networks is labeled as $x + yi$. By choosing $\alpha = a + bi$, two nodes A and B are connected if and only if $(A - B) mod\ \alpha$ is ±1 or ±i. Gaussian networks are degree four symmetric networks with $a^2 + b^2$ nodes.

The other recently introduced interconnection networks are called the Eisenstein-Jacobi networks, and are based on the quotient rings of the Eisenstein-Jacobi integers. In these networks, nodes are labeled as $x + y\rho$, where $\rho = (1 + i\sqrt{3})/2$ and by choosing $\alpha = a + b\rho$, two nodes A and B are adjacent if $(A - B) mod\ \alpha$ is ± 1, $\pm\rho$, or $\pm\rho^2$. The Eisenstein-Jacobi networks are degree six symmetric with $a^2 + b^2 + ab$ nodes.

In the rest of this chapter, a brief summary of results in the thesis are given.

1.1 Edge Disjoint Hamiltonian Cycles

The generation of edge disjoint Hamiltonian cycles for Gaussian networks with $\alpha = a + bi$ and $gcd(a, b) = 1$ has already been investigated in [15][3]. However, finding edge disjoint Hamiltonian cycles in these networks when the $\gcd(a, b) = d > 1$ has been an open research problem. This thesis provides a solution to this problem in Chapter 2.

Some efficient communication algorithms can be designed based on these disjoint Hamiltonian cycles. For example, consider the all-to-all communication algorithm, where each node broadcasts a message to all other nodes [22][13][10]. In a single I/O port model, this problem can be solved optimally, by first generating a Hamiltonian cycle of the nodes and then exchanging the message as follows. In the first step, each node sends its message to its neighbor in the ring. In the *i*th step, $i = 2, 3, \ldots, (N - 1)$, where $N = a^2 + b^2$ is the total number of nodes, each node

sends to its neighbor the message which it receives in the previous step. If we could find edge disjoint Hamiltonian cycles, the above algorithm could also be extended to the case of a multiport model, where a node can send or receive from all its neighbors in unit time.

1.2 Communication Algorithms and Resource Placement

The design of efficient communication algorithms in a multicomputer is quite crucial to its performance. There are four communication primitives [22][10]: one-to-all, all-to-all, one-to-all personalized, and all-to-all personalized. In the case of one-to-all communication, a node wants to send its message to all other nodes in the network, whereas in all-to-all communication each node wants to send its message to all other nodes in the network. In one-to-all personalized communication, a node sends a distinct message to every node in the network, and in all-to-all personalized communication, every node performs one-to-all personalized communication. In this thesis, all these algorithms are studied for Eisenstein networks with $\alpha = k + (k+1)\rho$. These algorithms can communicate messages with no node getting a redundant message. These algorithms are described in detail in Chapter 3.

In parallel systems, some resources, such as certain software, I/O, etc., are installed in some nodes and all other nodes share these resources. The optimal resource placement problem can be addressed in two different ways [27][31][11][6]. The so-called j-adjacency problem ensures that any non-resource node has j neighboring resource nodes and that no two resource nodes are next to each other. The other approach is the t-embedding or t-dominating set problem. This resource placement

will grant every node a resource within a distance of t and that the resource nodes are separated by at least $2t + 1$.

This problem is analogous to the selection of codewords in error-correcting codes [9][11][24][6]. Recently in [6], perfect codes based on Eisenstein-Jacobi graphs are addressed. In addition, the solution to a particular case of this problem is considered in [26][24]. In [26][24] it is assumed that the number of nodes in the network is $a^2 + b^2 - ab$, whereas in our case the number of nodes in the network is $a^2 + b^2 + ab$, where $a, b \geq 0$.

In Chapter 3, we present a solution to the t-dominating set problem for a subfamily of degree six circulant graphs based on Eisenstein-Jacobi integers. This gives a perfect code over Eisenstein-Jacobi integers as a solution to t-embedding and resource replacement problems. This problem has been solved in [25][29] with $\rho = \frac{-1+i\sqrt{3}}{2}$.

1.4 Thesis Structure and Results

In this section, we give details of the organization of the rest of this thesis. For each chapter, we summarize our main results.

Chapter 2 is devoted to the definition and properties of Gaussian graphs and constructing two edge disjoint Hamiltonian cycles for $\alpha = a + bi$ when $\gcd(a, b) = d > 1$. First, we define Gaussian graphs in Section 2.1 as graphs that are built over quotient rings of Gaussian integers. We also give a simple, two-dimensional drawing

for Gaussian graphs. This drawing is based on the fact that the number of graph vertices is the sum of two squares. In addition, we state theorems describing the diameter and the average distance of Gaussian graphs given in [27]. Section 2.2 is devoted to the problem of generating two edge disjoint Hamiltonian cycles in a Gaussian graph. There, we provide the previous work that has been done on this problem. Then in Section 2.3, an algorithm for generating such cycles for the Gaussian networks when $\gcd(a,b) = d > 1$ is given. Finally, Section 2.5 provides a summary of the chapter.

In Chapter 3, Section 3.1 gives the definition of the Eisenstein-Jacobi networks. We also provide a simple drawing to illustrate how to draw these graphs. In addition, we state theorems describing the diameter and the average distance of Eisenstein-Jacobi graphs. In Section 3.2, we present efficient algorithms for some basic communication operations in Eisenstein-Jacobi networks with $\alpha = k + (k+1)\rho$. In all of these algorithms, no node gets redundant data. In Section 3.2.1, we describe the procedure to implement an algorithm for one-to-all communication. Section 3.2.2 introduces an algorithm for all-to-all communication in the Eisenstein-Jacobi networks. In Section 3.2.3, an efficient algorithm for one-to-all personalized communication is given. Section 3.2.4 describes the procedure for all-to-all personalized communication in the Eisenstein-Jacobi networks.

We consider the problem of finding perfect t-dominating sets over Eisenstein-Jacobi graphs in Section 3.3. Section 3.3.1 is devoted to finding a solution to the resource

placement problem for Eisenstein-Jacobi networks. Finally, Section 3.4 draws the conclusions of this chapter.

Finally, in Chapter 4, after briefly describing the contributions, some open problems are presented.

Chapter 2

Edge Disjoint Hamiltonian Cycles in Gaussian Networks

2.1 Gaussian Graphs

Gaussian graphs have been recently introduced as a suitable topology model for interconnection networks [15][27]. In this chapter, we describe these networks using different Gaussian representations. The distance related properties and the diameter of these graphs are also explained. Finally, solutions to the problem of finding edge disjoint Hamiltonian cycles are given.

2.1.1 Quotient Rings of Gaussian Integers

The Gaussian integers $\mathbb{Z}[i]$ is the subset of the complex numbers with integer real and imaginary parts; that is,

$$\mathbb{Z}[i] = \{x + yi | x, y \in \mathbb{Z}\}$$

$Z[i]$ is a Euclidean domain and the norm is defined as:

$$N : Z[i] \rightarrow Z^{+}$$

$$x + yi \mapsto x^2 + y^2$$

Then, for every $\alpha, \pi \in Z[i]$ with $\pi \neq 0$ there exists $q, r \in Z[i]$ such that $\alpha = q\pi + r$ with $N(r) < N(\pi)$ where $N(\beta = a + bi) = a^2 + b^2$. This means that

there exists a Euclidean division algorithm for Gaussian integers. Typically, we denote the set of remainders of the division by any integer $N \neq 0$ as Z_N. This set is usually called the integers modulo N. In an analogous way, we can consider $Z[i]_\alpha$, i.e. the Gaussian integers modulo α. It is well known that the number of residue classes modulo a Gaussian integer $\alpha \neq 0$ is equal to $N(\alpha)$ [17] and various representations of these residue classes as points in a complex plane are given in [21].

One of the methods described in number theory books [17] is as follows. Given $0 \neq \alpha \in \mathbb{Z}[i]$, we consider the finite set of the Gaussian integers modulo α or $\mathbb{Z}[i]_\alpha := \{\beta \bmod \alpha \mid \beta \in \mathbb{Z}[i]\}$. For any $\alpha, \beta \in \mathbb{Z}[i]$, with $\alpha = a + bi$, one of the representation with smallest norm of its class in $\mathbb{Z}[i]_\alpha$ or $\beta \bmod \alpha$ can be computed as $\beta \bmod \alpha = \beta - \left[\frac{\beta\alpha^*}{a^2+b^2}\right]\alpha$. The operation [c+di] denotes rounding in Gaussian integers and is defined by $[c + di] = [c] + [d]i$ with [c] denoting the rounding to the closest integer. Besides, α^* is the conjugate of α.

2.1.2 Definition of Gaussian Graphs

In this section, we describe the Gaussian graphs and then show some examples to illustrate the interconnection topology.

Gaussian graphs are defined over quotient rings of Gaussian integers as follows.

Definition 2.1[27]: Given $\alpha = a + bi \in \mathbb{Z}[i]$, we define the graph $G_\alpha(V, E)$ where:

i) $V = \mathbb{Z}[i]_\alpha$ is the node set, and

ii) $E = \{(\beta, \gamma) \in V \times V \mid (\beta - \gamma) \equiv \pm 1, \pm i \ mod\ \alpha\}$ is the edge set.

We call G_α the Gaussian network generated by α.

Gaussian graphs are a regular degree four graph, since every vertex is adjacent with four other vertices. In addition, they are undirected, connected, and vertex-symmetric by definition.

Next, we describe how to draw these graphs in a constructive way. There are many ways to draw such a graph. We give four simple ways to represent these graphs as in [27][15][21]. For all these representations, we assume that $\alpha = a + bi \in \mathbb{Z}[i]$ with $0 < a \leq b$.

- ***Utah Representation:***

Gaussian graphs can be represented as a mesh-like fashion. The idea is to arrange $a^2 + b^2$ vertices in two attached square meshes of $a \times a$ and $b \times b$ vertices, respectively. The bigger square will be to the right of the smaller square. The zero vertex will be located at the bottom left corner of the smaller square as shown in Figure 2.1.

Each node is adjacent to four other vertices in four different directions: North, East, South, and West. Node B is adjacent to node A in the North direction if $i \equiv (B - A) mod\ \alpha$, in the East direction if $1 \equiv (B - A) mod\ \alpha$, in the South direction if

$-i \equiv (B - A) mod\ \alpha$, and in the West direction if $-1 \equiv (B - A) mod\ \alpha$. The connection between a node and its neighbors is clear if the vertex is not located at the borders of the two mesh squares. For the vertices located in the borders, Theorem 2.2 defines a simple way to find its adjacent neighbors.

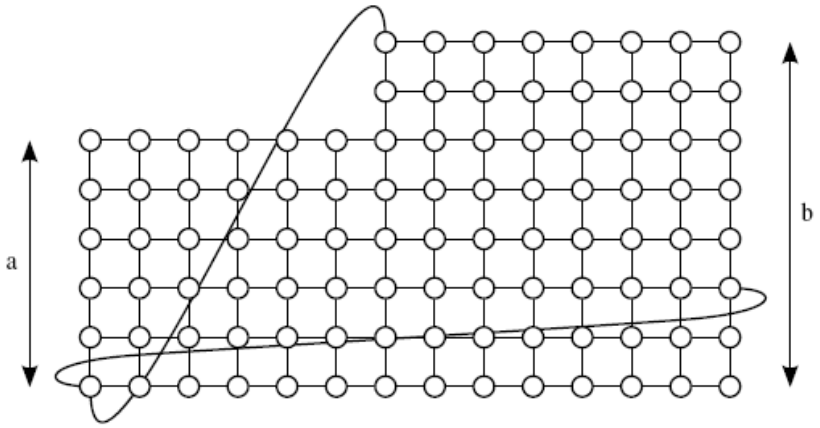

Figure 2.1: Utah Representation for the Graph Generated by $\boldsymbol{\alpha = 6 + 8i}$.

Theorem 2.2 [27]: Let $\alpha = a + bi$ be the graph generator where $0 < a \leq b$, and S be the set of all the vertices located at the borders of the two squares. Let $A = x + yi \in S$, then the wrap-around edges are defined as:

i) If $0 \leq x \leq b - 1$ then A is connected to $(x + a) + (b - 1)i$ from the South direction.

ii) If $b \leq x \leq a + b - 1$ then A is connected to $(x - b) + (a - 1)i$ from the South direction.

iii) If $0 \leq y \leq a - 1$ then A is connected to $(a + b - 1) + (y + b - a)i$ from the West direction.

iv) If $a \leq y \leq b - 1$ then A is connected to $(a + b - 1) + (y - a)i$ from the West direction.

Then, the graph defined by this adjacency pattern is isomorphic to the Gaussian graph generated by α. ■

The proof for adjacency pattern is easy by using the plane tessellations as in [14] or [37]. Figure 2.2 shows the plane tessellation associated with the Gaussian graph for $\alpha = 6 + 8i$.

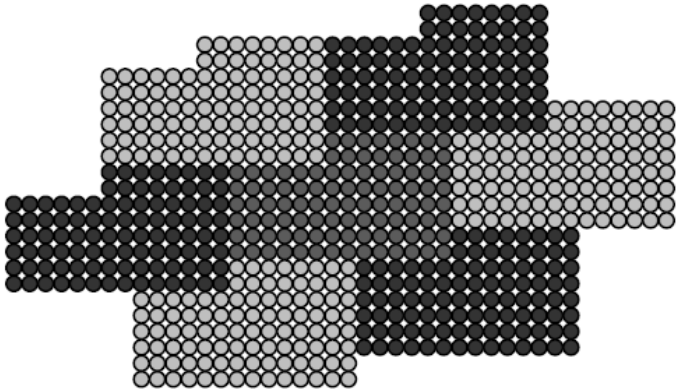

Figure 2.2: Plane Tessellation Associate with $\boldsymbol{\alpha = 6 + 8i}$.

Example 2.1: Let us consider the Gaussian graph generated by $\alpha = 3 + 4i$. We need to arrange the 25 vertices in two attached meshes of 3^2 and 4^2 vertices respectively. Then, by following the wrap-around edge patterns described in Theorem 2.2, we obtain the graph shown in Figure 2.3.

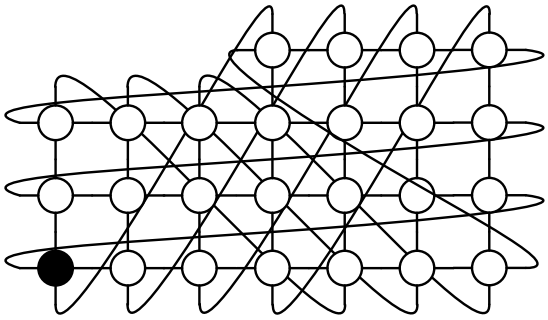

Figure 2.3: Utah Representation for the Graph Generated by $\boldsymbol{\alpha = 3 + 4i}$.

Example 2.2: Figure 2.4 shows some other Gaussian graphs for $\alpha = 6i, \alpha = 1 + 6i, \alpha = 2 + 6i$, and $\alpha = 6 + 6i$. As we can see, when $\alpha = 0$ the Gaussian graph becomes identical to Torus graph.

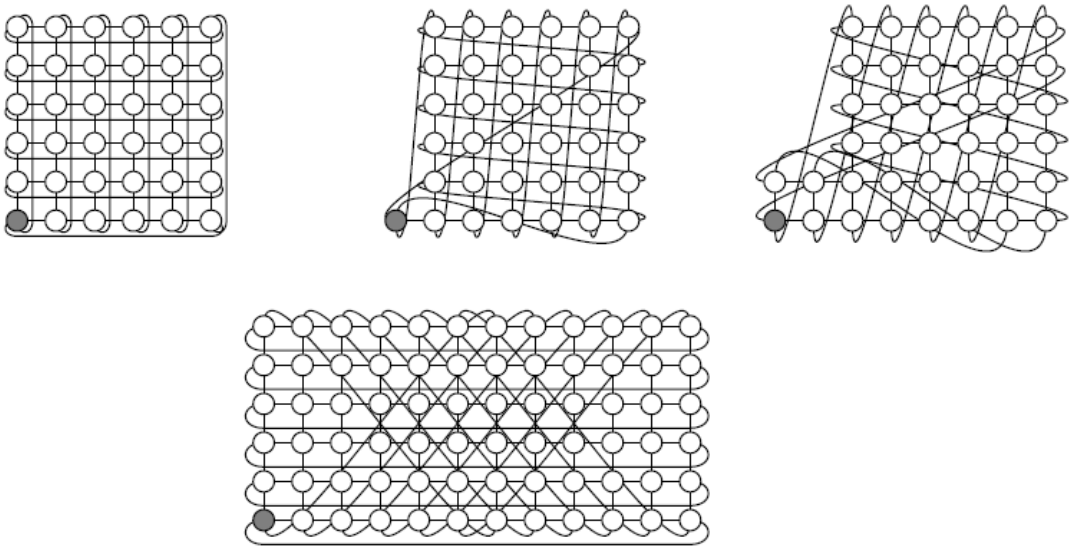

Figure 2.4: Utah Representation for Gaussian Graphs.

- ***Diamond Representation:***

This representation was recently introduced for Gaussian graphs in [15]. The idea here is to draw a square of size $|\alpha|^2$. We can draw this square by first locating the four points in the complex plane where the x-axis represents the real dimension and the y-axis represents the imaginary dimension. The points are $0, \alpha, i\alpha$, and $\alpha(i+1)$. Then, the graph vertices will be all points inside that square and the zero vertex. The only vertices landing on the square borders that will be counted on the graph are the points which land on the line 0 and α or 0 and $i\alpha$.

Each node in the square is connected to four other neighboring nodes, one from each side. The nodes in the border will have wrap-around links with other nodes. Theorem 2.3 defines a simple way for the wrap-around links.

Theorem 2.3: Let $\alpha = a + bi$ be the graph generator $0 < a \leq b$ and S be the set of all the vertices located at the square borders. Let $A = x + yi \in S$, then the wrap-around edges are defined as:

i) If $x = y = 0$ then A is connected to $-1 + \alpha$, $1 + i\alpha$ and $\alpha(1+i) - i$.

ii) If $0 < x \; and \; b > y$ then A is connected to $A + i\alpha + 1$ and $A + i\alpha - i$.

iii) If $0 > x \; and \; a > y$ then A is connected to $A + \alpha - 1$ and $A + \alpha - i$.

■

Figure 2.5 shows the plane tessellation associated with the Gaussian graph for $\alpha = 3 + 4i$ using the diamond representation.

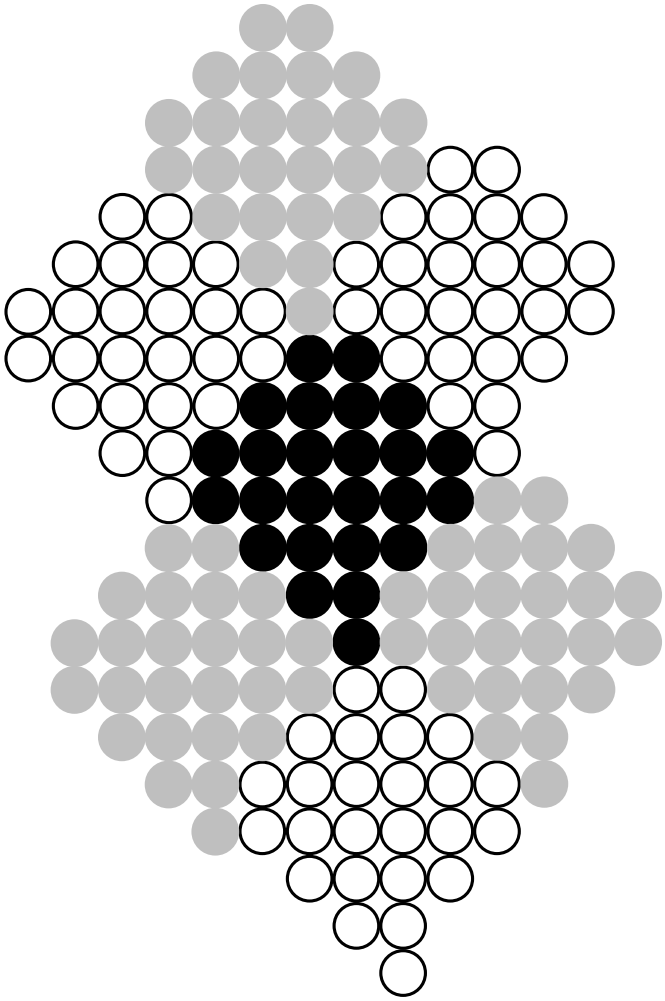

Figure 2.5: Plane Tessellation Associate with $\boldsymbol{\alpha = 3 + 4i}$.

Example 2.3: Let us consider the Gaussian graph generated by $\alpha = 3 + 4i$. We need to locate four points which are $0 = 0,\ \alpha = 3 + 4i,\ i\alpha = -4 + 3i,$ and $\alpha(i + 1) = -1 + 7i$. Then, by following the wrap-around edge patterns described in Theorem 2.3, we obtain the graph shown in Figure 2.6.

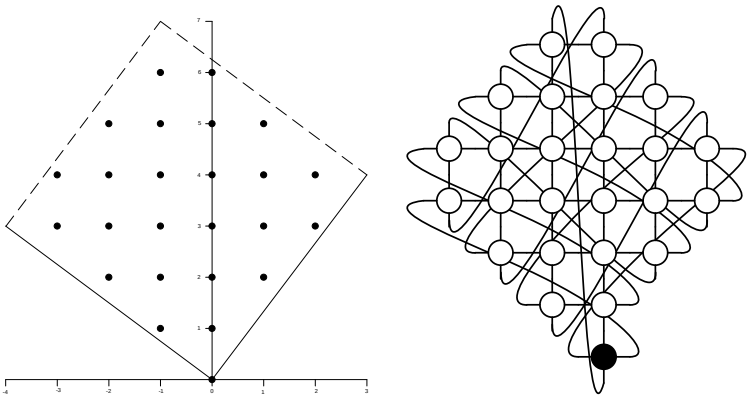

Figure 2.6: Diamond Representation for the Graph Generated by $\boldsymbol{\alpha = 3 + 4i}$.

- ***Rectangle Representation:***

This representation is more useful when $\gcd(a, b) = d > 1$. For $\alpha = a + bi$, with $\gcd(a, b) = d$, the set of point $S = \{\ x + yi\ \mid 0 \leq x < \frac{a^2+b^2}{d},\ 0 \leq y < d\}$ forms a complete residue classes $mod\ \alpha$ [21]. There are $a^2 + b^2$ points in this set. Furthermore, all points in this set are distinct under $mod\ \alpha$ as shown in [21].

The idea to represent the graph is to arrange $a^2 + b^2$ vertices in a rectangle of size $(\frac{a^2+b^2}{d}) \times d$. The zero node is located on the lower left corner of the rectangle, as Figure 2.7 shows. Each interior node in the rectangle is connected with four other neighboring nodes, one from each side. The nodes in the border have wrap-around links with other nodes, as explained in Theorem 2.4.

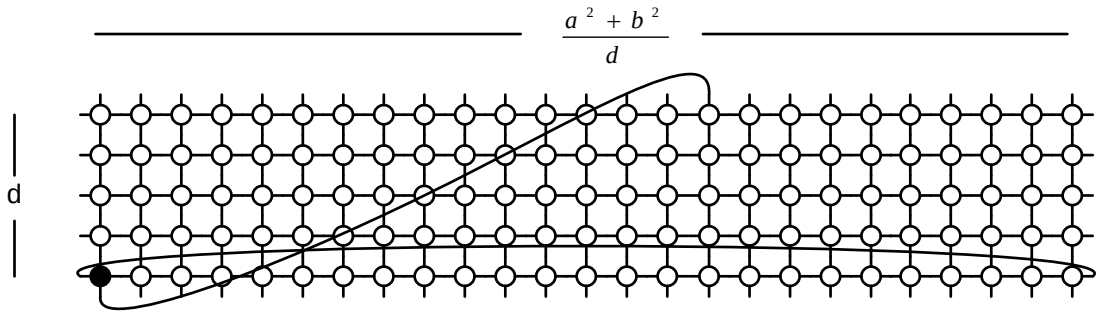

Figure 2.7: Rectangle Representation for the Graph Generated by $\boldsymbol{\alpha = 5 + 10i}$.

Theorem 2.4[21]: Let $\alpha = a + bi$ be the graph generator where $0 < a \leq b, \gcd(a,b) = d$, $r = \frac{a^2+b^2}{d}$ and S be the set of all vertices located at the rectangle borders. First, we need to find $-i \ (mod\ \alpha)$, since $\gcd(a,b) = d$, we can write d as:

$$ua + vb = d \tag{1}$$

Let $n = av - bu \ (mod\ r)$. Then, $-i \ (mod\ \alpha) = n + (d-1)i$. Now let $A = x + yi \in S$, then the wrap-around edges are define as:

i) If $0 \leq x < r - n$ then A is connected to $(x + n) + (d-1)i$ from the South.

ii) If $r - n < x \leq r - 1$ then A is connected to $(x - (r - n)) + (d-1)i$ from the South.

iii) If $0 \leq y \leq d - 1$ then A is connected to $(r-1) + yi$ from the West.

■

Example 2.4: Let us consider the Gaussian graph generated by $\alpha = a + bi = 4 + 4i$. Here, the $\gcd(a, b) = 4$ and $r = 8$. We need to arrange the 32 vertices in a rectangle of size 4×8. Then, by solving equation (1) we can choose $u = -1, v = 2$. Then $-i \; mod \; \alpha = 4 + 3i$. Now, by following the wrap-around edge patterns described in Theorem 2.4, we obtain the graph shown in Figure 2.8.

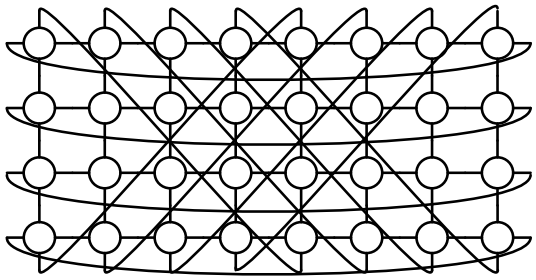

Figure 2.8: Rectangle Representation for the Graph Generated by $\boldsymbol{\alpha = 4 + 4i}$.

2.1.3 Distance Properties of Gaussian Graphs

In this section, the properties for the diameter and the average distance of a Gaussian graph are reviewed. The diameter k is defined as the length of any longest shortest path among all pairs of vertices of the graph. To this aim, we will describe the vertex-to-vertex distance distribution of any Gaussian graph [27]. Note that the distance between the two vertices β and γ in G_α can be expressed as:

$$D_\alpha(\beta, \gamma) = \min\{|x| + |y| \mid (\beta - \gamma) \equiv x + yi \; (mod \; \alpha)\}.$$

Also, since G_α is vertex-symmetric, we can define the weight of vertex β (its distance to vertex 0) as:

$$W_\alpha(\beta) = D_\alpha(\beta, 0) = \min\{|x| + |y| \mid \beta \equiv x + yi \ (mod\ \alpha)\}.$$

Example 2.5: Let $\alpha = 3 + 4i$. We want to find the distance between node -1 and $1 + i$. Figure 2.9 shows an example of this distance. Now, $(-1) - (1 + i) = (-2 - i) mod\ \alpha = 2i\ mod\ \alpha$. Thus, the distance is 2. The shortest path from (-1) to $(1 + i)$ is given by $-1, -1 - i, 1 + i$. Note that if we use the path $-1, 0, 1, 1 + i$, the distance becomes 3.

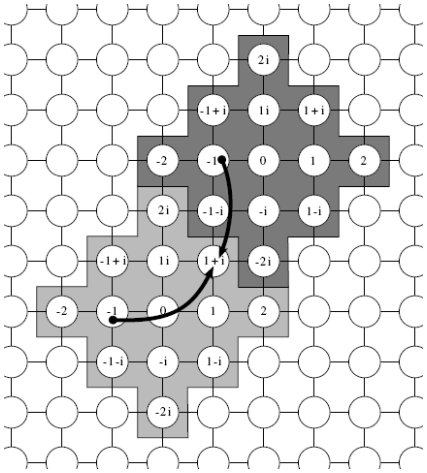

Figure 2.9: Graphical Representation of the Distance Induced by the Gaussian Graph Generated by $\boldsymbol{\alpha = 3 + 4i}$.

To compute the distance distribution of a Gaussian graph, it is enough to find the number of vertices of weight s, for $s = 0, 1, \ldots, k$, where k is the diameter of the graph. This number will be denoted as $\Delta_\alpha(s)$. Next, we state a couple of theorems that characterize the distance distribution of odd and even order Gaussian graphs.

Theorem 2.5[27]: Let $0 \neq \alpha = a + bi \in Z[i]$ be such that $0 < a \leq b, N = a^2 + b^2$ be an odd integer, and $t = \frac{a+b-1}{2}$. The distance distribution of the graph G_α is as follows:

i) $\Delta_\alpha(0) = 1.$

ii) $\Delta_\alpha(s) = 4s$ if $0 < s < t.$

iii) $\Delta_\alpha(s) = 4(b - s)$ if $t < s \leq b - 1.$

Theorem 2.6[27]: Let $0 \neq \alpha = a + bi \in Z[i]$ be such that $0 < a \leq b, N = a^2 + b^2$ be an even integer, and $t = \frac{a+b}{2}$. The distance distribution of the graph G_α is as follows:

i) $\Delta_\alpha(0) = 1.$

ii) $\Delta_\alpha(s) = 4s$ if $0 < s < t.$

iii) $\Delta_\alpha(t) = 2(b - 1).$

iv) $\Delta_\alpha(s) = 4(b - s)$ if $t < s < b.$

v) $\Delta_\alpha(b) = 1.$

We refer to [27] for the proofs of Theorem 2.5 and 2.6. Using the distance distribution of Gaussian graphs proved in Theorems 2.5 and 2.6 in [27], a closed formula for their diameter and the average distance can be easily deduced.

Corollary 2.7[27]: Let $0 \neq \alpha = a + bi \in Z[i]$ be such that $0 < a \leq b$. Let $N = a^2 + b^2$ be the norm of α. The diameter k of the Gaussian graph G_{a+bi} is:

$$k = \begin{cases} b & \text{if } N \text{ is even} \\ b-1 & \text{if } N \text{ is odd} \end{cases}$$

Example 2.6: Figure 2.10 shows a Gaussian network generated by $\alpha = 3 + 5i$ with 34 nodes and diameter $k = b = 5$.

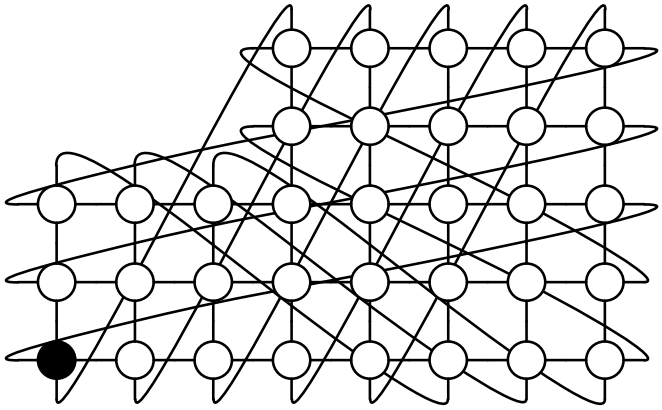

Figure 2.10: Gaussian Network with $\boldsymbol{\alpha = 3 + 5i}$.

2.2 Edge Disjoint Hamiltonian Cycles in Gaussian Networks

2.2.1 Previous Work

Finding edge disjoint Hamiltonian cycles for Gaussian graphs has been investigated before in [15][3]. In both papers, a solution to this problem is given only when $\gcd(a, b) = 1$, where the graph is generated by $\alpha = a + bi$.

However, the edge disjoint Hamiltonian cycle for Gaussian graphs generated by $\alpha = a + bi$ when the $\gcd(a, b) = d > 1$ has not been investigated, and in this section we give some solutions to this problem. When the $\gcd(a, b) = 1$, the first Hamiltonian cycle can be generated by adding 1 to the starting node $N(a + bi)$ times.

For example, starting from node 0, the jth node will be $0 + j\ (mod\ \alpha)$. The other cycle can be generated by adding i to the starting node. Figure 2.11 shows an example with $\alpha = 3 + 5i$; the solid line cycle and the dotted line cycle form two edge disjoint Hamiltonian cycles. The following argument shows why these two Hamiltonian cycles are edge disjoint.

Suppose $0 + s = 0 + t\ (mod\ \alpha)$ for some integers s and t. Then $s - t = 0\ mod\ \alpha$, and $s - t$ is an integer. This implies $\alpha|(s - t)$. However, the smallest integer that α can divide is $a^2 + b^2$. This implies $s - t = a^2 + b^2$. Thus, the cycle is Hamiltonian. Similarly, it can be proved that adding *i*'s to a starting node will result in a Hamiltonian cycle. Furthermore, from any node, the first cycle traverses along the real dimension edges and the second along the imaginary dimension edges; so there is no common edge between these two cycles [15][3].

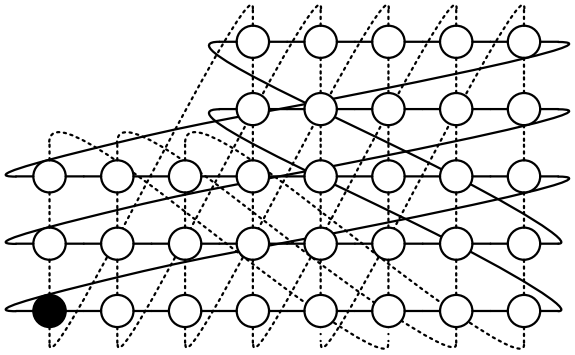

Figure 2.11: Edge Disjoint Hamiltonian Cycle with $\boldsymbol{\alpha = 3 + 5i}$.

2.3 Edge Disjoint Hamiltonian Cycles in Gaussian Networks with $gcd(a,b) = d > 1$

Here, we consider how to construct two edge disjoint Hamiltonian cycles when the $\gcd(a,b) = d > 1$. The following theorem is useful in generating the Hamiltonian cycle.

Theorem 2.8: Let $\alpha = a + bi \in \mathbb{Z}[i]$ and the $\gcd(a,b) = d$. Then in the Gaussian network generated by α, there exist two sets of d node disjoint cycles, each cycle is of length $r = (a^2 + b^2)/d$. The edges in the first set of cycles are in the real dimension and the second along the imaginary dimension.

Proof: Start with node β_1 and traverse the successive nodes with node addresses increased by $+1$ (or always increased by $+i$). Since the number of nodes is finite, some node has to be revisited such that the cycle is the length of the least integer k, i.e. $\beta_1 + k = \beta_1 \, mod \, \alpha$. This implies $k = 0 \, mod \, \alpha$, i.e. k is the least Gaussian multiple of α that is an integer.

This means $k = \beta\alpha = (x + yi)(a + bi) = (ax - by) + (ay + bx)i$. Since k is an integer, the imaginary part is zero and so $ay + bx = 0$. Since the $\gcd(a,b) = d$, we get $a = a_1 d$, $b = b_1 d$ and $\gcd(a_1, b_1) = 1$. Therefore, $d(a_1 y + b_1 x) = 0$. Since the $\gcd(a_1, b_1) = 1$, we get $-a_1 y = b_1 x$. This implies $y = -gb_1, x = ga_1$. Therefore, $k = (ax - by) = (aga_1 + bgb_1) = gd\left({a_1}^2 + {b_1}^2\right) = g(\frac{a^2+b^2}{d})$.

We need to prove that $g = 1$. If $\frac{a^2+b^2}{d} = 0 \, mod \, \alpha$, then $g = 1$. Now,

$$\frac{a^2+b^2}{d} = d\left({a_1}^2 + {b_1}^2\right) = d(a_1 + b_1 i)(a_1 - b_1 i)$$

$$= (a_1 d + b_1 di)(a_1 - b_1 i) = (a + bi)(a_1 - b_1 i).$$

This implies $(a + bi)(a_1 - b_1 i) \, mod \, \alpha = 0$. Thus, $g = 1$.

This proves there is a cycle of length $\frac{a^2+b^2}{d}$ if we start with node β_1 and traverse the successive nodes by adding 1 to the previous node address.

In addition, we need to prove there are d node disjoint cycles. Suppose β_2 is not a node in the above cycle. We need to prove, by adding successive 1's to β_2 we get another cycle of length $\frac{a^2+b^2}{d}$ and there is no common node between this cycle and the above cycle. We will prove this by contradiction. Suppose $(\beta_2 + k_1) = (\beta_1 + k_2) mod \, \alpha$ where β_1 is a node in the first cycle and $k_1, k_2 \leq r$. This implies $\beta_1 + (k_2 - k_1) = \beta_2 \, (mod \, \alpha)$. This is not possible because $|k_2 - k_1| \leq r$; therefore, β_2 is not in the first cycle. Extending the above argument, we can see that there are d node disjoint cycles of length $\frac{a^2+b^2}{d}$.

■

Example 2.7: Consider the graph generated by $\alpha = 4 + 4i$. Then, the graph has four node disjoint cycles in each dimension, and each cycle contains 8 nodes. Figure 2.12 shows this graph in rectangle representations.

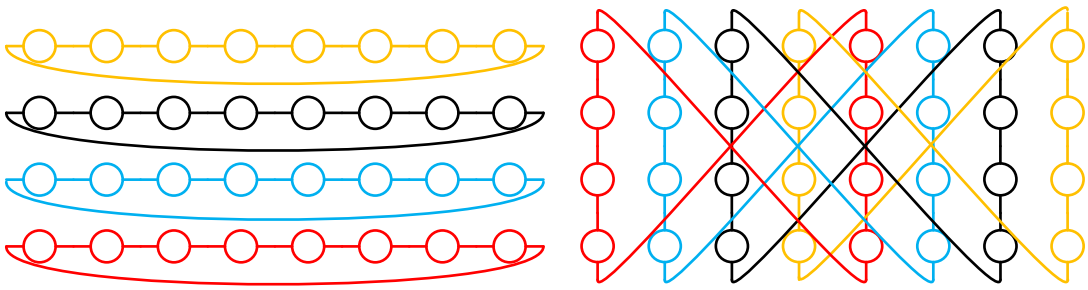

Figure 2.12: Four Disjoint Cycles in both dimensions in Gaussian Network with $\boldsymbol{\alpha = 4 + 4i}$.

Theorem 2.8 indicates that we have d disjoint cycles of length $(a^2 + b^2)/d$ in each dimension. Thus, we not only need to combine these cycles to form one Hamiltonian cycle but also to make sure that the remaining edges also form another Hamiltonian cycle.

Now based on the value of d, we can divide the Gaussian networks into two cases: Gaussian network with odd number of cycles (d-odd) and Gaussian network with even number of cycles (d-even). First, we present and prove how to generate such cycles when d is odd, and then, by using the same argument, we show how to generate these cycles when d is even.

Let $d = 2t + 1$, the node visiting sequence for the first Hamiltonian cycle when d is odd, is as follows:

1. Starting from the zero node, go to node $-i$ which takes us to cycle number $d - 1$; from this node, go in the right direction (along the real dimension) and visit all the nodes except the last two nodes in this cycle.

2. The following two steps are repeated for $(t-1)$ times.
    a. Go to the next cycle by adding $-i$. Then going along the left direction, visit all the nodes except the last node in this cycle.
    b. Go to the next cycle by adding $-i$. Then going along the right direction, visit all the nodes except the last three nodes in this cycle.
3. Then visit all nodes in cycles 2 through $d-1$ not visited in Step 1 and Step 2.
4. From the last node visited in cycle $d-1$, visit the adjacent node in cycle number zero and all the remaining nodes in this cycle.
5. Then move to cycle number 1, and visit all the nodes in this cycle, then get back to the starting node.

Figure 2.13 shows the node sequence used to generate the first Hamiltonian cycle.

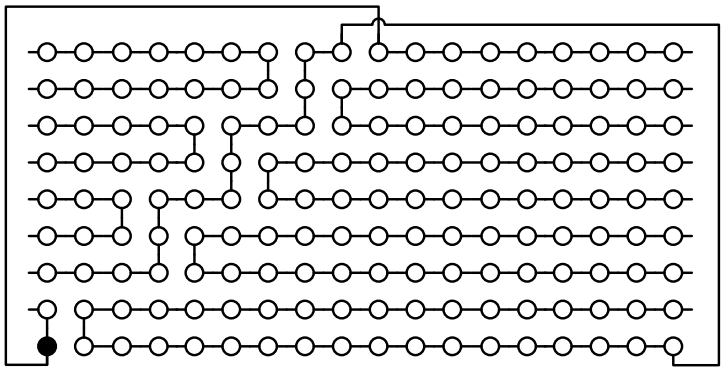

Figure 2.13: First Hamiltonian Cycle for $\boldsymbol{G_{9+9i}}$.

Table 2.1 shows the node sequence generating the first Hamiltonian cycle, and Table 2.2 shows another node sequence generating the second Hamiltonian cycle when d is odd. In each of the tables, we represent a Gaussian integer as a pair of two

numbers $x + yi = (x, y)$ where $r = (a^2 + b^2)/d$. These numbers need to be taken under $(mod\ \alpha)$.

Cycle Number	Nodes	
0	(0,0)	Step 1
-1	(0,-1),(1,-1),(2,-1),…,(r-3,-1)	
-(2j)	(r-2j-1,-2j),(r-2j-2,-2j),…,(r-2j+1,-2j)	Step 2 j=1,2,...,t-1
-(2j+1)	(r-2j+1,-(2j+1)),(r-2j+2,-(2j+1)),…,(r-2j-3,-(2j+1))	
…	…	...
2(k-t)-1	(r-2t-2+2k,-2t-1+2k),(r-2t-1+2k,-2t-1+2k), (r-2t+2k,-2t-1+2k)	Step3 k=1,2,...,t-1
2(k-t)	(r-2t+2k,-2t+2k)	
...	...	...
-1	(r-2,-1),(r-1,-1)	Step 4
0	(r-1,0),(r-2,0),…,(1,0)	
-d+1	(1,-d+1),(2,-d+1),…,(0,-d+1)	Step 5
0	(0,0)	

Table 2.1: The First Hamiltonian Cycle for $\boldsymbol{d}$ Odd.

Lemma 2.9: The sequence shown in Table 2.1 gives a Hamiltonian cycle.

Proof: Let $d = 2t + 1$. Each consecutive pair of nodes in the table represent an edge. Each row in the table represents the order in which the nodes are visited. The nodes visited in each row are in the real dimension. The last node of a row is adjacent to the first node in the next row, and this edge is in the imaginary dimension. In the last row, the last node of the cycle is $(0, 0)$ which is the starting node. The number of distinct nodes visited in each step is calculated as follows:

1. In Step one, $(r - 1)$ nodes are visited.
2. In Step two, $(t - 1)((r - 1) + (r - 3))$ nodes are visited.
3. In Step three, $(t - 1)(1 + 3)$ nodes are visited.
4. In Step four, $2 + (r - 1)$ nodes are visited.
5. In Step fife, r nodes are visited.

Thus, the total number of nodes is $(2t+1)r = dr = a^2 + b^2$ nodes. Therefore, the node sequence given in Table 2.1 generates a Hamiltonian cycle. ■

Cycle Number	Nodes	
0	(0,0)	Step 1
-1	(-1,0),(-1,1),(-1,2),…,(-1,r-3)	
-(2j)	(-2j,r-2j-1,),(-2j,r-2j-2),…,(-2j,r-2j+1)	Step 2 j=1,2,...,t-1
-(2j+1)	(-(2j+1),r-2j+1),(-(2j+1),r-2j+2),…,(-(2j+1),r-2j-3)	
…	…	
2(k-t)-1	(-2t-1+2k,r-2t-2+2k),(-2t-1+2k,r-2t-1+2k), (-2t-1+2k,r-2t+2k)	Step 3 k=1,2,...,t-1
2(k-t)	(-2t+2k,r-2t+2k)	
...	...	...
-1	(-1,r-2),(-1,r-1)	Step 4
0	(0,r-1),(0,r-2),…,(0,1)	
-d+1	(-d+1,1),(-d+1,2),…,(-d+1,0)	Step 5
0	(0,0)	

Table 2.2: The Second Hamiltonian Cycle for $\boldsymbol{d}$ Odd.

Lemma 2.10: The sequence shown in Table 2.2 gives a Hamiltonian cycle.

Proof: It is similar to the proof of Lemma 2.9. Note that the node sequence given in this table is obtained from the node sequence in Table 2.1 as follows. If two consecutive nodes differ by $+1, -1, +i,$ or $-i$, the corresponding nodes in Table 2.2 differ by $+i, -i, +1$, or -1, respectively. Here, the nodes visited in each row are in the imaginary dimension; again, the last node in a row is adjacent to the first node in the next row, and this edge is along the real dimension. Since the nodes are all distinct, with $a^2 + b^2$ elements, the sequence forms a Hamiltonian cycle. ■

Theorem 2.11: The two Hamiltonian cycles constructed using Lemma 2.9 and 2.10 are edge disjoint.

Proof: Now we need to prove that these two Hamiltonian cycles are edge disjoint. As Table 2.1 shows, we construct the first Hamiltonian cycle by visiting the nodes using the edges in the real dimension by either adding or subtracting 1, unless we want to visit the next cycle, in which case we need to use one edge in the imaginary dimension. Similarly, Table 2.2 shows that we construct the second Hamiltonian cycle by visiting the nodes using the edges in the imaginary dimension by either adding or subtracting i, unless we want to visit the next cycle, in which case we use one edge in the real dimension.

Note that the first Hamiltonian cycle uses exactly $2d-2$ edges belonging to the imaginary dimension, and the remaining edges are in the real dimension. On the other hand, the second Hamiltonian cycle uses $2d-2$ edges belonging to the real dimension, and the remaining edges are along the imaginary dimension.

In order to prove that these two Hamiltonian cycles are edge disjoint, we have to show the $2d-2$ edges, which belong to the imaginary dimension in Table 2.1, are not being used in Table 2.2; similarly, we have to show the $2d-2$ edges, which belong to the real dimension in Table 2.2, are not in Table 2.1.

Table 2.3 shows all edges traveling in the imaginary dimension used in the first Hamiltonian cycle using the representation as in Table 2.1 and their equivalent edge values using the representation as in Table 2.2. Similarly, Table 2.4 shows all edges

belonging to the real dimension in the second Hamiltonian cycle using the representation as in Table 2.2 and their equivalent edge values using the representation as in Table 2.1.

As we can see, none of the edges given in Table 2.3 is in Table 2.2, and similarly, none of the edges given in Table 2.4 is in Table 2.1. The edges shown in columns A and B are equivalent to the edges shown in columns C and D. This is because $A - C \equiv B - D \equiv (r - ri) mod\ \alpha \Rightarrow r(1 - i) = \frac{(a+bi)(a-bi)(1-i)}{d} \equiv 0\ mod\ \alpha$. Thus, these two Hamiltonian cycles generated by the node sequences given in Tables 2.1 and 2.2 are edge disjoint. ■

First Hamiltonian cycle edges in imaginary direction		Equivalent edges value	
A	B	C	D
(0,0)	(0,-1)	(0,0)	(0,-1)
(0,1)	(1,1)	(0,1)	(1,1)
(0,0)	(0,-1)	(0,0)	(0,-1)
(r-1,0)	(r-1,-1)	(-1,r)	(-1,r-1)
(r-2j-1,-2j+1)	(r-2j-1,-2j)	(-2j-1,r-2j+1)	(-2j-1,r-2j)
(r-2j,-2j+1)	(r-2j,-2j)	(-2j,r-2j+1)	(-2j,r-2j)
(r-2j+1,-2j)	(r-2j+1,-(2j+1))	(-2j+1,r-2j)	(-2j+1,r-(2j+1))
(r-2j,-2j)	(r-2j,-(2j+1))	(-2j,r-2j)	(-2j,r-(2j+1))

Table 2.3: The $\boldsymbol{2d - 2}$ Edges Belonging to the Imaginary Dimension in Table 2.1 (here $\boldsymbol{j = 1, 2, \ldots, t - 1}$).

First Hamiltonian cycle edges in imaginary direction		Equivalent edges value	
A	B	C	D
(0,0)	(-1,0)	(0,0)	(-1,0)
(1,0)	(1,1)	(1,0)	(1,1)
(0,0)	(-1,0)	(0,0)	(-1,0)
(0,r-1)	(-1,r-1)	(r,-1)	(r-1,-1)
(-2j+1,r-2j-1)	(-2j,r-2j-1)	(r-2j+1,-2j-1)	(r-2j,-2j-1)
(-2j+1,r-2j)	(-2j,r-2j)	(r-2j+1,-2j)	(r-2j,-2j)
(-2j,r-2j+1)	(-(2j+1),r-2j+1)	(r-2j,-2j+1)	(r-(2j+1),-2j+1)
(-2j,r-2j)	(-(2j+1),r-2j)	(r-2j,-2j)	(r-(2j+1),-2j)

Table 2.4: The $\boldsymbol{2d-2}$ Edges Belonging to the Real Dimension in Table 2.2 (here $\boldsymbol{j = 1, 2, \dots, t-1}$).

Example 2.8: Let $\alpha = 3 + 6i$. By following the node sequence given in Table 2.1, we get the first Hamiltonian cycle as:

$$0, -i, 1-i, 2-i, 3-i, 4-i, 5-i, 6-i, 7-i, 8-i, 9-i, 10-i, 11-i, 12-i, 13-i, 14-i, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 1+i, 2+i, 3+i, 4+i, 5+i, 6+i, 7+i, 8+i, 9+i, 10+i, 11+i, 12+i, 13+i, 14+i, i, 0.$$

Then we take the $mod\ \alpha$ for each of these nodes. The modular operation varies depending on which representation we use. Moreover, by following the node sequence given in Table 2.2, we get the second Hamiltonian cycle as:

$$0, -1, -1+i, -1+2i, -1+3i, -1+4i, -1+5i, -1, +6i, -1+7i, -1+8i, -1+9i, -1+10i, -1+11i, -1+12i, -1+13i, -1+14i, 14i, 13i, 12i, 11i, 10i, 9i, 8i, 7i, 6i, 5i, 4i, 3i, 2i, i, 1+i, 1+2i, 1+3i, 1+4i, 1+5i, 1+6i, 1+7i, 1+8i, 1+9i, 1+10i, 1+11i, 1+12i, 1+13i, 1+14i, 1, 0.$$

Figure 2.14 shows the first Hamiltonian cycle, and Figure 2.15 shows the second Hamiltonian cycle. It can be seen that no edge is common in both cycles. The cycles are shown using both Utah and rectangle representations.

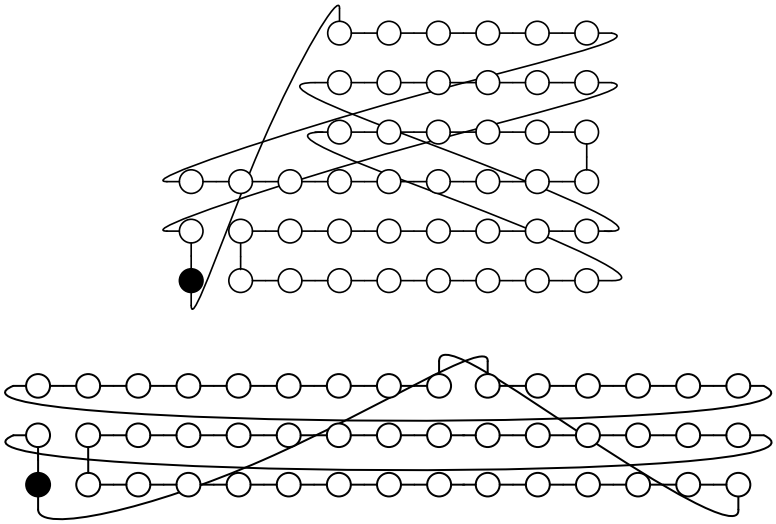

Figure 2.14: The First Edge Disjoint Hamiltonian Cycle for $\boldsymbol{\alpha = 3 + 6i}$.

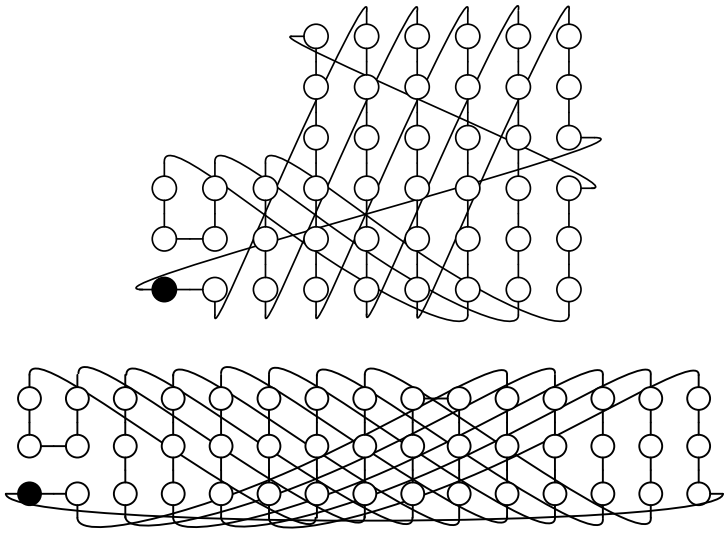

Figure 2.15: The Second Edge Disjoint Hamiltonian Cycle for $\boldsymbol{\alpha = 3 + 6i}$.

Next, we describe the result when d is even. Let $d = 2t$, the node visiting sequence for the first Hamiltonian is as follows:

1. Starting from the zero node, go to node $-i$ which takes us to cycle number $d - 1$; from this node, go in the right direction (along the real dimension) and visit all the nodes except the last two nodes in this cycle.
2. The following two steps are repeated for $(t - 2)$ times.
    a. Go to the next cycle by adding $-i$. Then going along the left direction, visit all the nodes except the last node in this cycle.
    b. Go to the next cycle by adding $-i$. Then going along the right direction, visit all the nodes except the last three nodes in this cycle.
3. Then go one more step to the right by adding one. After that, go to cycle number 2, and visit all the nodes in this cycle.
4. Then visit all nodes in cycles 2 through $d - 1$ not visited in Step 1 and Step 2.
5. From the last node visited in cycle $d - 1$, visit the adjacent node in cycle number zero and all the remaining nodes in this cycle.
6. Then move to cycle number 1, and visit all the nodes in this cycle, then get back to the starting node.

Figure 2.16 shows the node sequence used to generate the first Hamiltonian cycle.

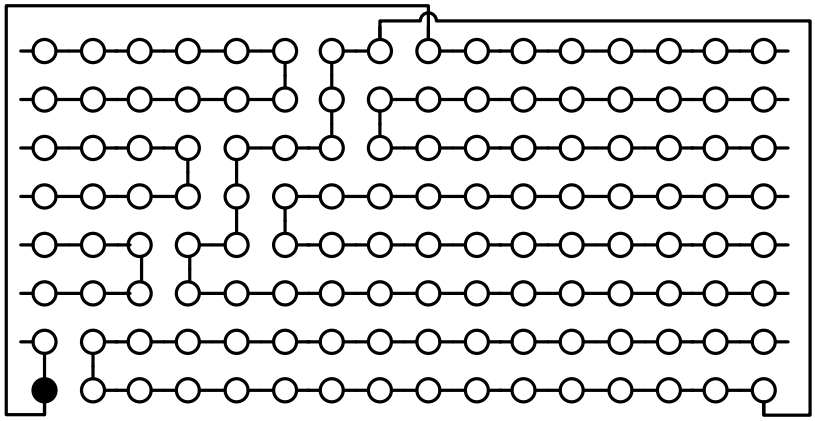

Figure 2.16: First Hamiltonian Cycle for $\boldsymbol{G_{8+8i}}$.

Table 2.5 shows the node sequence generating the first Hamiltonian cycle when d is even, and Table 2.6 shows another node sequence generating the second Hamiltonian cycle when d is even. In each of the tables, we represent a Gaussian integer as a pair of two numbers $x + yi = (x, y)$ where $r = (a^2 + b^2)/d$. These numbers need to be taken under $(mod\ \alpha)$.

Cycle Number	Nodes	
0	(0,0)	Step 1
-1	(0,-1),(1,-1),(2,-1),…,(r-3,-1)	
-(2j)	(r-2j-1,-2j),(r-2j-2,-2j),…,(r-2j+1,-2j)	Step2 j=1,2,...,t-2
-(2j+1)	(r-2j+1,-(2j+1)),(r-2j+2,-(2j+1)),…,(r-2j-3,-(2j+1))	
…	…	...
-d+3	(r-2t+2,-d+3)	Step 3
-d+2	(r-2t+2,-d+2),(r-2t+1,-d+2),...,(r-2t+3,-d+2)	
-d+3	(r-2t+3,-d+3),(r-2t+4,-d+3)	Step 4 k=2,3,...,t-2
2(k-t)	(r-2t+2k,-2t+2k)	
2(k-t)+1	(r-2t+2k,-2t+2k+1),(r-2t+2k+1,-2t+2k+1), (r-2t+2k+2,-2t+2k+1)	
...	....	
-2	(r-2,-2)	
-1	(r-2,-1),(r-1,-1)	Step 5
0	(r-1,0),(r-2,0),…,(1,0)	
-d+1	(1,-d+1),(2,-d+1),…,(0,-d+1)	Step 6
0	(0,0)	

Table 2.5: The First Hamiltonian Cycle for ***d*** Even.

Lemma 2.12: The sequence shown in Table 2.5 gives a Hamiltonian cycle.

Proof: It is similar to the proof of Lemma 2.9.

Cycle Number	Nodes	
0	(0,0)	Step 1
-1	(-1,0),(-1,1),(-1,2),…,(-1,r-3)	
-(2j)	(-2j,r-2j-1,),(-2j,r-2j-2),…,(-2j,r-2j+1)	Step 2 j=1,2,...,t-2
-(2j+1)	(-(2j+1),r-2j+1),(-(2j+1),r-2j),…,(-(2j+1),r-2j-3)	
…	…	
-d+3	(-d+3,r-2t+2)	Step 3
-d+2	(-d+2,r-2t+2),(-d+2,r-2t+1),...,(-d+2,r-2t+3)	
-d+3	(-d+3,r-2t+3),(-d+3,r-2t+4)	Step 4 k=2,3,...,t-2
2(k-t)	(-2t+2k,r-2t+2k)	
2(k-t)+1	(-2t+2k+1,r-2t2k),(-2t+2k+1,r-2t+2k+1), (-2t+2k+1,r-2t+2k+2)	
...	....	
-2	(-2,r-2)	
-1	(-1,r-2),(-1,r-1)	Step 5
0	(0,r-1),(0,r-2),…,(0,1)	
-d+1	(-d+1,1),(-d+1,2),…,(-d+1,0)	Step 6
0	(0,0)	

Table 2.6: The Second Hamiltonian Cycle for ***d*** Even.

Lemma 2.13: The sequence shown in Table 2.6 gives a Hamiltonian cycle.

Proof: It is similar to the proof of Lemma 2.10.

Theorem 2.14: The two Hamiltonian cycles constructed using Lemma 2.12 and 2.13, are edge disjoint.

Proof: The proof is similar to that of Theorem 2.11. Table 2.7 shows all edges traveling in the imaginary dimension used in the first Hamiltonian cycle, using the representation as in Table 2.5 and their equivalent edge values, using the representation as in Table 2.6. Similarly, Table 2.8 shows all edges belonging to the

real dimension used in the second Hamiltonian cycle, using the representation as in Table 2.6 and their equivalent edge values using the representation as in Table 2.5.

As we can see, none of the edges given in Table 2.5 is in Table 2.6, and similarly, none of the edges given in Table 2.7 is in Table 2.8. Thus, these two Hamiltonian cycles generated by the node sequences given in Table 2.1 and 2.2 are edge disjoint.

■

First Hamiltonian cycle edges in imaginary direction		Equivalent edges value	
A	B	C	D
(0,0)	(0,-1)	(0,0)	(0,-1)
(0,1)	(1,1)	(0,1)	(1,1)
(0,0)	(0,-1)	(0,0)	(0,-1)
(r-1,0)	(r-1,-1)	(-1,r)	(-1,r-1)
(r-4,3)	(r-4,2)	(-4,r+3)	(-4,r+2)
(r-5,3)	(r-5,2)	(-5,r+3)	(-5,r+2)
(r-2j-1,-2j+1)	(r-2j-1,-2j)	(-2j-1,r-2j+1)	(-2j-1,r-2j)
(r-2j,-2j+1)	(r-2j,-2j)	(-2j,r-2j+1)	(-2j,r-2j)
(r-2j+1,-2j)	(r-2j+1,-(2j+1))	(-2j+1,r-2j)	(-2j+1,r-(2j+1))
(r-2j,-2j)	(r-2j,-(2j+1))	(-2j,r-2j)	(-2j,r-(2j+1))

Table 2.7: The $\boldsymbol{2d-2}$ Edges Belonging to the Imaginary Dimension in Table 2.5 (here $\boldsymbol{j = 1, 2, \dots, t-2}$).

First Hamiltonian cycle edges in imaginary direction		Equivalent edges value	
A	B	C	D
(0,0)	(-1,0)	(0,0)	(-1,0)
(1,0)	(1,1)	(1,0)	(1,1)
(0,0)	(-1,0)	(0,0)	(-1,0)
(0,r-1)	(-1,r-1)	(r,-1)	(r-1,-1)
(3,r-4)	(2,r-4)	(r+3,-4)	(r+2,-4)
(3,r-5)	(2,r-5)	(r+3,-5)	(r+2,-5)
(-2j+1,r-2j-1)	(-2j,r-2j-1)	(r-2j+1,-2j-1)	(r-2j,-2j-1)
(-2j+1,r-2j)	(-2j,r-2j)	(r-2j+1,-2j)	(r-2j,-2j)
(-2j,r-2j+1)	(-(2j+1),r-2j+1)	(r-2j,-2j+1)	(r-(2j+1),-2j+1)
(-2j,r-2j)	(-(2j+1),r-2j)	(r-2j,-2j)	(r-(2j+1),-2j)

Table 2.8: The $\boldsymbol{2d-2}$ Edges Belonging to the Real Dimension in Table 2.6 (here $\boldsymbol{j = 1, 2, \dots, t-2}$).

2.4 Conclusion

In this chapter, it has been shown that the Gaussian network with $\gcd(a,b) = d > 1$ contains d node disjoint cycles. Any two consecutive nodes in these cycles differs by $\pm 1 (\text{or } \pm i)$. By removing $2d$ appropriate edges from these cycles and connecting $2d$ appropriate edges belonging to imaginary dimension, we have shown that the resultant sequence of edges form the first Hamiltonian cycle. Furthermore, the remaining edges form the second Hamiltonian cycle, which is edge disjoint from the first one.

Chapter 3

Communication Algorithms and Resource Placement in Eisenstein-Jacobi Networks

Eisenstein-Jacobi graphs have been recently introduced as a suitable topology for interconnection network in [25]. The methodology described in the previous chapter for the Gaussian networks can also be applied to define the Eisenstein-Jacobi graphs. Here, we describe the Eisenstein-Jacobi networks whose vertices are labeled by the elements of quotient rings of Eisenstein-Jacobi integers.

The rest of this chapter is organized as follows. In Section 3.1, Eisenstein-Jacobi networks are described. In addition, the distance properties are also given. In Section 3.2, some communication algorithms are designed. Section 3.3 describes some solutions to the resource placements in these networks.

3.1 Eisenstein-Jacobi Graphs

3.1.1 Quotient Rings of Eisenstein-Jacobi Integers

The ring of the Eisenstein-Jacobi integers $\mathbb{Z}[\rho]$ is defined as

$$Z[\rho] = \{ x + y\rho \mid x, y \in Z \}$$

where $\rho = (1 + i\sqrt{3})/2$ and $\rho^2 = -1 + \rho$. It can be proved that $Z[\rho]$ is a Euclidean domain with norm:

$$\mathcal{N}: \quad \mathbb{Z}[\rho] \longrightarrow \mathbb{N}$$

$$x + y\rho \longmapsto x^2 + y^2 + xy$$

Note that $\mathcal{N}(x + y\rho) = (x + y\rho)\overline{(x + y\rho)}$. Since $\overline{(x + y\rho)} = (x + y) - y\rho$,

$$\begin{aligned}\mathcal{N}(x + y\rho) &= (x + y\rho)\big((x + y) - y\rho\big)\\ &= x^2 + xy - xy\rho + xy\rho + y^2\rho - y^2\rho^2\\ &= x^2 + xy + y^2\rho - y^2(-1 + \rho)\\ &= x^2 + xy + y^2\rho + y^2 - y^2\rho\\ &= x^2 + y^2 + xy\end{aligned}$$

The units of $\mathbb{Z}[\rho]$ are the elements with norm equal to one; that is, $\{\pm 1, \pm\rho, \pm\rho^2\}$.

For every $0 \neq \alpha \in \mathbb{Z}[\rho]$, we can consider $\mathbb{Z}[\rho]_\alpha = \{\beta (mod\ \alpha) \mid \beta \in \mathbb{Z}[\rho]\}$, which is clearly a finite set.

3.1.2 Definition of Eisenstein-Jacobi Networks

In this section, we describe Eisenstein-Jacobi networks, and then show some examples to illustrate the interconnection topology.

Eisenstein-Jacobi graphs are defined over the quotient rings of Eisenstein-Jacobi integers as follows.

Definition 3.1: Let $\alpha = x + y\rho \in \mathbb{Z}[\rho]$ and consider $\mathbb{Z}[\rho]_\alpha$. We denote the Eisenstein-Jacobi graph generated by α as $EJ_\alpha = (V, E)$, and it is defined as follows:

• $V = \mathbb{Z}[\rho]_\alpha$ is the set of nodes, and

• $E = \{(\beta, \gamma) \in V \times V \mid (\beta - \gamma) = \pm 1, \pm\rho, \pm\rho^2 \ (mod\ \alpha)\}$ is the set of edges.

Note that any Eisenstein-Jacobi graph is a regular graph of degree six, since every vertex is adjacent to exactly six other vertices.

The representation of the graph has the special feature that all the vertices are obtained at a minimum distance from the central vertex, which we have stated to be vertex zero. Figure 3.1 gives an example of this representation for the Eisenstein-Jacobi graph generated by $\alpha = 3 + 4\rho$.

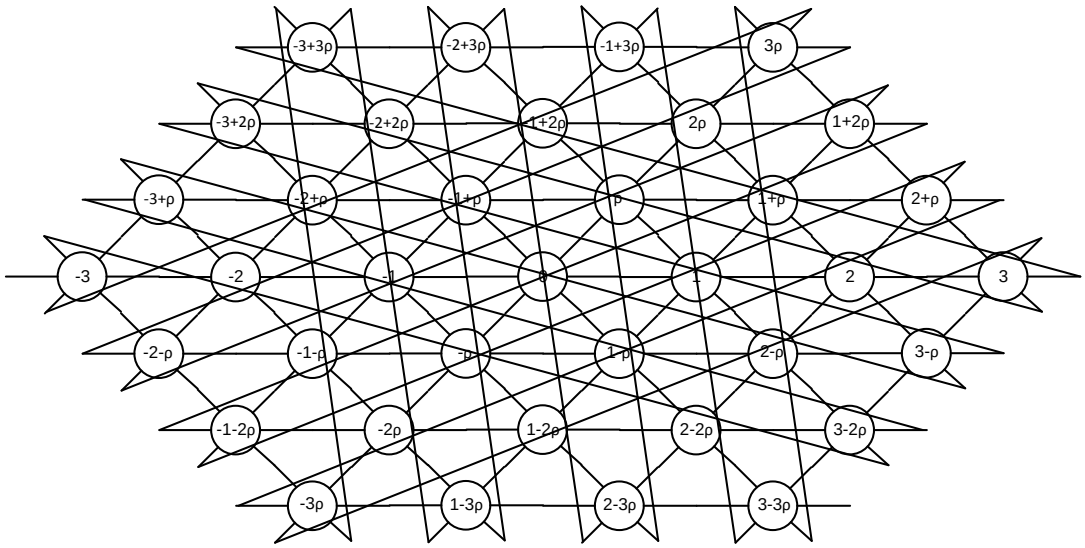

Figure 3.1: Eisenstein-Jacobi Network with $\boldsymbol{\alpha = 3 + 4\rho}$.

3.1.3 Distance Properties of Eisenstein-Jacobi Graphs

Over this quotient ring of Eisenstein-Jacobi integers, we can define a new distance, which is presented in [20][30] as follows.

Definition 3.4*:* Let $0 \neq \alpha \in \mathbb{Z}[\rho]$. For $\beta, \gamma \in \mathbb{Z}[\rho]_\alpha$, consider $x + y\rho + z\rho^2$ in the class of $\beta - \gamma$ with $|x| + |y| + |z|$ minimum. The distance D_α between β and γ is:

$$D_\alpha = |x| + |y| + |z|$$

As in the Gaussian case, D_α defines a distance over the quotient ring $\mathbb{Z}[\rho]_\alpha$.

Definition 3.5[15]: Let nonzero $w \in C$. Then w is in the j-th sector if w is between ρ^{j-1} and ρ^j.

Theorem 3.6[15]: Let $w \in C$. If w is in the j-th sector, then $D_\alpha(w, 0) = x + y$ where $w = x\rho^{j-1} + y\rho^j$ and $x, y \geq 0$. ■

Theorem 3.7[15]: Let $\alpha = a + b\rho$ be nonzero with $0 \leq a \leq b$, $T = (a + b)/2$, and $M = (a + 2b)/3$. For any positive integer t, let $W(t)$ be the number of nodes with distance t from 0 in the EJ network generated by α. Then

$$W(t) = \begin{cases} 1 & if\ t = 0 \\ 6t & if\ 1 \leq t \leq T \\ 18(M - t) & if\ T < t < M \\ 2 & if\ b \equiv a\ (mod\ 3) and\ t = M \\ 0 & if\ t > M \end{cases}$$

■

Example 3.8: For the EJ network in Figure 3.4, we have $\alpha = 2 + 3\rho$ with $4 + 9 + 6 = 19$ nodes in the network, and $T = 5/2$, $M = 8/3$. Theorem 3.6 distributes the nodes as follows:

$$W(0) = 1,\ \ W(1) = 6,\ \ W(2) = 12.$$

3.2 Communication Algorithms in Eisenstein-Jacobi Network

In parallel systems, processors need to exchange their data. The efficiency of the parallel programs depends on the efficiency of how this data is transmitted among the nodes. There are four different, commonly used communication patterns in parallel systems. They are one-to-all communication, all-to-all communication, one-to-all personalized communication, and all-to-all personalized communication [10][22]. Efficient implementation of these procedures can improve the performance of the system and reduce the development effort and cost.

The simplest and most fundamental communication operation is the one-to-all communication (or broadcasting). In this case, a node sends its message to all other nodes in the network. The all-to-all broadcast is defined as the process in which every node broadcasts its information to all other nodes in the system. One-to-all personalized communication is the process in which the source node sends unique information to each node in the system. All-to-all personalized communication is the process by whitch every node sends distinct information to every other node.

We now briefly explain how these communication algorithms are used in some real applications. For the time being, assume that a parallel system has n processor nodes.

Consider the matrix-vector multiplication $A_{n\times n} \times B_{n\times 1} = C_{n\times 1}$. Assume that the processor node 1 contains both A and B. A fast parallel algorithm can be designed as follows. Processor 1 can send the i^{th} row of A and the column vector B to the i^{th} processor node, and that node calculates the i^{th} row element of C. Here, we need to send B to all processor nodes, which is the one-to-all broadcasting. Alternatively, sending i^{th} row of A to the i^{th} processor node requires one-to-all personalized communication.

Now consider a matrix-matrix multiplication $A \times B = C$. Assume all processor nodes have both A and B. Each processor node is responsible for calculating a sub-matrix of C. If all processor nodes need the entire C matrix, then an all-to-all broadcasting is required.

To see the usefulness of the all-to-all personalized communication, consider the transpose of a matrix $A[i,j]^T = A[j,i]$, where $1 \leq i,j \leq n$. Assume that the i^{th} processor node contains the i^{th} row. The i^{th} processor node sends the element $A[i,1]$ to processor p_1, $A[i,2]$ to processor p_2, and $A[i,n]$ to processor p_n. In this case, every processor node sends a distinct element to every other processor node, which is the all-to-all personalized communication.

In this chapter, we present algorithms for each of the communication patterns mentioned above.

3.2.1 One-to-All Communication

In this section, an optimal broadcast routing for the Eisenstein-Jacobi networks where $\alpha = k + (k+1)\rho$ is presented. The algorithm is simple, and since the network is vertex symmetric, any node in the network can be assumed as the source node. Hence, it can be easily implemented in hardware.

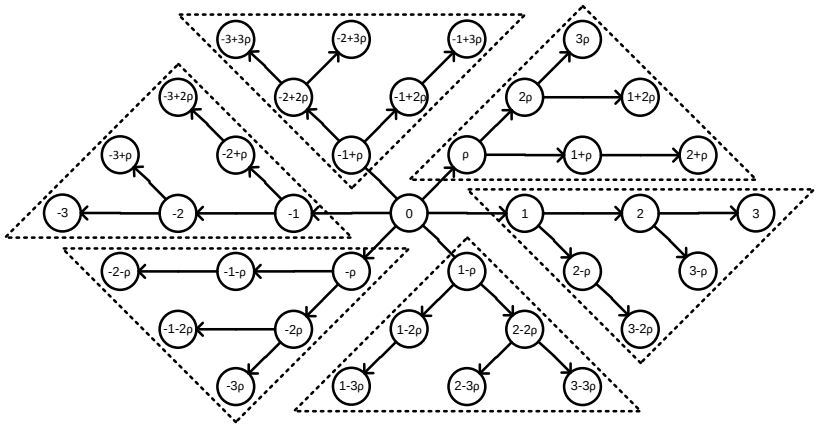

Figure 3.2: One-to-All Broadcasting in Eisenstein-Jacobi Network with $\boldsymbol{\alpha = 3 + 4\rho}$.

The number of nodes in the Eisenstein-Jacobi network with $\alpha = a + b\rho$ is equal to the sum of the squares of a and b plus their product. In our case, a and b are respectively k and $k+1$, and consequently, $(k^2 + (k+1)^2 + k(k+1)) \, mod \, 6 = 1$. This suggests that, once an arbitrary node is fixed, the rest of the network nodes can be divided into six different subsets, each having $\frac{k(k+1)}{2}$ nodes. Figure 3.2 shows the central node $(0,0)$ and the six partitions, each with $\frac{k(k+1)}{2}$ nodes. Each of these partitions forms a discrete triangle, and we call this special triangle a k-triangle.

The main idea of the broadcast algorithm is as follows. In the initial step of the broadcasting algorithm, the node $(0,0)$ sends its message to the six neighbors, and then each of these neighboring nodes broadcasts the message to the nodes in their k-triangle.

We assume a router model with half-duplex links and all-port capability. In this case, a node cannot send and receive messages through an edge at the same time. Routers can support both broadcast and unicast, with the first header bit in every packet (B/U) indicating the class of routing service, i.e. when B/U=1, this indicates the broadcasting, and B/U=0, the unicasting. In the case of broadcast routing, the second field in the packet header, denoted as distance, will be set to the network diameter k when the broadcast communication starts. Before each new hop, every router will decrement this field, and when distance reaches zero, the broadcast is completed. The third and last field in the packet header, denoted as direction, has six bits to indicate to the router the output ports to which the packet will be forwarded. Figure 3.3 shows the directions of the edges.

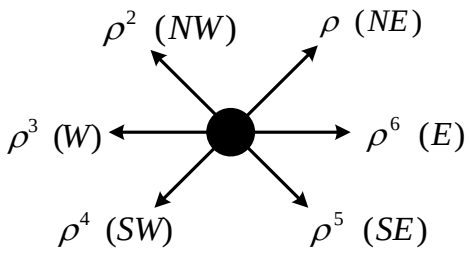

Figure 3.3: The Directions of the Edges.

Assume node $(0,0)$ is the source node. As explained before, in the initial step, it sends the message to its six neighboring nodes $1 = \rho^6\ (E), \rho^2(NW), -1 = \rho^3(W), -\rho = \rho^4(SW)$, and $-\rho^2 = \rho^5(SE)$, and these six nodes are responsible for broadcasting the message to the nodes within their own k-triangles, call it as the j^{th} k-triangle. As Figure 3.4 shows, at Step t, where $t = 1,2, \dots, k-1$, in the j^{th} k-triangle, the node $(t)\rho^j$ sends the message to nodes $(t+1)\rho^j$ and $(t)\rho^j + \rho^{j-1}$, whereas all other nodes which received the message at Step $t-1$ send the message to the nodes along the dimension ρ^{j-1}. The complete algorithm is shown in Figure 3.5.

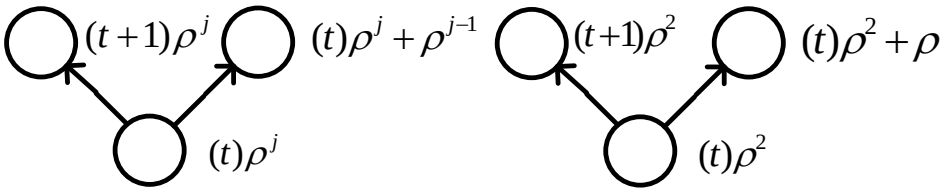

Figure 3.4: $\boldsymbol{\rho^j}$ Dimension.

In this algorithm, no node receives duplicate messages. Note that the utilization of the network links is balanced, as in step d, there are d packets traveling in each of the network quadrants. This means it is possible to make a balanced use of the E, NE, NW, W, SW, and SE network links when all nodes broadcast at once.

In this algorithm, since no node receives duplicate messages and it requires k steps where k is the network diameter, this algorithm is optimal. Besides, since the network is node symmetric, any node can be a source node, and thus this algorithm is universal for every node in the network.

The time complexity of the algorithm can be calculated as follows. Let t_s be the startup time to send a packet, t_w be the time to transfer one word, and w be the size of the packet. In this algorithm, a packet has to travel exactly k nodes away from the source node. Thus, the total time required is:

$$T_{one-to-all} = (t_s + t_w w)k.$$

Algorithm:

```
If distance = k then
  distance = distance -1
  send packet to E with
  direction = 110000
  send packet to NE with
  direction = 100001
  send packet to NW with
  direction = 000011
  send packet to W with
  direction = 000110
  send packet to SW with
  direction = 001100
  send packet to SE with
  direction = 011000
End
If distance = 0 then
  CONSUME pocket
End
   If 0 < distance < k then
  CONSUME pocket
  distance = distance -1
  -incoming pockets with direction = 110000
  Forward to E and Forward to SE with
direction = 010000
  -incoming pockets with direction = 100001
  Forward to NE and Forward to E with
direction = 100000
  -incoming pockets with direction = 000011
    Forward to NW and Forward to NE with
direction = 000001
  -incoming pockets with direction = 000110
  Forward to W and Forward to NW with
direction = 000010
  -incoming pockets with direction = 001100
  Forward to SW and Forward to W with
direction = 000100
  -incoming pockets with direction = 011000
  Forward to SE. Forward to SW with direction
= 001000
  -incoming pockets with direction = 100000
Forward to E
  -incoming pockets with direction = 010000
Forward to SE
  -incoming pockets with direction = 001000
Forward to SW
-incoming pockets with direction = 000100
Forward to W
  -incoming pockets with direction = 000010
Forward to NW
  -incoming pockets with direction = 000001
Forward to NE
End
```

Figure 3.5: One-to-All Broadcasting Algorithm for Eisenstein Network where $\boldsymbol{\alpha = k + (k+1)\rho}$.

3.2.2 All-to-All Communication

In this section, we describe the all-to-all broadcasting algorithm for the Eisenstein-Jacobi networks when $\alpha = k + (k+1)\rho$. In this communication pattern, each node in the network sends its message to every other node in the network. More clearly, each node performs a one-to-all broadcasting.

Since no node can send and receive a message over the same edge at the same time, we cannot apply the one-to-all broadcasting algorithm for all the nodes in the network at the same time. However, the idea of the proposed algorithm is based on a three-phase one-to-all broadcasting algorithm where in every phase, each node broadcasts its message to nodes in two distinct k-triangles. By the end of phase three, every node has broadcast its message to all six triangles. We use this three-phase algorithm in order to avoid network contention. The algorithm is first described using an example.

Example 3.9: Consider the EJ network generated by $\alpha = 2 + 3\rho$. In phase one, every node sends its messages using NW and W edges. Figure 3.6 shows step one of phase one. As Figure3.7 shows, in step two, every node broadcasts the received message in the previous step to its corresponding triangle. After that, phase two starts, in which each node sends its message using NE and E edges and repeats the broadcasting as in phase one. Finally, in phase three, each node sends its message using SE and SW edges and again repeats the broadcasting as in phase one.

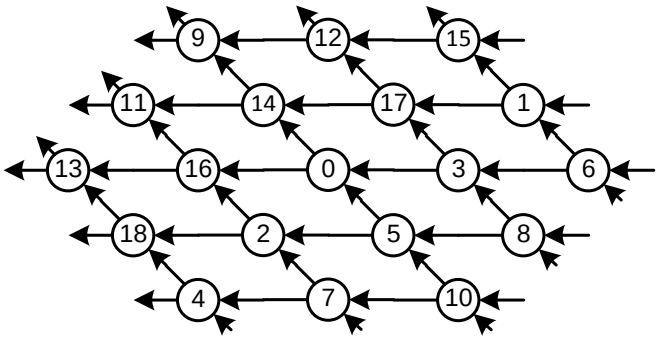

Figure 3.6: Step One of Phase One for All-to-all Broadcasting in $\boldsymbol{EJ_{2+3\rho}}$.

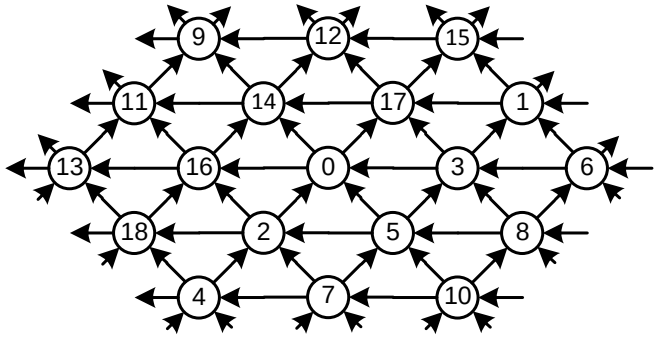

Figure 3.7: Step Two of Phase One for All-to-all Broadcasting in $\boldsymbol{EJ_{2+3\rho}}$.

Algorithm:

```
If distance = k and phase = 1 then
  distance = distance -1
  phase = 2
  send packet to NW with
  direction = 000011
  send packet to W with
  direction = 000110
else if distance = k and phase = 2 then
  distance = distance -1
  phase = 3
  send packet to E with
  direction = 110000
  send packet to NE with
  direction = 100001
else distance = k and phase = 3 then
  distance = distance -1
  send packet to SW with
  direction = 001100
  send packet to SE with
  direction = 011000
End

If distance = 0 and phase ≠ 3 then
  CONSUME pocket
  distance = k
else If distance = 0
  CONSUME pocket
End

   If 0 < distance < k then
    CONSUME pocket
    COMBINE pockets
    distance = distance -1
```

```
-incoming pockets with direction = 110000
  Forward to E and Forward to SE with
direction = 010000
  -incoming pockets with direction = 100001
 Forward to NE and Forward to E with
 direction = 100000
  -incoming pockets with direction = 000011
    Forward to NW and Forward to NE with
direction = 000001
  -incoming pockets with direction = 000110
  Forward to W and Forward to NW with
direction = 000010
  -incoming pockets with direction = 001100
  Forward to SW and Forward to W with
direction = 000100
  -incoming pockets with direction = 011000
  Forward to SE. Forward to SW with
direction = 001000

  -incoming pockets with direction = 100000
Forward to E
  -incoming pockets with direction = 010000
Forward to SE
  -incoming pockets with direction = 001000
Forward to SW
-incoming pockets with direction = 000100
Forward to W
  -incoming pockets with direction = 000010
Forward to NW
  -incoming pockets with direction = 000001
Forward to NE
End
```

Figure 3.8: All-to-All Broadcasting Algorithm for Eisenstein Network where $\boldsymbol{\alpha = k + (k+1)\rho}$.

As Figure 3.8 shows, in the initial step of phase 1, a node sends its message along two dimensions, NW and W. Thus, at the end of this step, a given node receives messages from at most two nodes, which are adjacent to it in SE and E directions. In the next step, the node that receives a message from its SE adjacent node (E adjacent node) sends the message along NW and NE directions (along W and NW directions). From this step onwards, all the messages flow only along the W, NW, and NE directions. This is because at the initial step, no message was sent along the NE direction, and so all the messages that flow along the NE direction must go only in this direction.

Therefore, a node can receive messages from its E, SE, and SW adjacent nodes and send along W, NW, and NE directions. Thus, there is no edge contention in this algorithm.

Now, let us calculate the time complexity of each phase, which will be the same for each. In phase one, for example, since there are k steps required for each phase the start up time is kt_s. Now, we calculate the transmission time. At the first step, a node will receive two messages, one from the E and the other from SE adjacent nodes. At the j^{th} step, $j = 2,3, \dots, k-1$:

1. A node receives one message from the E adjacent node (this message is from the node at a distance j along the E direction).
2. A node receives j messages from the SE adjacent node (these messages are from nodes at a distance j from the receiving node - distance j_1 along the SE

direction and j_2 distance along the E direction, where $j_1 + j_2 = j, 0 \leq j_1, j_2 \leq j$).

3. A node receives $j - 1$ messages from the SW adjacent node (these messages are from nodes at a distance j from the receiving node - distance j_1 along the SW direction and distance j_2 along the SE direction, where $j_1 + j_2 = j, 1 \leq j_1, j_2 \leq j - 1$).

As Figure 3.9 shows, the $j - 1$ messages received from the SW adjacent node and the only message received from the E adjacent node, i.e. message sent by the node at a distance j along the SE direction, must be sent to the NE adjacent node. Similarly, the j messages received from the SE adjacent node and the one message received from the E adjacent node (a total of $j + 1$ messages) must be sent to the NW adjacent node. Furthermore, the message received from the E adjacent node must be sent to the W adjacent node.

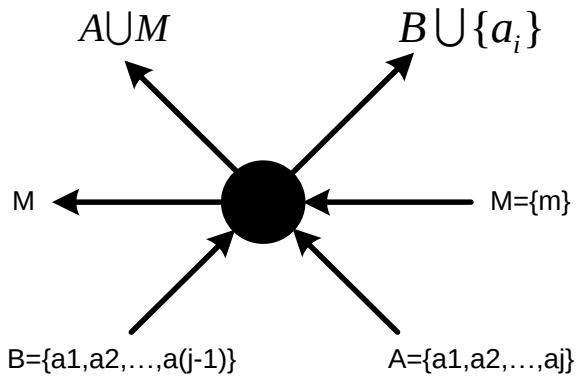

Figure 3.9: Message Broadcast in Step $\boldsymbol{i}, \boldsymbol{i} = \boldsymbol{2}, \boldsymbol{3}, .., \boldsymbol{k}$.

To summarize, at the j^{th} step of the algorithm, a node has to send $j+1$, j, and 1 messages to its NW, NE, and W adjacent nodes respectively, for $j = 1,2,\ldots,k-1$. Thus, in each step, the messages sent to the NW adjacent nodes take the maximum time. Therefore, the transmission time required is:

$$(1+2+3+\cdots+k)t_w w = t_w w\frac{k(k+1)}{2}$$

Since the algorithm uses three phases, the total time required is:

$$T_{all-to-all} = 3(t_s k + \left(t_w w\frac{k(k+1)}{2}\right)).$$

3.2.3 One-to-All Personalized Communication

The one-to-all personalized communication is different from one-to-all broadcast in that a node starts with $N-1$ distinct messages, where N is the number of nodes, and each of the messages has to be sent to different nodes. In other words, a single source node sends a distinct message to every other node in the network.

Initially, the source node (node 0) contains all the messages, and each message is identified by the labels of its destination node. In the first communication step, the source node divides the messages into six groups according to the six k-triangles and then sends each group of messages to the node responsible for that k-triangle. In each of these triangles, every node follows the same broadcast pattern used in the one-to-all communication except that it first combines all the messages needed for the next step, and then sends this long message.

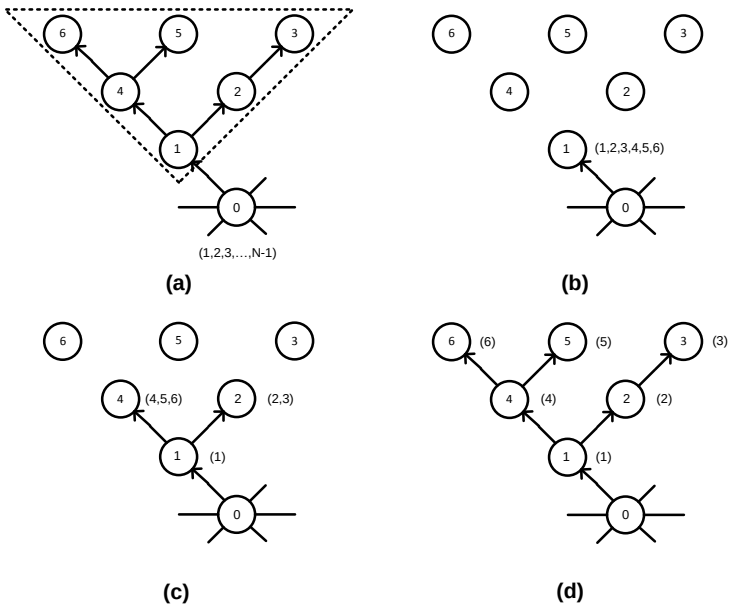

Figure 3.10: One-to-all Personalized Communication Broadcast.

Figure 3.10 shows the communication steps for the one-to-all personalized communication broadcast in a single triangle, which has six nodes. In the starting phase (a), the source node contains all the messages. In the first step (b), the source node divides the messages into six groups and then sends the six messages associated with the NW k-triangle. In step two (c), node 1 keeps its message, divides the remaining messages into two groups according to the routing algorithm used in one-to-all broadcast, and sends them to its adjacent nodes in this k-triangle. Finally, in step three (d), nodes 4 and 2 keep their messages and send the other messages to their next neighbors. Thus, after step three all the nodes have received their messages.

In this algorithm, no node receives a duplicate message. Besides, because the network is node symmetric, any node can be a source node, and thus this algorithm is universal for every node in the network. The time required for this broadcast can be

calculated as follows. In the j^{th} k-triangle, the number of messages to be sent along the ρ^j dimension is more than or equal to the messages sent along the ρ^{j-1} dimension. So we will calculate the time required to send along the ρ^j dimension. The number of messages sent along the ρ^j dimension at step t is $1 + 2 + \cdots + k - t$ for $t = 0,1,2, \ldots, k-1$. Thus, the number of messages sent is:

$$= \sum_{i=1}^{k} \frac{i(i+1)}{2} = \frac{1}{2}\sum_{i=1}^{k}(i^2 + i) = \frac{1}{2}\left(\sum_{i=1}^{k} i^2 + \sum_{i=1}^{k} i\right)$$

$$= \frac{1}{2}\left(\frac{k(k+1)(2k+1)}{6} + \frac{k(k+1)}{2}\right) = \frac{k(k+1)}{4}\left(\frac{2k+1}{3} + 1\right) = \frac{k(k+1)(k+2)}{6}.$$

Therefore, the required time for this communication is:

$$T_{one-to-all\ personalized} = t_s k + t_w w \frac{k(k+1)(k+2)}{6}.$$

3.2.4 All-to-All Personalized Communication

In the all-to-all personalized communication, each node sends a distinct message to every other node in the network. More clearly, every node performs a one-to-all personalized communication. This communication is one of the most expensive operations in terms of communication complexity. This pattern is useful in applications like fast Fourier transform, matrix transpose, and some parallel database join operations.

The communication patterns of the all-to-all personalized communication are identical to those of the all-to-all broadcast. Only the size and the contents of messages

are different. The algorithm idea is to combine the all-to-all and the one-to-all personalized communications together.

Again, this algorithm uses three phases, similar to the all-to-all communication algorithm. Initially, each node contains all the messages, the labels of their destination nodes identified within the messages. In phase one, each node uses only two edges to send the messages in order to avoid the conflict of sending and receiving the messages at the same time through an edge. In the first communication step, each node divides the messages according to the six k-triangles and then sends two groups of messages to two adjacent nodes, say W and NW adjacent nodes. These are responsible for sending messages to their corresponding k-triangles. In each of these two k-triangles, each node follows the same broadcast pattern used in the one-to-all broadcast, except that it sends all the messages needed for the next step as a group in the same way as in the one-to-all personalized communication pattern. Similarly, the same occurs in phases two and three, except that each node sends the messages to two different adjacent nodes (and so two different k-triangles) in each phase. In each phase, every node sends its messages to two of the six triangles and so by the end of the phase three, every node sends its messages to the entire network.

Similar to what we have shown in the case of the all-to-all communication algorithm, in the first step of the algorithm, a node can receive from its E and SE adjacent nodes. From next step onwards, a node receives messages from only its E, SE, and SW adjacent nodes and sends to its W, NW, and NE adjacent nodes. Thus, there is no edge conflict in this algorithm.

Steps	Direction	Number of messages
Step 1	W	1+2+...+k
	NW	1+2+...+k
	NE	0
Step 2	W	1+2+...+(k-1)
	NW	(1+2+...+(k-1))+(k-1)
	NE	k-1
Step 3	W	1+2+...+(k-2)
	NW	(1+2+...+(k-2))+2(k-2)
	NE	2(k-2)
...	...	...
Step j	W	1+2+...+(k-j+1)
	NW	(1+2+...+(k-j+1))+(k-j+1)(j-1)
	NE	(k-j+1)(j-1)

Table 3.1: The Number of Messages Sent to the W, NW, and NE Adjacent Nodes in Phase One.

Now, we calculate the time taken by the algorithm. Table 3.1 shows the size of the messages sent along W, NW, and NE adjacent nodes in each step. From this table, it is clear that the data transfer time is the maximum for sending data to the NW adjacent node of a given node. Thus, the total number of data transfer is as follows:

$$T_{data-transfer} = \sum_{j=1}^{k}(1 + 2 + \cdots + (k - j + 1)) + \sum_{j=1}^{k}(k - j + 1)(j - 1)$$

$$= \sum_{j=1}^{k}\frac{j(j+1)}{2} + \sum_{j=1}^{k}(k - j + 1)(j - 1)$$

$$= \sum_{j=1}^{k}\frac{j^2}{2} + \sum_{j=1}^{k}\frac{j}{2} + \sum_{j=1}^{k}((k + 1)j - (k + 1) - j^2 + j)$$

$$= \sum_{j=1}^{k}\frac{j^2}{2} + \sum_{j=1}^{k}\frac{j}{2} + \sum_{j=1}^{k}(k + 2)j - \sum_{j=1}^{k}(k + 1) - \sum_{j=1}^{k}j^2$$

$$= \sum_{j=1}^{k}\left(k + 2 + \frac{1}{2}\right)j - \sum_{j=1}^{k}\frac{j^2}{2} - \sum_{j=1}^{k}(k + 1)$$

$= \frac{(2k+5)}{2} \sum_{j=1}^{k} j - \sum_{j=1}^{k} \frac{j^2}{2} - k(k+1)$

$= \frac{k(2k+5)(k+1)}{4} - \frac{1}{2}\left(\frac{k(k+1)(2k+1)}{6}\right) - k(k+1)$

$= \frac{k(k+1)(2k+1)}{6}$.

Thus, the total time complexity of the algorithm is:

$$T_{all-to-all\ personalized} = 3\left(t_s k + t_w w \frac{k(k+1)(2k+1)}{6}\right).$$

3.3 Resource Placement in Eisenstein-Jacobi Graphs

As we explained before, a parallel system may contain a limited amount of resources, and these resources need to be distributed so that all nodes in the system can access them uniformly. One such resource placement method is the t-embedding. The t-embedding, or t-dominating, set problem is defined as a resource placement that grants every node a resource within a distance of t, and the resource nodes are separated by at least $2t + 1$.

Some solutions to this problem for the EJ network are given in [25][29] with $\alpha = a + b\rho$, where $\rho = \frac{-1+i\sqrt{3}}{2}$ and the norm of $\alpha, N(\alpha) = a^2 + b^2 - ab$. In this case, the number of nodes in the EJ network is $a^2 + b^2 - ab$. Here, we have given a result assuming $\alpha = a + b\rho$, where $\rho = \frac{1+i\sqrt{3}}{2}$ and the norm of $\alpha, N(\alpha) = a^2 + b^2 + ab$. Thus, the number of nodes in our case is $a^2 + b^2 + ab$, whereas, the number of nodes in [25][29] as mentioned before is $a^2 + b^2 - ab$; in some cases the number of nodes

can be equal. For example, if $a, b > 0$ for the method proposed here and one of a or b is less than 0 in the method proposed in [25][29], then we get the same number of nodes.

3.3.1 The t*-Dominating Set Problem*

As described earlier, the problem of perfect t-dominating sets has been previously considered for other graphs [28][6]. The concepts of domination and perfect dominating set are defined as follows:

Definition 3.10: A vertex u of a graph G is said to t-dominate another vertex v if $D(u, v) \leq t$, where D denotes the graph distance. Then, a vertex subset $S \subset V$ is called a perfect t-dominating set if every vertex of G is t-dominated by a unique vertex in S.

In this section, we study the existence of perfect t-dominating sets over Eisenstein-Jacobi graphs. Such sets are ideal of $\mathbb{Z}[\rho]_\alpha$. This problem is directly related to the designing t-error correcting perfect codes in the ring of Eisenstein-Jacobi integers using the Eisenstein-Jacobi graph distance metric. Let us first introduce some necessary definitions.

Given $0 \neq \alpha = a + b\rho \in \mathbb{Z}[\rho]$, an integer $t > 0$, and a vertex $\mu \in E_\alpha$, a ball of radius t centered in μ embedded in EJ_α is the set:

$$B_t(\mu) = \{\gamma \in EJ_\alpha | D_\alpha(\gamma, \mu) \leq t\}.$$

For any β, the cardinality of $B_t(\beta)$ is $1 + \sum_{d=1}^{t} 6d = 3t^2 + 3t + 1$.

A subset S of the vertices in EJ_α is called a perfect dominating set if every node in EJ_α is t-dominated by exactly one node in S. If S is a t-dominating set, then the set $S' = \{\mu | \mu = \beta + \alpha, \alpha \in S\}$ is also a t-dominating set.

An ideal I in the quotient ring $\mathbb{Z}[\rho]_\alpha$ is a subset of $\mathbb{Z}[\rho]_\alpha$ such that the following rules hold:

- For any $\gamma, \mu \in I$, then $\gamma - \mu \in I$.
- For any $\mu \in \mathbb{Z}[\rho]_\alpha$ and any $\gamma \in I$, then $\gamma\mu \in I$.

The ideal generated by an element $\beta \in \mathbb{Z}[\rho]$ is denoted as (β). This means every element in (β) is a multiple of β. The following theorem gives the main result of this section.

Theorem 3.11: Let $\alpha = a + b\rho$ and t be a positive integer; then,

i) If $\beta = t + (t+1)\rho$ divides α then $S = (\beta) \subseteq \mathbb{Z}[\rho]_\alpha$ is a perfect t-dominating set in EJ_α.

ii) If $\beta' = (2t+1) - (t+1)\rho$ divides α then $S = (\beta') \subseteq \mathbb{Z}[\rho]_\alpha$ is a perfect t-dominating set in EJ_α .

iii) If $\beta'' = -(2t+1) + t\rho$ divides α then $S = (\beta') \subseteq \mathbb{Z}[\rho]_\alpha$ is a perfect t-dominating set in EJ_α .

Proof: First, note that S has exactly $\frac{\mathcal{N}(\alpha)}{\mathcal{N}(\beta)}$ elements [29]. Next, we prove the first part of the theorem. The second and the third parts can be proved using a similar argument. We need to prove that, if $\beta_1, \beta_2 \in S$, then $D_\alpha(\beta_1, \beta_2) \geq 2t+1$, where D_α denotes the Eisenstein-Jacobi graph distance.

Consider $\beta_1, \beta_2 \in S$. Then $\beta_1 = \alpha_1\beta$ and $\beta_2 = \alpha_2\beta$, with $\alpha_1, \alpha_2 \in \mathbb{Z}[\rho]$. We have to prove that $D_\alpha(\beta_1, \beta_2) = D_\alpha(\alpha_1\beta, \alpha_2\beta) = D_\alpha\big((\alpha_1 - \alpha_2)\beta, 0\big) \geq 2t+1$. Note that it is enough to prove that, for any $\in \mathbb{Z}[\rho]$, $D_\alpha(\mu\beta, 0) \geq 2t+1$.

Suppose $D_\alpha(\mu\beta, 0) \leq 2t$. This means that there exists $x, y,$ and z such that $\mu\beta = x + y\rho + z\rho^2 \ (mod\ \alpha)$, with $|x| + |y| + |z| \leq 2t$. Note that $N(\mu\beta) = N(\mu)N(\beta)$. Now $(\beta) = 3t^2 + 3t + 1$, and $N(x + y\rho + z\rho^2) = N\big((x - z) + (y + z)\rho\big)$

$= (x - z)^2 + (y + z)^2 + (x - z)(y + z)$

$= x^2 + y^2 + z^2 + yz + xy - xz$

$\leq (|x| + |y| + |z|)^2 \leq 4t^2$, since $|x| + |y| + |z| \leq 2t$.

As $N(\mu)N(\beta) \leq 4t^2$ then $N(\mu) \leq \frac{4t^2}{3t^2+3t+1} < 2$ for $t > 0$. This implies μ is a unit, i.e. $\mu = \{\pm 1, \pm\rho, \pm\rho^2\}$. Since $\mu \in \{\pm 1, \pm\rho, \pm\rho^2\}$, $D_\alpha(\mu\beta, 0) = D_\alpha(\beta, 0) = 2t+1$, and this leads to a contradiction. Thus, the distance between any two resource nodes is at least $2t+1$. Furthermore, the number of elements in S is $\frac{N(\alpha)}{N(\beta)}$, the set S is a perfect t-dominating set.

For the second part of the theorem,

$\beta' = (2t+1) - (t+1)\rho = (2t+1)(\rho + \rho^5) - (t+1)\rho$

$= (2t+1)\rho^5 + t\rho = (2t+1)\rho^5 + (1-\rho^5)$

$= (t+1)\rho^5 + t = (t+1)\rho^5 + t\rho^6.$

Thus, by Theorem 3.6, $D_\alpha(\mu\beta', 0) = 2t+1$.

Similarly, for the third part,

$\beta'' = -(2t+1) + t\rho = (-2t-1) + t\rho = t(-1+\rho) - (t+1) = t\rho^2 + (t+1)\rho^3,$

and hence, by Theorem 3.6, $D_\alpha(\mu\beta'', 0) = 2t+1$.

■

Example 3.12: Let $\alpha = 5 + 6\rho$. Since $1 + 2\rho$ divides $5 + 6\rho$, the ideal $S = \{1 + 2\rho, -4 - \rho, -3 + \rho, -2 + 3\rho, -1 + 5\rho, 5 - 4\rho, -5 + 4\rho, 1 - 5\rho, 2 - 3\rho, 3 - \rho, 4 + \rho, -1 - 2\rho, 0\}$ is a perfect 1-dominating set in $EJ_{5+6\rho}$. Figure 3.11 illustrates this example. The wrap-around links of $EJ_{5+6\rho}$ have been omitted for the sake of clarity.

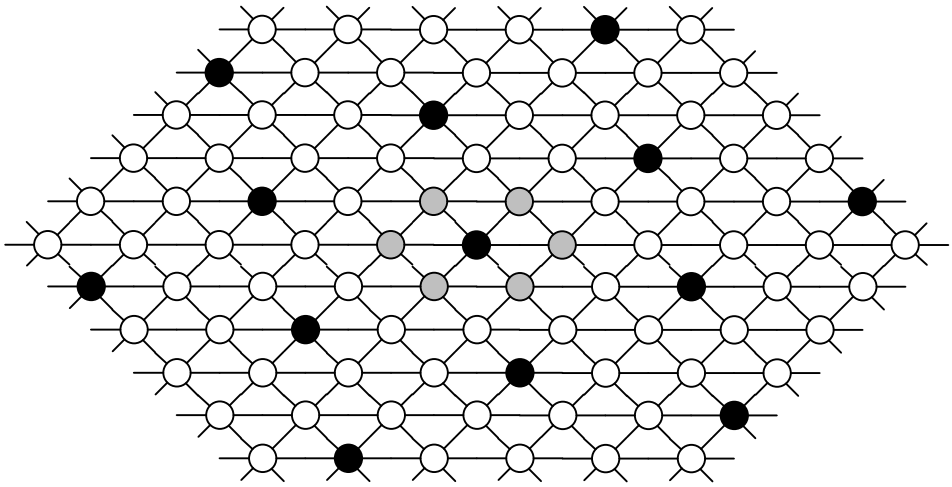

Figure 3.11: Perfect 1-Dominating Set in $\boldsymbol{EJ_{5+6\rho}}$.

Table 3.2 gives some t-dominating generators β for a given networks generated by α.

α	β	t
$-1+5\rho$	$1+2\rho$	1
$5-\rho$	$3-2\rho$	1
$-4-\rho$	$-3+\rho$	1
$-1+8\rho$	$2+3\rho$	2
$8-\rho$	$5-3\rho$	2
$-7-\rho$	$-5+2\rho$	2
$-1+11\rho$	$3+4\rho$	3
$11-\rho$	$7-4\rho$	3
$-10-\rho$	$-7+3\rho$	3

Table 3.2: Some Values for $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ where $\boldsymbol{\beta}$ Divides $\boldsymbol{\alpha}$.

3.4 Conclusion

In summary, we presented communication algorithms for the Eisenstein-Jacobi networks when $\alpha = k + (k+1)\rho$. In all proposed algorithms, the number of steps required to complete the algorithm is minimum. Let t_s be the startup time to send a packet, t_w be the time to transfer one word, and w be the size of the packet. When k is the network diameter, the number of steps and time required for each communication pattern are as follows:

- One-to-all broadcast required k steps and $(t_s + t_w w)k$ time.
- All-to-all broadcast required $3k$ steps and $3(t_s k + \left(t_w w \frac{k(k+1)}{2}\right))$ time.
- One-to-all personalized broadcast required k steps and

  $t_s k + t_w w \frac{k(k+1)(k+2)}{6}$ time.
- All-to-all personalized broadcast required $3k$ steps and

  $3\left(t_s k + t_w w \frac{k(k+1)(2k+1)}{6}\right)$ time.

In all these algorithms, no node receives redundant messages. Furthermore, the proposed algorithms are optimal or close to optimal, at least in terms of the number of steps required.

In [25][29], a solution to t-dominating set problem is given for EJ network with $\alpha = a + b\rho$, assuming $\rho = \frac{-1+i\sqrt{3}}{2}$ and the norm of $\alpha, N(\alpha) = a^2 + b^2 - ab$. In this section, we have given another solution for the t-dominating set problem for EJ

networks with $\alpha = a + b\rho$, where $\rho = \frac{1+i\sqrt{3}}{2}$ and the norm of , $N(\alpha) = a^2 + b^2 + ab$, which gives the total number of nodes in the network as $a^2 + b^2 + ab$.

Chapter 4

Future Research

There are some open research problems related to the topics of this thesis. In this chapter, the problems are divided into three sections. Section 4.1 introduces the problem of finding three edge disjoint Hamiltonian cycles in Eisenstein-Jacobi networks. In Section 4.2, we describe the j-adjacency placement problem in Eisenstein-Jacobi networks. Finally, in Section 4.3, the problem of broadcasting in general Eisenstein-Jacobi networks is described.

4.1 Edge Disjoint Hamiltonian Cycles in Eisenstein-Jacobi Networks

Chapter 2 of this thesis provided a solution for generating two edge disjoint Hamiltonian cycles in Gaussian networks when $\gcd(a,b) = d > 1$, and so the edge disjoint Hamiltonian cycle in the Gaussian network is now completely solved. On the other hand, for Eisenstein-Jacobi networks, the problem is solved when $\gcd(a,b) = 1$ in [15][3], but it is not clear how to solve this problem when $\gcd(a,b) = d > 1$.

Studying and analyzing the relationship between the real dimension and the imaginary dimension helped us to construct the edge disjoint Hamiltonian cycles for the Gaussian networks. However, in the case of the Eisenstein-Jacobi network, similar approaches seemed too complicated because these networks are of degree six.

Our preliminary investigation indicates that, starting from some node, if we proceed along the $1, \rho,$ or ρ^2 dimension, we get a cycle of length $\frac{a^2+b^2+ab}{d}$. It is not clear how to combine these smaller cycles to form Hamiltonian cycles in such a way that the edges form three edge disjoint Hamiltonian cycles.

4.2 Broadcasting in Eisenstein-Jacobi Networks in the General Case

Chapter 3 introduced four primitive communication patterns: one-to-all, all-to-all, one-to-all personalized, and all-to-all personalized for Eisenstein-Jacobi networks when $\alpha = k + (k+1)\rho$. Designing efficient algorithms for general Eisenstein-Jacobi networks are challenging research topics.

4.3 J-adjacency in Eisenstein-Jacobi Networks

Resource placement problems are divided into three classes. The first class of the problem is called the distance-t problem. This problem considers placing resources such that each node in the network either has a copy of the resource or is at a distance of at most t from at least one node having a copy of the resource. The second class of the problem is called the j-adjacency problem. This problem considers placing resources such that each nonresourse node is adjacent to j resource nodes. The third class of the problem is the generalized placement problem, which is a combination of the distance-t and the j-adjacency problems.

In chapter 3, we provided a solution to the distance-t problem for the Eisenstein-Jacobi network and solutions for the other two classes of problems are not yet known.

Bibliography

[1] F. Aguil´o, M.A. Fiol and C. Garc´ıa. “Triple Loop Networks with Small Transmission Delay”. Discrete Mathematics 167/168 (1997) pp. 3-16.

[2] Bader Almohammad, Bella Bose: Fault-Tolerant Communication Algorithms in Toroidal Networks. FTCS 1998: 186-194, 1998.

[3] Madhusudhanan Anantha. *Gray codes and their applications*. PhD thesis, School of EECS, Oregon State University, 2007.

[4] Madhusudhanan Anantha, Bella Bose, Bader F. AlBdaiwi: Mixed-Radix Gray Codes in Lee Metric. IEEE Trans. Computers 56(10): 1297-1307 (2007).

[5] M. Bae, B. Bose: Edge Disjoint Hamiltonian Cycles in k-Ary n-Cubes and Hypercubes. IEEE Trans. Computers 52(10): 1271-1284 (2003)

[6] M. Bae and B. Bose, Resource Placement in Torus-Based Networks, *IEEE Transactions on Computers*, Vol. 46, No. 10, pp. 1083-1092, October 1997.

[7] R. Beivide, C. Martínez, E. Vallejo, J. Gutierrez, C. Izu, “Gaussian Interconnection Networks”. Spanish Parallelism Conferences, Sept. 05, Granada, Spain.

[8] R. Beivide, E. Herrada, J.L. Balcázar and A. Arruabarrena. "Optimal Distance Networks of Low Degree for Parallel Computers". IEEE Transactions on Computers, Vol. C-40, No. 10, pp. 1109-1124, 1991.

[9] E. R. Berlekamp. “Algebraic Coding Theory”. Aegean Park Press, 1984.

[10] Dimitri P. Bertsekas , John N. Tsitsiklis, *Parallel and distributed computation: numerical methods*, Prentice-Hall, Inc., Upper Saddle River, NJ, 1989

[11] B. Bose, B. Broeg, Y. Known and Y. Ashir. ”Lee Distance and Topological Properties of k-ary n-cubes”. IEEE Transactions on Computers,Vol. 44, No. 8, pp. 1021-1030, 1995.

[12] Z. Cvetanovic. Performance analysis of the Alpha 21364-based HP GS1280 multiprocessor. In *Proceedings of the 30th Annual International Symposium on Computer.*

[13] J. Duato, S. Yalamanchili, and L. Ni, Interconnection Networks: An Engineering Approach. IEEE Computer Society Press, 1997.

[14] M.A. Fiol, J.L. Yebra, I. Alegre and M. Valero. "A Discrete Optimization Problem in Local Networks and Data Alignment". IEEE Transactions on Computers, Vol. 36, No. 6, pp. 702-713, 1987.

[15] M. Flahive and B. Bose, The Topology of Gaussian and Eisenstein-Jacobi Interconnection Networks". Submitted to IEEE Transitions on Parallel and Distributed Systems.

[16] E. Gabidulin, C. Mart´ınez, R. Beivide and J. Gutierrez. “On the Weight Distribution of Gaussian Graphs with an Application to Coding Theory”. Proceedings of the 8th International Symposium on Communication Theory and Applications (ISCTA’05). Ambleside, UK. 17-22 July 2005.

[17] G.H. Hardy, E.M. Wright, "An introduction to the theory of numbers" , Oxford Univ. Press (1979).

[18] K. Huber. “Codes over Eisenstein-Jacobi integers”. Finite fields: theory, applications, and algorithms (Las Vegas, NV, 1993), 165–179, Contemp. Math., 168, Amer. Math. Soc., Providence, RI, 1994.

[19] K. Huber. “Codes Over Gaussian Integers”. IEEE Transactions on Information Theory, Vol. 40, No. 1, pp. 207-216, January 1994.

[20] K. Huber. “Codes over Tori”. IEEE Trans. on Information Theory, Vol. 43, No. 2, pp. 740-744, March 1997.

[21] J. H. Jordan and C. J. Potratz, Complete residue systems in Gaussian integers, MATHEMATICS MAGAZINE, 38 (1965) 1-12.

[22] V. Kumar et al., “Introduction to Parallel Computing: Design and Analysis of Parallel Algorithms,” Addison-Wesley Pub Co., January 1994.

[23] C. Martìnez, R. Beivide, C. Izu and J. Gutierrez. “Distance-Hereditary Embeddings of Circulant Graphs”. Proc. of IEEE Int. Conf. on Information Technology: Coding and Computing (ITCC-2003), pp. 320-324. Las Vegas, April 2003.

[24] C. Martìnez, R. Beivide, J. Gutierrez and E. Gabidulin. “On the Perfect t-Dominating Set Problem in Circulant Graphs and Codes over Gaussian Integers”. Proceedings of the 2005 IEEE International Symposium on Information Theory (ISIT’05). Adelaide, Australia. September, 2005.

[25] C. Martìnez, Beivide, R., and Gabidulin, E., Perfect Codes for Metrics Induced by Circulant Graphs, *IEEE Trans. Inform. Theory*, 2007, vol. 53, no. 9, pp. 3042–3052.

[26] C. Martìnez, M. Moret´o, R. Beivide, E. Gabidulin and E. Stafford. “Mesh Interconnection Networks Modeling Through the Ring of the Gaussian Integers”. Submitted to IEEE Transanctions on Computers.

[27] C. Martìnez, M. Moreto, R. Beivide, E. Gabidulin and E. Stafford. Modeling Toroidal Networks with the Gaussian Integers. IEEE Transactions on Computers, vol. 57, no. 8, August 2008.

[28] C. Martìnez, Moret´o, M., Beivide, R., and Gabidulin, E., A Generalization of Perfect Lee Codes over Gaussian Integers, in *Proc. 2006 IEEE Int. Sympos. on Information Theory (ISIT’06), Seattle, USA*, 2006, pp. 1070–1074.

[29] C. Martìnez, E. Stafford, R. Beivide, E. M. Gabidulin, Modeling hexagonal constellations with Eisenstein-Jacobi graphs, Journal: Problems of Information Transmission,ISSN: 0032-9460, Vol: 44, Issue: 1,2008.

[30] C. Martìnez, E. Stafford, R. Beivide and E. Gabidulin. "Perfect Codes over Eisenstein-Jacobi Graphs". Proc.Thenth International Workshop on Algebraic and Combinatorial Coding Theory. Zvenigorod, Russia, September 03-09 2006.

[31] C. Martìnez, E. Stafford, R. Beivide and E. Gabidulin. "Perfect Codes in Metrics Induced by Eisenstein-Jacobi Integers". Problems of Information Transimission, Russian Publication, 2008.

[32] C. Martìnez, E. Stafford, R. Beivide and E. Gabidulin. "Perfect Codes over Lipschitz Integers". Submitted to 2007 IEEE International Symposium on Information Theory.

[33] C. Martìnez, E. Vallejo, R. Beivide, C. Izu and M. Moret´o. "Dense Gaussian Networks: Suitable Topologies for On-Chip Multiprocessors". International Journal of Parallel Programming (ISSN - 0885-7458) Vol: 34 pp. 193-211, June 2006. Springer Netherlands.

[34] K. G. Shin. "HARTS: A Distributed Real-Time Architecture." IEEE Computer, vol. 24, no. 5, pp 25-35 (1991).

[35] S. L. Scott and G. M. Thorson. The Cray T3E network: Adaptive routing in a high performance 3D torus. In *Proceedings of HOT Interconnects, IV*, August 1996.

[36] R. Rowley and Bose. On the number of arc-disjoint Hamiltonian circuits in the DeBruijn graphs. Parallel Processing Letters, pages 375-380, 1993.

[37] C. K. Wong and D. Coppersmith. "A Combinatorial Problem Related to Multimodule Memory Organizations". Journal of the ACM, Vol. 21, No. 3, pp. 392-402, 1974.