Surface Areas of Some Interconnection Networks

Fatemeh Salahi

Submitted in partial fulfillment
of the requirements for the degree of

Master of Science

Department of Computer Science
Brock University
St. Catharines, Ontario

Abstract

An interesting property of an interconnected network (G) is the number of nodes at distance i from an arbitrary processor (u), namely the node centered surface area which is denoted by $B_{G,u}(i)$. This is an important property of a network due to its applications in various fields of study. In this research, we investigate on the surface area of two important interconnection networks, (n,k)-arrangement graphs and (n,k)-star graphs. Abundant works have been done to achieve a formula for the surface area of these two classes of graphs, but in general, it is not trivial to find an algorithm to compute the surface area of such graphs in polynomial time or to find an explicit formula with polynomially many terms in regards to the graph's parameters. Among these studies, the most efficient formula in terms of computational complexity is the one that Portier and Vaughan proposed in [25] which allows us to compute the surface area of a special case of (n,k)-arrangement and (n,k)-star graphs when $k = n-1$, in linear time which is a tremendous improvement over the naive solution with complexity order of $O(n \times n!)$. The recurrence we propose here has the linear computational complexity as well, but for a much wider family of graphs, namely $A(n,k)$ for any arbitrary n and k in their defined range. Additionally, for (n,k)-star graphs, we prove properties that can be used to achieve a simple recurrence for its surface area.

Acknowledgements

I would like to express my deep and sincere gratitude to my supervisor, Dr. Ke Qiu for all his support, patience and encouragement throughout this research. His guidance has helped me in all the time of this process and without him this thesis would not have been completed or written. Besides my advisor, I would like to thank the members of my supervisory committee, Dr. B. Ombuki-Berman, Dr. S. Houghten and Dr. Eddie Cheng for their time and insightful comments.

I also would like to thank the Department of Computer Science, Brock University, for providing me with financial means and for the opportunity I've been given to study and work in such a friendly environment.

Finally, I am grateful to my amazing family for their unconditional love and constant support. This accomplishment would not have been possible without them. Thank you.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Parallel Computing

Two major approaches to solving computational problems are sequential processing and parallel processing. In the earlier approach a single processor is used to perform defined tasks in a given order to obtain the final solution. In the former approach, multiple processors are working on a given problem simultaneously by breaking the problem into several subproblems and solving each of these smaller problems in a different processor. Processors may communicate with each other to share partial results. The final outcome is then obtained by combining the partial results of each processor. Clearly, the idea of parallelism is not only to solve problems in a more efficient way and reduce the waiting time to perform tasks, but also it makes solving some particular problems possible as these problems regardless of the time it takes to be solved, are impossible to be tackled sequentially. Figure 1.1 demonstrates the architecture of parallel computation.

One of the classifications of parallel computing is based on their communication method which can be shared-memory or interconnection network.

In this chapter, first we have a review of computational classification

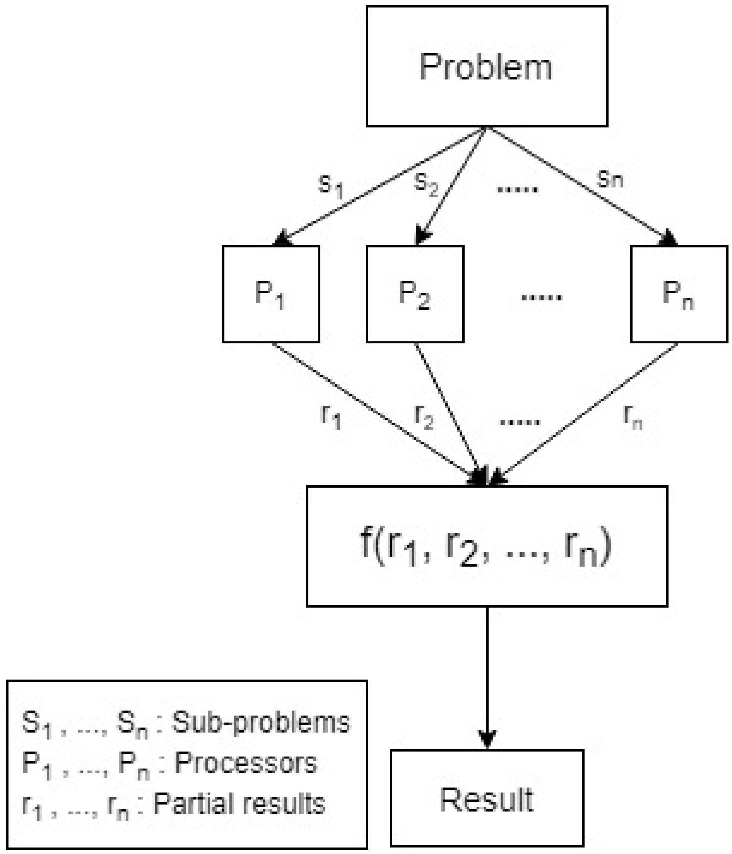

Figure 1.1: Parallel computing architecture

of parallel programming, then, we introduce the terminology that is used throughout this research. We will then introduce the problem of computing the surface areas of interconnection networks and its applications. And finally, we describe the organization of this thesis.

1.2 Parallel Computational Models

Formally, a computational model is a coherent collection of mechanisms for communication, synchronization, partitioning, placement and scheduling. In this thesis, we consider computational models based on how processors communicate with each other which classifies models into shared-memory and interconnection networks.

Three main components of digital systems are logic, memory and communication. Nowadays, the performance in most digital systems is limited by their communication or interconnection and not by their logic or memory and this reveals the importance of studying interconnection networks.

1.2.1 Shared-memory Networks

Shared-memory computing or parallel random access machine (PRAM) consists of the following elements:

- N identical processors $P_1, P_2, ..., P_N$. In theory N is unlimited but in practice N is an arbitrary large yet finite number.
- A shared memory with M locations. Likewise, M is unbounded in principle but in practice it is an arbitrary large finite number such that $M \geq N$.
- A memory access unit (MAU) that allows the processors to gain access to the memory and based on the way they get this access, PRAM is classified into four sub-models:
  - Exclusive Read, Exclusive Write (EREW), in which no two processors have access to the same location of the memory at the same time.
  - Exclusive Read, Concurrent Write (ERCW), in which no two processors can read from the same location of memory at the same time, but multiple processors can write to the same location at the same time.
  - Concurrent Read, Exclusive Write (CREW), in which processors can read from the same location of the memory at the same time but they can not write to the same location.

– Concurrent Read, Concurrent Write (CRCW), in which both reading from and writing to the same location of the memory at the same time for multiple processors is allowed.

Figure 1.2 shows the structure of a shared-memory computer.

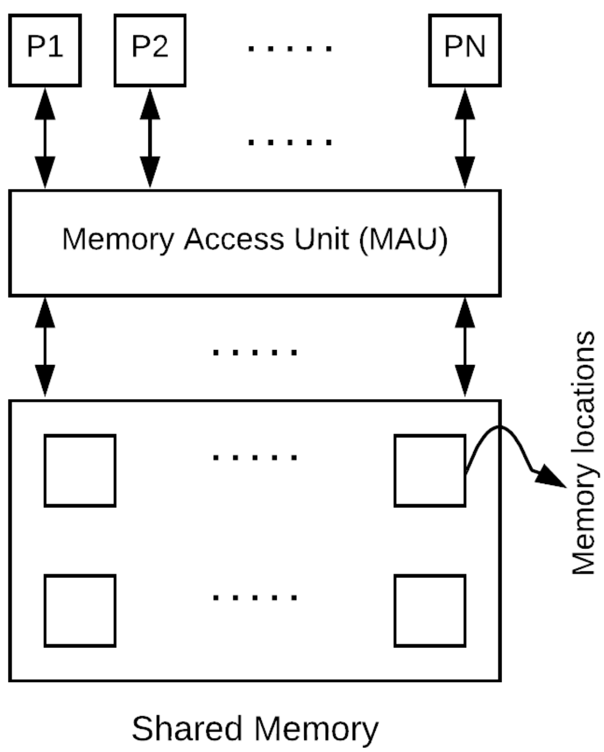

Figure 1.2: Shared memory scheme

1.2.2 Interconnection Networks

Another important model for parallel computers is Interconnection Networks in which instead of having a common shared memory as in

PRAM, processors have their own dedicated memory and they are communicating with each other by links that are connecting processors to each other. Two processors are called neighbors if they are directly connected by a link. Two processors that are neighbors can communicate simultaneously. Hence, each link between two processors represents two links in two directions from one processor to the other and vice versa. For simplicity, in all figures here, one undirected line between two processors represents two-way communication. The way these connections are constructed suggests different topologies for interconnection networks. However, all these topologies can be modeled as a graph $G = (V, E)$ in which V is the set of nodes (processors) and E is the set of edges (links) such that $E \subseteq V \times V$. Thus, standard graph terminology and related algorithms will be used in this study and we will review them briefly in Section 1.3. Interconnection network model is shown in Figure 1.3.

An important question one may ask while designing an interconnection network is what topology should the network have? In other words, how many neighbors a processor should have and does all processors have the same number of neighbors and which processors should be neighbors? The other key factor to choose a satisfactory network is the time it takes to transfer a message between two arbitrary processors. Moreover, are the paths static or dynamic? namely are the paths established by the designer beforehand or is the algorithm flexible in choosing the paths? Answering above questions assists in choosing a decent network for one's requirements. There are a number of works studying different networks (graphs) and their properties to analyze their cost and performance regarding network requirements [4], [16], [18], [19], [20]. One of the interesting structures is the hypercube and its variants. We will overview these networks in Section 1.4, but we will first review the parameters of an interconnection network and some graph terminology

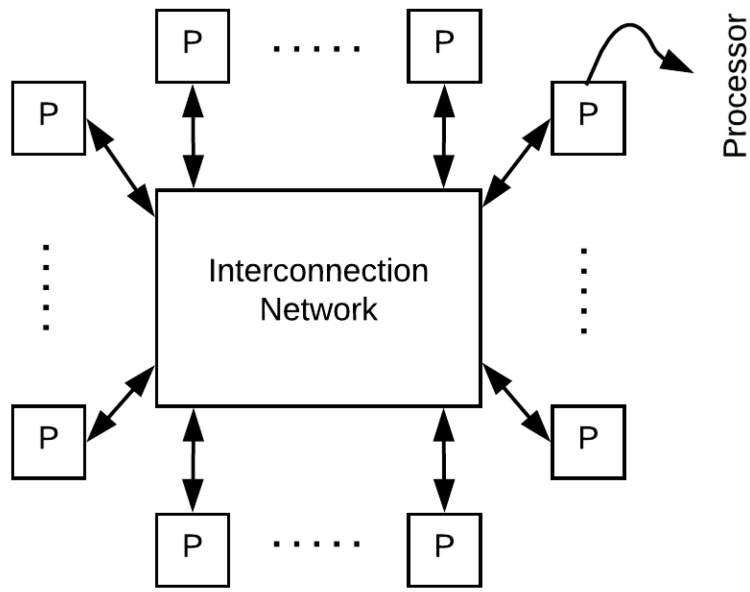

Figure 1.3: Interconnection networks scheme

that are used throughout this study.

1.2.2.1 Parameters of Interconnection Networks

To understand the requirements of designing an interconnection network, it is necessary to contemplate network parameters. Some of these parameters are:

- The number of terminals or processors
- How processors interact with each other
- The peak bandwidth and the average bandwidth of each terminal
- The required latency
- The expected traffic pattern

- The message size
- The desired quality of the service
- The desired reliability and availability of the interconnected network

For our purpose of study, we drop factors focusing on hardware parameters and message exchanging. In this research, we are interested in the number of terminals and how they are connected and we study different topologies for such connections, only emphasizing on network topology, i.e. the number of processors and how they are connected to each other.

1.3 Terminology

Definition 1.1. Complete graph. A simple, undirected graph $G(V, E)$ is complete if every pair of distinct nodes are adjacent. A complete graph with n nodes is indicated by K_n.

Definition 1.2. Path. A path in an undirected graph between two nodes v_1 and v_n, is a finite sequence of nodes $P = (v_1, v_2, ..., v_n) \in V \times V \times ... \times V$ such that v_i and v_{i+1} are adjacent for $1 \leq i < n$. Such a path P between v_1 and v_n is of length $n-1$ indicated with $|P| = n-1$.

Definition 1.3. Shortest path. If $P = \{P_1, P_2, ..., P_k\}$ is the set of all paths between two nodes v_1 and v_2, then the shortest path between these two nodes is $P_i \in P$ such that $|P_i| \leq |P_j|$ for $1 \leq j \leq k$.

Definition 1.4. Connected (disconnected) graph. A graph is connected if there is a path between any pair of nodes and is disconnected otherwise.

Definition 1.5. Distance. The distance between two nodes in a graph is the number of edges in the shortest path between these two nodes.

Definition 1.6. Diameter(Ecentricity). The diameter or ecentricity of a connected graph G is the greatest distance over all pairs of nodes and is denoted by $D(G)$.

Definition 1.7. Connectivity. Connectivity is the minimum number of nodes that need to be removed to make the graph disconnected and it is a metric of graph (network) resilience.

Definition 1.8. Degree of a node. In a graph $G(V, E)$, the degree of a node v is the number of nodes in G that are adjacent to v and it is denoted by $deg(v)$.

Definition 1.9. Degree of a graph. The degree of a graph $G(V, E)$, $(V = \{v_1, v_2, \cdots, v_n\})$ is defined as largest degree of its nodes and is denoted by $deg(G)$. In other words,

$$deg(G) = \max\{deg(v_1), deg(v_2), \cdots, deg(v_n)\}.$$

Definition 1.10. Regular graph. A graph is regular if all its nodes have the same degree. More precisely, Graph $G(V, E)$, $(V = \{v_1, v_2, \cdots, v_n\})$ is called regular of degree k (or $k-$regular) if $deg(v_i) = k$, for all $1 \leq i \leq n$.

Definition 1.11. Automorphism. An automorphism of a graph $G = (V, E)$ is a permutation π of the vertex set V, such that for each pair of nodes (u, v), u and v are connected if and only if the pair $\pi(u)$ and $\pi(v)$ are also connected.

Definition 1.12. Symmetric graph. A graph $G = (V, E)$ is node symmetric if given any two pairs of adjacent nodes (u_1, v_1) and (u_2, v_2) of G, there is an automorphism $\pi : V(G) \to V(G)$ such that $\pi(u_1) = \pi(u_2)$ and $\pi(v_1) = \pi(v_2)$. Similarly, a graph is edge symmetric if for any pair of edges e_1 and e_2 in $E(G)$, there is an automorphism that maps e_1 to e_2.

Definition 1.13. Undirected and directed graph. A graph is undirected if all the edges are bidirectional. In contrast, a graph where the edges point in a direction is called a directed graph.

Definition 1.14. Tree and its common expressions. In graph theory, a **tree** is an undirected graph in which there is exactly one path between any two nodes. Every node in a tree (except one) is connected by an edge from exactly one other node. This node is called a **parent**. The only node that has no parent is called **root**. Each node can be connected to arbitrary number of nodes, which are called **children**. Nodes with no children are called **leaves**, or external nodes. Nodes which are not leaves or root are called **internal nodes**. Nodes with the same parent are called **siblings**.

Definition 1.15. Cost factor. Cost factor is defined as diameter $\times$ degree of a node and it is a good criterion to measure the performance of a network [3].

1.4 Review of Some Interconnection Networks

In this section, we will review some of the popular interconnection network topologies.

1.4.1 Complete Graph

The most obvious way to connect N processors is to connect each pair by a two-way link. Therefore, each processor has $N - 1$ neighbors and can send a message directly to or receive a message directly from any other processor. From this point of view a complete network is the most powerful network, however, it is expensive or in some cases impossible to be implemented due to high number of links. In graph theory such topology is a complete graph and it is denoted by K_n where

n is the number of nodes. Clearly, K_n is $(n-1)$-regular and node (edge) symmetric.

1.4.2 Linear Array and Ring

One simple topology to link N processors is to consider all processors in a line and connect each processor to those next to them. In other words, processors are interconnected in a one dimensional array and each of them is connected to two other processors except the first and the last one which are only connected to one processor. This topology is called a linear array. As a special case of linear array, one can connect the first and the last processors to each other and make a ring. A ring is 2-regular and node (edge) symmetric.

1.4.3 Mesh and Torus

A mesh (or lattice or grid) topology is a two dimensional array such that each processor is connected to its four neighbors except the ones on the boundaries. More precisely, an $m \times n$ mesh has $m \times n$ number of processors that are placed in m rows and n columns such that the processor in row i and column j ($1 < i < m$ and $1 < j < n$), $P_{i,j}$ is connected to $P_{i-1,j}$, $P_{i+1,j}$, $P_{i,j-1}$ and $P_{i,j+1}$. The connection for boundary nodes is trivial considering they are only connected by links inside the grid. For example, $P_{1,j \neq n}$ is connected to $P_{2,j}$, $P_{1,j-1}$ and $P_{1,j+1}$. A 4×3 mesh is shown in Figure 1.4.

A torus is a special case of mesh, when the first and the last node in each row and column are connected to each other. A torus is 4-regular and symmetric.

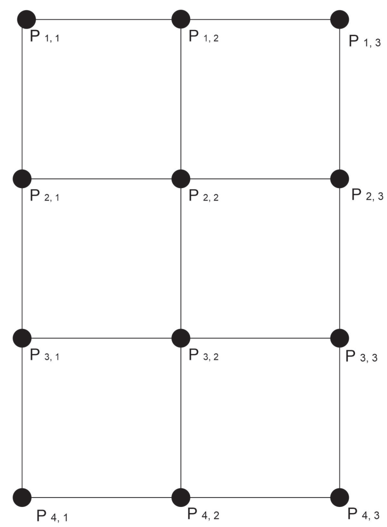

Figure 1.4: A 4×3 mesh

1.4.4 Full Binary Tree

A full binary tree is a tree in which each node has exactly two children. A node v is in level l if there are l edges between the root and v. A full binary tree has interesting properties. To name one, if we define the height of the tree as the number of edges from the root to the deepest leaf, then the full binary tree provides the best possible ration between the number of nodes and the height. It can easily be proved that the height of a full binary tree with n nodes is at most $O(\log n)$. A full binary tree with 3 levels is shown in Figure 1.5

1.4.5 Hypercubes

Hypercubes are simple yet interesting structure for an interconnection network. A hypercube of n dimension has $N = 2^n$ nodes for some $n \geq 0$. Each node (processor) is represented by a binary string of length n and two nodes (processors) are connected if their binary representations differ in exactly one bit. A hypercube with n dimensions is represented as Q_n. Figure 1.6 shows the hypercube structure of dimensions 3.

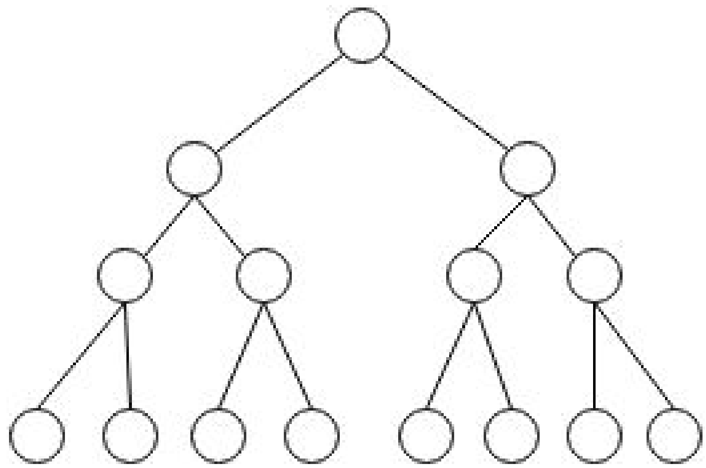

Figure 1.5: A full binary tree with 3 levels

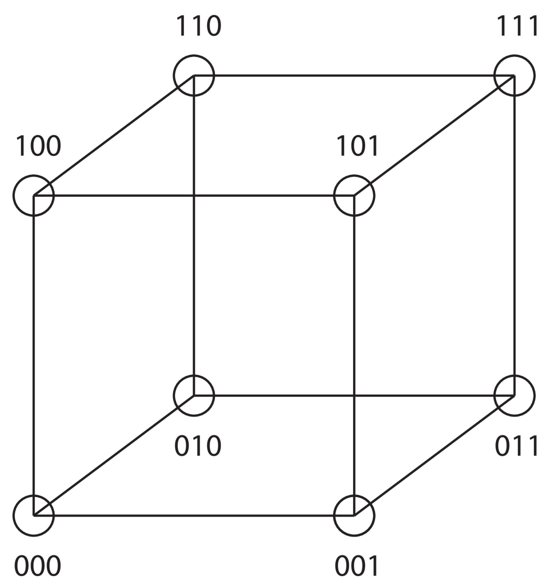

Figure 1.6: A hypercube of dimension 3

1.4.5.1 Properties of Hypercubes

Hypercubes have a number of interesting properties that make them a competitive candidate in interconnection networks based on performance and practicality metrics. Some of these properties are:

- **Symmetry**. The hypercubes are node and edge symmetric (node and edge transitive respectively). This implies that for some problems such as routing algorithm the source node can be any node. Here, we always assume the source node is $\overbrace{00\cdots0}^{n}$, except saying otherwise.

- **Node degree**. Every node in Q_n has n neighbors. In other words, Q_n is n-regular.

- **Connectivity**. In a hypercube of dimension n there are n disjoint paths between each pair of nodes. In additions, node degree is uniformly distributed in a hypercube and each node is connected to n nodes, therefore, the connectivity of Q_n is n.

- **Distance**. The length of a shortest path, namely the distance between two nodes u and v in Q_n is the number of bits their binary representations differ which is denoted by $H(u, v)$ and is called the Hamming distance.

- **Diameter**. The diameter of a n-dimensional hypercube is n, simply because the maximum value of the Hamming distance among all node pairs is at most n. It is interesting to notice that the relationship between the diameter and number of nodes in Q_n (2^n) is logarithmic, which is an important property of hypercubes.

- **Recursive structure**. Hypercubes can be constructed by starting with two connected nodes. If more nodes are required, the structure is duplicated and interconnected by adding links between the original and the duplicated nodes [4]. More formally, the n-cube can be constructed recursively as the Cartesian Product of Q_{n-1} and K_2. This can be highly functional in utilizing recursive approaches to solve Q_n problems.

Some other properties of hypercubes are: Q_n has 2^n nodes and $n2^{n-1}$ edges. Also, if the shortest path between two nodes in Q_n is P, then

there are $|P|$ parallel (disjoint) paths of length $|P|$ between them.
By considering hypercube structure one may notice that this architecture suffers from scalability issues as moving from dimension d to $d+1$ doubles the number of processors.

1.4.6 Folded Hypercubes

The folded hypercube is a standard hypercube with some extra edges between its nodes. In other words, a folded hypercube of dimension n, $FHC(n)$ is constructed from a Q_n, by linking each node to the unique node that is farthest from it [16]. A folded hypercube of degree 3, $FHC(3)$ is shown in Figure 1.7.

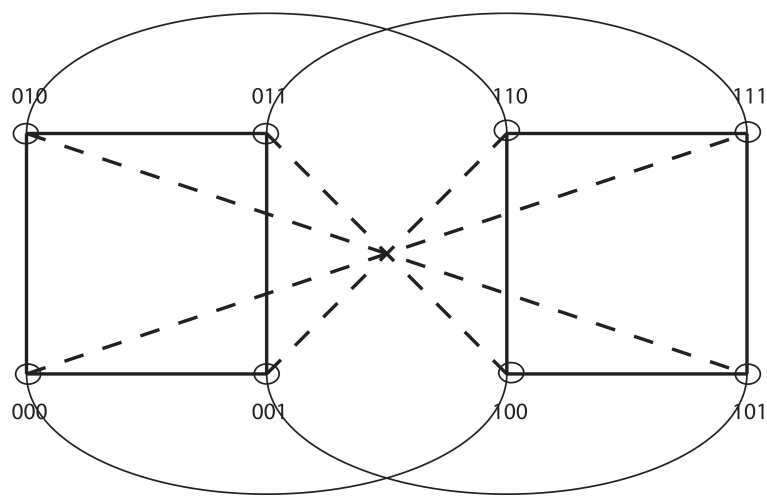

Figure 1.7: A folded hypercube of degree 3

1.4.6.1 Properties of Folded Hypercubes

1) $FHC(n)$ is regular of degree $n+1$ and the number of node disjoint paths between any two nodes in $FHC(n)$ is $n+1$.
2) $FHC(n)$ has 2^n nodes and $(n+1)2^{n-1}$ edges and the diameter is $\lceil n/2 \rceil$.
3) It is node symmetric and it has a hierarchical structure as proved in

[16], given an $FHC(n)$ and an integer $0 < k < n$, an $FHC(k)$ can be constructed out of nodes and edges in $FHC(n)$.

1.4.7 Augmented Cubes

As an enhancement on the hypercube Q_n, Choudum and Sunitha [5] proposed augmented cubes, AQ_n, to not only keep some of the favorable properties of Q_n but also possess some embedding properties that Q_n does not. For example, AQ_n contains cycles of all lengths from 3 to 2^n, but Q_n contains only even cycles [6].

Definition 1.16. The augmented cube denoted by AQ_n of dimension n has 2^n nodes. Each node is represented by an n-bit binary string. Two nodes $v = v_1 v_2 \cdots v_n$ and $u = u_1 u_2 \cdots u_n$ are adjacent to each other if and only if there exists an integer k, $1 \leq k \leq n$ such that either:

1) Their binary representations only differ in bit k,

2) Their binary representations are exactly the same for the first $k-1$ bits, and for $k \leq i \leq n$ the binary representation of one node is the complement of the other one.

Figure 1.8 illustrates augmented cubes of dimension 3.

1.4.7.1 Properties of Augmented Cubes

1) Node and edge symmetry

2) Can be defined recursively [5] as: AQ_1 is a complete graph K_2. For $n \geq 2, AQ_n$ is obtained by taking two copies of the augmented cube AQ_{n-1}, and adding $2 \times 2^{n-1}$ edges between the two as follows:

Let's define $V(AQ^0_{n-1})$ as the set of nodes with the first bit of 0. Similarly, $V(AQ^1_{n-1})$ is set of nodes starting with 1. A node from $V(AQ^0_{n-1})$ is joined to a node from $V(AQ^1_{n-1})$ if either:

- $u_i = v_i \ for \ 1 \leq i \leq n-1$; or

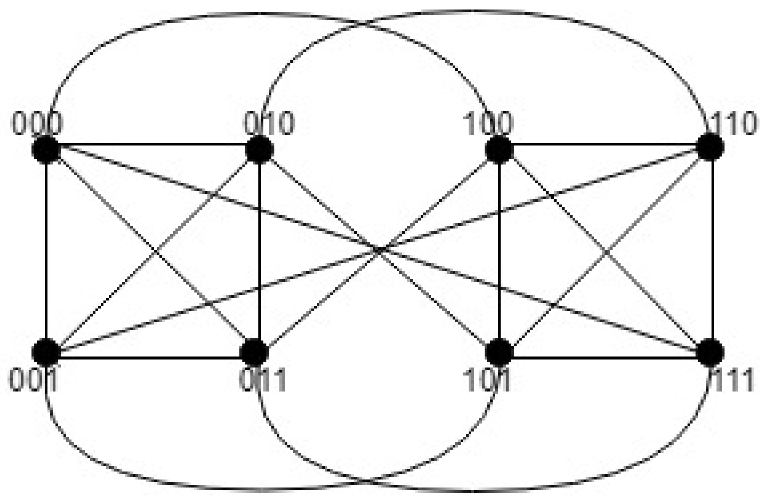

Figure 1.8: An augmented cube of dimension 3

- $u_i = \overline{v_i}$ for $1 \leq i \leq n-1$

3) Regular of degree $2n-1$ and the connectivity is also $2n-1$.
4) The diameter of AQ_n is $\lceil n/2 \rceil$.

1.4.8 Star Graphs

As a competitor of hypercubes, the star graph was introduced in [7]. Each node in a star graph S_n is a permutation of n symbols from $\{1, 2, ..., n\}$ and can be represented as $v(1)v(2)...v(n)$ such that $v(i) \neq v(j)$ for all $i \neq j$, and therefore, S_n has $n!$ nodes. Two nodes are connected if their first and i-th symbols are replaced with each other. Thus each node is connected to $n-1$ nodes. Although the star graph has many attractive properties inherited from hypercubes due to its node and edge symmetry, it still suffers from scaling problem as one may face with either too many or too few processors [8]. Figure 1.9 shows a star graph of dimension 4.

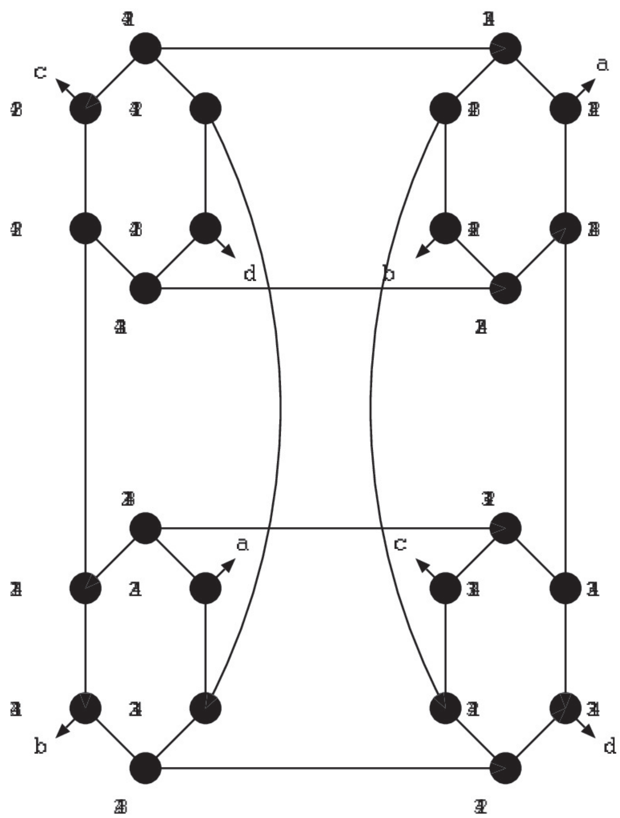

Figure 1.9: A star graph with 4 dimensions, $S(4)$

1.4.8.1 Properties of Star Graphs

Star graphs have inherited many of interesting properties of hypercubes. For instance, S_n is node (edge) symmetric; it is $(n-1)$-regular graph and has the connectivity $n-1$; and like the hypercube, it has a recursive structure. However, finding the distance between two arbitrary nodes in S_n is not as simple and straightforward as in Q_n. There are some algorithms that find this parameter in S_n such as those proposed in [9, 10, 11]. For a similar number of nodes, the star graph has a lower node degree, a shorter diameter, and a smaller average distance than the comparable hypercube [12]. An important drawback of S_n is the restriction on the number of nodes which has to be $n!$.

1.4.9 (n,k)-Star Graphs

The (n,k)-star graph denoted by $S(n,k)$, is a generalized form of star graph that has addressed the problem of scalability in star graphs [8]. Each node in an (n,k)-star graph is a string of length k and its symbols are from $\{1,2,...,n\}$ and can be represented as $v(1)v(2)...v(k)$ such that $v(i) \neq v(j)$ for all $i \neq j$. Thus (n,k)-star has $\frac{n!}{(n-k)!}$ nodes. Two nodes $v = v(1)v(2)...v(k)$ and $u = u(1)u(2)...u(k)$ are connected if either of the following conditions holds:

1) $v(1) = u(i)$ and $u(1) = v(i)$ for some $2 \leq i \leq k$ and $v(j) = u(j)$ for all $2 \leq j \leq k$ and $j \neq i$.

2) $v(1) \neq u(1)$ and $v(j) = u(j)$ for all $2 \leq j \leq k$.

1.4.9.1 Properties of (n,k)-Star Graphs

1) $S(n,k)$ has $\frac{n!}{(n-k)!}$ nodes and $\frac{n!(n-1)}{2(n-k)!}$ edges, and it is node (but not edge) symmetric.
2) It is $(n-1)$-regular and its connectivity is $(n-1)$.
3) It has a hierarchical structure such that the graph can be partitioned into n subgraphs that each partition is isomorphic to an $S(n-1,k-1)$. One way to conduct this decomposition is based on the last symbol of the nodes. In other words, $S(n,k)$ can be decomposed into n partitions as $A_1(n-1,k-1), A_2(n-1,k-1), ..., A_n(n-1,k-1)$ such that in $A_i(n-1,k-1)$ there are only nodes whose last symbols are i. Figure 1.10 shows the hierarchical structure of $S(4,2)$ which is constructed of four $(3,1)$-star graphs.
4) The diameter of $S(n,k)$ is $\begin{cases} 2k-1, & 1 \leq k \leq \lfloor \frac{n}{2} \rfloor \\ k + \lfloor \frac{n-1}{2} \rfloor, & \lfloor \frac{n}{2} \rfloor + 1 \leq k \leq n-1 \end{cases}$

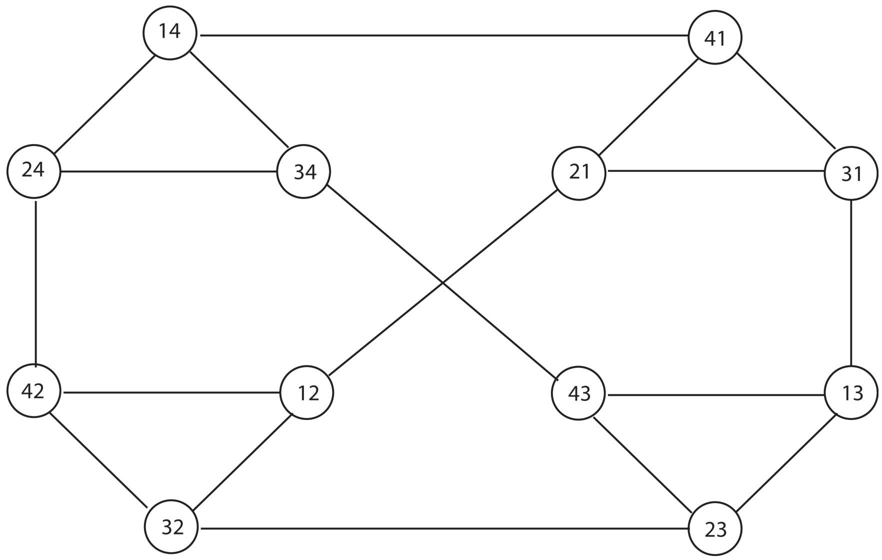

Figure 1.10: Hierarchical structure of S(4,2)

1.4.10 (n, k)-Arrangement Graphs

Besides (n, k)-star graphs, the arrangement graph is one of the major classes of graphs that was introduced to address the problem of scalability of hypercubes [8].

Definition 1.17. An arrangement graph is denoted by $A(n, k) = (V, E)$ in which,

$V = \{v_1 v_2 ... v_k \mid v_i \in \{1, 2, ..., n\}\ and\ v_i \neq v_j\ for\ i \neq j\}$ and

$E = \{(u, v) \mid u, v \in V\ and\ for\ some\ i \in \{1, 2, ..., k\}\ v_i \neq u_i\ and\ v_j = u_j\ forj \neq i\}$.

In other words, nodes in an $A(n, k)$, are arrangements of k elements out of n elements, and two nodes are connected if they differ in exactly one position. Figure 1.11 demonstrates a $(4, 2)$-arrangement graph.

1.4.10.1 Properties of (n, k)-Arrangement Graphs

Like Q_n and S_n, $A(n, k)$ is node and edge symmetric and has a hierarchical structure. In this study, we can partition $A(n, k)$ similar to the method that was described in Section 1.4.9 for $S(n, k)$ which is based

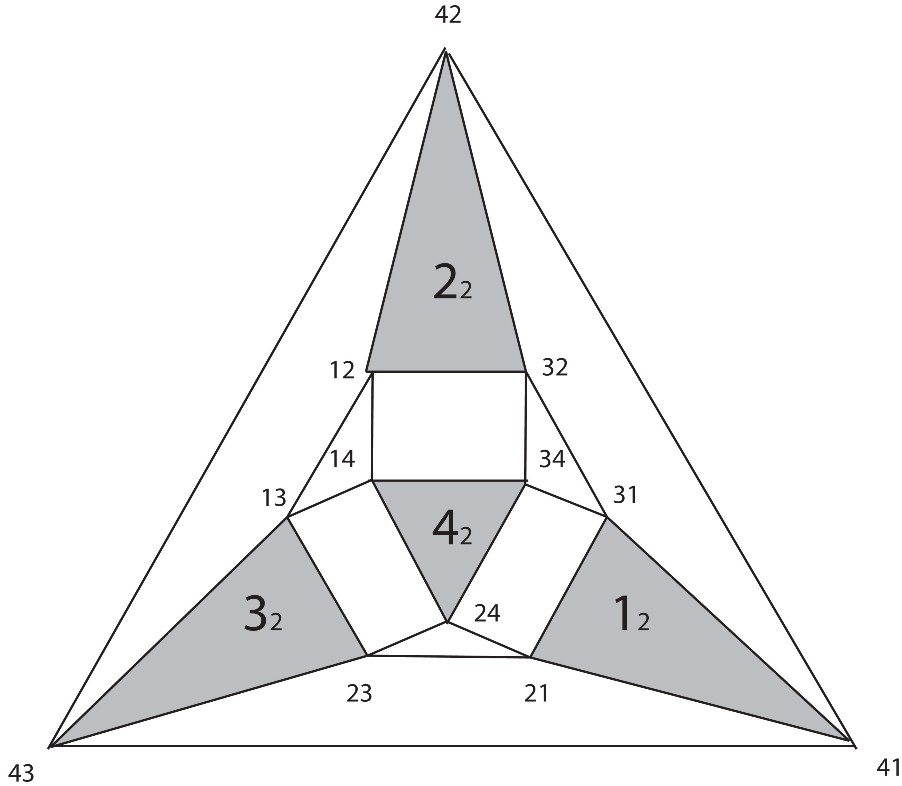

Figure 1.11: A $(4, 2)$-arrangement graph

on last symbol of node representation. Figure 1.11 shows this heirarchical structure of a $(4, 2)$-arrangement graph. $A(n, k)$ is a regular graph of degree $k(n-k)$ with $\frac{n!}{(n-k)!}$ nodes and diameter $\lfloor \frac{3k}{2} \rfloor$. It has been shown [13] that $A(n, n-1)$ is isomorphic to the S_n.
In Table 1.1 a brief comparison of some of the aforementioned interconnection networks is provided.

1.5 Shortest Path Routing in (n, k)-Star and (n, k)-Arrangement Graphs

Throughout this research we are focused on distances between nodes. Therefore, we need an efficient routing algorithm for these networks. In this section, we are reviewing the approach that was introduced in [21] and [24] to use cyclic structure of (n,k)-star and (n,k)-arrangement graphs respectively in order to compute the distance between two arbitrary nodes in these graphs. Before describing the routing problem, note that due to node symmetry in (n,k)-star and (n,k)-arrangement graphs any node can be embedded to the identity node, $e = 12...(k-1)k$. Therefore, all the paths between any two arbitrary nodes $\mathcal{V}$ and $\mathcal{U}$ are isomorphic to all the paths between $\mathcal{V}$ and e, since for these graphs there is a mapping function $\mathcal{M}$ that embeds $\mathcal{U}$ to e. Hence, without loss of generality in the routing problem for these graphs we always assume that the destination node is e.

Definition 1.18. Internal and external symbols. In the node representation of $v = v_1v_2...v_n$ in (n,k)-star and (n,k)-arrangement graphs, v_i is an internal symbol if $v_i \in \{1,2,...,k\}$ and it is external if $v_i \in \{(k+1),(k+2),...,n\}$.

Definition 1.19. Invariant symbols. Symbol v_i is called invariant when it is in its correct position in the node representation, namely $v_i = i$.

If we denote an internal cycle by C_i and an external cycle by C'_i and the number of symbols in an internal (external) cycle by m_i (m'_i), the construction of cycle representation of node $v = v_1v_2...v_k$ in an (n,k)-star and (n,k)-arrangement graph is as follows:

(1) For each external symbol $x_{m'_i}$ in v, construct an external cycle as $C'_i = (x_1, x_2, ..., x_{m'_i})$ such that the desired position of x_j in v is held by x_{j+1} for $1 \leq j \leq m'_i - 1$.

(2) Construct an internal cycle as $C_i = (x_1, x_2, ..., x_{m_i})$ such that the position of x_{j+1} in v is desired by x_j for $1 \leq j \leq m_i - 1$.

(3) If we have α internal and β external cycles, the cycle representation of node v is $C(v) = C_1C_2...C_\alpha C'_1C'_2...C'_\beta$. Note that in

cycle representation we are not concerned of the order of cycles. Additionally, invariant symbols are ignored as they are already in the correct position.

The routing from v to e then is solved by correcting the position of symbols in these cycles as follows:

(1) Correction of an internal cycle $C_i = (x_1, x_2, ..., x_{m_i})$:

(1.1) if $v_1 = x_1$, move x_1 to its correct position which is held by x_2. Then move x_2 to its correct position which is held by x_3 and continue to x_{m_i-1}. Note that in this case x_{m_i} is corrected as a result of correcting x_{m_i-1}.

(1.2) otherwise, move the x_1 to the first position, then correct the position of $x_1, x_2, ..., x_{m_i}$ same as 1.1.

(2) Correction of an external cycle $C'_i = (x_1, x_2, ..., x_{m'_i})$: The correction in an external cycle is the same process as (1) with the difference that whenever the external symbol has moved to the first position we exchange it with the desired symbol whose desired position is held by the external symbol that is in the first position now.

We now demonstrate cycle representation and its correction through an example. In an $(9,7)$-star graph, consider the node $v = 9765132$. v has two internal cycles $C_1 = (7,2)$ and $C_2 = (6,3)$, and one external cycle $C' = (9,1,5,4)$ and the cyclic representation is $(7,2)(6,3)(9,1,5,4)$. We first correct the internal cycles:
The $C_1C_2 = (7,2)(6,3)$ is corrected through the path:
$9765132 \rightarrow 7965132 \rightarrow 2965137 \rightarrow 9265137 \rightarrow 6295137 \rightarrow 3295167 \rightarrow 9235167$.
At this point the position of symbols $2, 3, 6$ and 7 is corrected. Now we correct the external cycle:

Table 1.1: Structural comparison of some interconnection networks

Graph	# of nodes	# of edges	Node degree	Diameter	Connectivity	Cost factor
Q_n	2^n	$n2^{n-1}$	n	n	n	$O(n^2)$
$FHC(n)$	2^n	$(n+1)2^{n-1}$	$n+1$	$\lceil n/2 \rceil$	$n+1$	$O(n^2)$
AQ_n	2^n	$2^{n-1}(2n-1)$	$2n-1$	$\lceil n/2 \rceil$	$2n-1$	$O(n^2)$
S_n	$n!$	$\frac{n!(n-1)}{2}$	$n-1$	$\lfloor 3(n-1)/2 \rfloor$	$n-1$	$O(n^2)$
$S(n,k)$	$\frac{n!}{(n-k)!}$	$\frac{n!(n-1)}{2(n-k)!}$	$n-1$	$2k-1; if\ 1 \leq k \leq \lfloor n/2 \rfloor$ $k + \lfloor \frac{n-1}{2} \rfloor$; $Otherwise$	$n-1$	$O(kn)$
$A(n,k)$	$\frac{n!}{(n-k)!}$	$\frac{kn!}{2(n-k-1)!}$	$k(n-k)$	$\lfloor \frac{3k}{2} \rfloor$	$k(n-k)$	$O(k^2 n)$

$9235167 \rightarrow 4235167 \rightarrow 5234167 \rightarrow 1234567$.

Therefore, with a path of length 9 we traveled from v to e. In [21] Chiang *et al.* showed that the path achieved by this method is a shortest path and therefore, the distance from any node to identity node is computed based on the cycle correction. If we denote the total number of cycles of a node $v = v_1 v_2 ... v_k$ (including external cycles) by c and z stands for the number of external symbols in v and m be the total number of misplaced symbols of v, then the distance from node v to the identity node e in an (n, k)-star graph is

$$d(v) = \begin{cases} c + m + z & if v_1 = 1 \\ c + m + z - 2 & otherwise \end{cases}$$

Using the above formula for previous example, $c = 3$, $m = 7$ and $z = 1$ and since $v_1 \neq 1$, then $d(v) = 3 + 7 + 1 - 2 = 9$.

Similar to (n, k)-star graphs, the cycle representation for a node v in (n, k)-arrangement graphs is defined as $C(v) = C_1 C_2 ... C_\alpha C'_1 C'_2 ... C'_\beta$ where α and β are the number of internal and external cycles respectively. Using the same definitions for c and m, the distance from the node v to the identity node in an (n, k)-arrangement graph is shown to be [24]:

$$d(v) = c + m - 2\beta$$

1.6 Organization of the Thesis

In the next chapter, we first discuss about one of the interesting properties of interconnection networks, namely the surface area. Then, we will study this property for hypercubes and some other interconnection networks in Chapter 2. In Chapter 3, we discuss a property of functions which is important in regards of achieving a closed form formula, namely forward difference of order n. In Chapter 4, we propose a general formula to compute the surface area of (n, k)-arrangement graphs. We also show a number of considerable properties for another important interconnection network, (n, k)-star graphs, that can be used to obtain a recursive function to compute surface area of this class of graph. Finally, we conclude this study in Chapter 5.

Chapter 2

Surface Areas: A Literature Review

2.1 Definitions

An interesting property of an interconnected network is the number of nodes at distance i from an arbitrary processor. Similarly, in graph theory given a node v of a connected graph G, how many nodes in G are at distance i from v in which $0 \leq i \leq D(G)$. In this study, we refer to this quantity as surface area with radius i centered at v as used in [27], however, it has been called with other titles in different literatures such as the Whitney numbers of the second kind of a poset in [25] and the distance distribution of nodes in [26].
There are some generalizations for this concept. For instance, instead of node centered one can define a subgraph centered surface area.
Definition 2.1. The distance from a node u to a subgraph H in G is denoted by $d_G(u, H)$ and is defined as the minimum distance from u to any other node in H. In other words, $d_G(u, H) = \min_{v \in H}\{d(u, v)\}$.
Definition 2.2. H-centered surface area of a graph G, $B_G(H, i)$ is

equal to:

$$\begin{cases} |V(H)| & \text{when } i = 0 \\ |\{d_G(u, H) = i, \text{for } u \notin H\}| & \text{when } i \in \{1, 2, ..., D(G)\} \end{cases}$$

As a special case of H-centered surface area one can consider H as a length l path and define the length l path surface area of a graph as follows:

Definition 2.3. Given a length l path $p_l(u_1, u_2, ..., u_{l+1})$ in a graph G, the number of nodes that are at distance i from p_l is called length l surface area and is denoted by $B_G^{p_l}(u_1, u_2, ..., u_{l+1}, i)$.

In [28] the length two surface area for (n, k)-star graphs is studied.

Another variant of surface area of a graph is introduced by E. Cheng *et al.* in [30]:

Definition 2.4. The number of nodes at distance i from a given edge (u, v) in a graph G is called edge centered surface area and is denoted by $B_G(u, v, i)$.

For the purpose of our research, we will study node centered surface area which is formally defined as follows:

Definition 2.5. The node centered surface area of a given graph G with diameter $D(G)$ and a given node u is the total number of nodes in G that are at distance i from u and it is denoted by $B_{G,u}(i)$.

Throughout this research, whenever we talk about surface area of a graph, we mean node centered surface area. Additionally, when we are considering symmetric graphs we drop the parameter u from $B_{G,u}(i)$ and simply refer to it as $B_G(i)$.

As an example, for the graph G in Figure 2.1, $B_{G,v1}(3)$ is 2 and $B_{G,v2}(1)$ is 3.

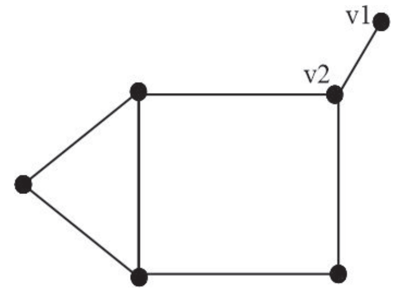

Figure 2.1: An example of surface area value for the graph G

2.2 Applications

Computing the surface area of a graph is a fundamental problem and abundant research has been done to derive a formula of this property for different networks (graphs) [8, 9, 15, 17, 25, 26, 27, 28, 30, 31, 32, 34, 35]. The surface area is an important property of a network as it can be used in evaluating network performance. It also provides valuable information in the study of genome rearrangements [29]. Some other applications mentioned in [15] include

- characterizing the $\mathfrak{t}$-neighborhood broadcasting behavior of a structure. $\mathfrak{t}$-neighborhood broadcasting is a term used when a node u in a network has to send a message to all nodes that are at distance less than or equal $\mathfrak{t}$ from u [22].

- deriving the average distance of a network structure which is the average length of a shortest path between all pairs of nodes.

- identifying spanning trees. A spanning tree of a graph G is a subset of G which is a tree that includes all nodes of G with minimum possible number of edges. The spanning tree has a huge range of applications in the fields of computer science, chemistry (for determination of the geometry and dynamics of compact polymers), medicine (identifying history of transmission of HCV infection), biology (in quantitative description of cell structures in light microscopic images), astronomy (to compare the aggregation of bright galaxies with faint ones), archeology (for identifying close proximity analysis), and many others [23].

- helping in resource placement in network structures
- derivation of the transmission of a graph defined as the sum of all distances.

2.3 Related Works

2.3.1 Surface Area of Hypercubes

As discussed in Section 1.4.5, hypercubes are node and edge symmetric. Using this fact and without loss of generality, let's assume that u in $B_{Q_n,u}(i)$ is $0^n = \overbrace{00\cdots0}^{n}$. Clearly, a node at distance i to node 0^n has exactly i 1's in its binary representation. Therefore, the number of nodes at distance i from 0^n is $B_{Q_n}(i) = \binom{n}{i}$.

2.3.2 Surface Area of Folded Hypercube

In [16], A. El-Amawy *et al.* have proved that

$$B_{FHC(n)}(i) = \begin{cases} \binom{n+1}{i} & 0 < i < \lceil n/2 \rceil \\ \binom{n+1}{n/2+1} & i = \lceil n/2 \rceil \text{ and } n \text{ is even} \\ \binom{n+1}{\lceil n/2 \rceil} & i = \lceil n/2 \rceil \text{ and } n \text{ is odd} \end{cases}$$

2.3.3 Surface Area of Augmented Cubes

In [15], Eddie Cheng *et al.* presented a recursive relationship for the surface area of augmented cubes as follows:

$$B_{AQ_n}(0) = 1$$
$$B_{AQ_n}(1) = 2n - 1$$
$$B_{AQ_n}(i) = B_{AQ_{n-1}(i)} + 2B_{AQ_{n-2}(i-1)},\ \ for\ i \geq 2.$$

By utilizing methods such as generating functions, they also have proposed the closed form formula for $B_{AQ_n}(i)$ as:

$$B_{AQ_n}(i) = \begin{cases} 1 & i = 0 \\ 2^{i-1}(\binom{n-i+1}{i} + \binom{n-i}{i}) & \text{otherwise} \end{cases}$$

2.3.4 Surface Area of Star Graphs

There are a number of studies on surface area of star graphs [9], [17], [25], [27], [31], [32]. Among these studies Portier and Vaughan [25] used an approach based on a generating function to derive a formula for computing surface area of star graphs. Later, Shen and Qiu showed a few errors in their derivation and provided a correction in [33]. Before we review their work a definition is required.

Definition 2.6. The (n, k) unsigned Stirling number of the first kind is denoted by $c(n, k)$ and is equal to the number of permutations of n symbols having exactly k cycles. The (n, k) signed Stirling number is denoted by $s(n, k)$ and its sign depends on the parity of n and k. In particular, $s(n, k) = (-1)^{n-k}c(n, k)$.

Based on [25] and [33], the surface area of star graph, $B_{S_n}(i)$, is given as follows:

$$B_{S_n}(i) = \sum_{k=0}^{L} \sum_{t=T_k}^{S_k} \binom{n-1}{k} \binom{n-1-k}{t} s(k+1, i-k+1-2t)(-1)^{i+2-t}$$

where $L = \min\{n-1, i+1\}$, $T_k = \max\{0, \lceil\frac{i-2k}{2}\rceil\}$ for $k \in [0, L]$, $S_k = \min\{n-1-k, \lfloor\frac{i+1-k}{2}\rfloor\}$ and $s(n, k)$ is the signed Stirling number of the first kind. Utilizing generating functions, they established then a recurrence for surface area of star graphs as:

$$B_{S_n}(0) = 1, \quad B_{S_n}(1) = n-1, \quad B_{S_n}(2) = (n-1)(n-2)$$

and for $i \geq 3$,

$$B_{S_n}(i) = B_{S_{n-1}}(i) + (n-1)B_{S_{n-1}}(i-1) - (n-2)B_{S_{n-2}}(i-1) + (n-2)B_{S_{n-2}}(i-3)$$

In another study, Ke Qiu *et al.*, have proved the following recursive relationship for $B_{S_n}(i)$ [9].

$$B_{S_n}(0) = 1, \quad B_{S_n}(1) = n-1, \quad B_{S_n}(2) = (n-1)(n-2)$$

and for $n \geq 1$ and $3 \leq i \leq D(S_n)$,

$$B_{S_n}(i) = (n-1)B_{S_{n-1}}(i-1) + \sum_{j=1}^{n-2} jB_{S_j}(i-3).$$

In Section 3.4, we will discuss the closed form formula that Ke Qiu *et al.* have presented for $B_{S_n}(i)$.

2.3.5 Surface Area of (n, k)-Arrangement and (n, k)-Star Graphs

In [34] Cheng *et al.* proposed an elementary counting approach to derive an explicit formula of the surface area for the (n, k)-arrangement graphs. The main contribution of their work is that this counting approach can be used to generalize a framework of deriving explicit formulas for the surface areas of other vertex symmetric graphs, as long as a distance formula can be derived in terms of the cycle structure of its nodes. They have used the same approach in [35] to propose an explicit formula for the surface area of (n, k)-star graphs.

We need the following notations before we review the formulas in [34] and [35]:

g_I: the number of internal cycles in cyclic structure

g_E: the number of external cycles in cyclic structure

b_I: the number of symbols contained in the respective g_I

b_E: the number of symbols contained in the respective g_E

b: total number of symbols in cyclic representation $(b_I + b_E)$

In an (n, k)-arrangement graph, for all $n \geq 2$ and $1 \leq k < n$ and $1 \leq i \leq D(A(n, k))$, we have $B_{A(n,k)}(i) =$

$$\sum_{g_E=0}^{n-k} \sum_{b_E=2g_E}^{k+g_E} \sum_{g_I=\max\{0,1-g_E\}}^{\lfloor i-b_E+g_E/3 \rfloor} \binom{n-k}{g_E} \binom{k}{b_E - g_E} \binom{k - b_E + g_E}{i - g_I + g_E - b_E}$$

$$\times (b_E - g_E)! \binom{b_E - g_E - 1}{g_E - 1} d(i - g_I + g_E - b_E, g_I).$$

where $d(n, k) = \sum_{j=0}^{n} (-1)^{n+k-j} \binom{n}{k} s(n - j, k - j)$.

For (n, k)-star graphs, for all $n \geq 2$ and $1 \leq k < n$ and $1 \leq i \leq D(S(n, k))$, we have $B_{S(n,k)}(i) =$

$$\sum_{g_I=\max\{1,i-k+1\}}^{\lfloor \frac{i}{3} \rfloor} \binom{k-1}{i-g_I} d(i - g_I, g_I) +$$

$$\sum_{g_I=\max\{1,i-k+2\}}^{\lfloor \frac{i+2}{3} \rfloor} \binom{k-1}{i - g_I + 1} d(i - g_I + 2, g_I) +$$

$$\sum_{g_E, g_I, b_E} \binom{n-k}{g_E} \binom{k-1}{i-g_I-g_E-1} \binom{i-g_I-g_E-1}{b_E-g_E} p(b_E - g_E, g_E) d(i - b_E - g_I - 1, g_I) +$$

$$\sum_{g_E, g_I, b_E} \binom{n-k}{g_E} \binom{k-1}{i-g_I-g_E} \binom{i-g_I-g_E+1}{b_E-g_E} p(b_E - g_E, g_E) d(i - b_E - g_I + 1, g_I).$$

where $p(m, b)$ is defined as:

$$\begin{cases} 1 & m = b = 1 \\ m!\binom{m-1}{b-1} & m \geq 1,\ b \in [1, m] \end{cases}$$

As it's noticeable from the previous formulas, surface areas of these non-trivial interconnection networks are not as straightforward as hypercubes and folded hypercubes and in cases such as generalized form of star graphs, $S(n, k)$, or (n, k)-arrangement graphs a closed-from formula has not been proposed yet. In Chapter 4 we propose a simple recurrence to compute surface area of (n, k)-arrangement graphs.

Chapter 3

Closed Form Solutions by Forward Differences

3.1 Definition of Forward Differences

Definition 3.1. Forward distance property of order n for a function $f(x)$ is defined as, for some $h > 0$,

$$\Delta_h^n f(x) = \sum_{i=0}^{n} (-1)^i \binom{n}{i} f(x + (n - i - h)).$$

For example, the first-order forward difference of a function $f(x)$ on real numbers will be $\Delta f(x) = f(x+h) - f(x)$ for a given $h > 0$ and its second-order forward difference is $\Delta\Delta f(x) = \Delta^2 f(x) = \Delta f(x+h) - \Delta f(x) = (f(x+2h) - f(x+h)) - (f(x+h) - f(x)) = f(x+2h) - 2f(x+h) + f(x) = \sum_{i=0}^{2} (-1)^i \binom{2}{i} f(x + (2-i)h)$.

Throughout this research, when we are talking about forward difference property of order n for a function $f(x)$, we mean the n-th forward difference of $f(x)$ is 0, namely $\Delta^n f(x) = \sum_{i=0}^{n} (-1)^i \binom{n}{i} f(x + (n-i)) = 0$. Due to the nature of our problem and data, we will assume, from now on, that $h = 1$.

3.2 Applications of Forward Difference Property

Clearly, a preferable formula for surface area of interconnection networks would be a closed form one. For certain graphs, it has been observed that for surface areas at distance $\mathfrak{R}$ for different dimensions, their $(\mathfrak{R}+1)$-th forward difference is 0. This means that the recurrences for the surface areas at distance $\mathfrak{R}$ for different dimensions are linear homogeneous ones with constant coefficients and order $\mathfrak{R}+1$ for which a known algorithm exists to solve them, resulting in a closed form solution of an $\mathfrak{R}$-th order polynomial.

3.3 Examples of Forward Difference Property on Surface Area of Some Graphs

3.3.1 Hypercubes

As discussed in Section 2.3.1 a closed form surface area of hypercubes is quite straightforward and it is $B_{Q_n,u}(i) = \binom{n}{i}$. It has been shown in [17] that $B_{Q_n,u}(i)$ holds the forward difference property of order $i+1$. In other words, $\sum_{k=0}^{i+1}(-1)^k\binom{i+1}{k}\binom{n+i+1-k}{i} = 0$.

3.3.2 Folded Cubes

Here we present a direct proof to show this property also holds for folded hypercubes.

Theorem 3.2. For $B_{FHC(n)}(i)$, the $(i+1)$-th order forward difference is 0.

Proof. By induction on i, assume that for distance i,

$\sum_{k=0}^{i+1}(-1)^k\binom{i+1}{k}\binom{n+1+i+1-k}{i} =$
$\binom{n+1+i+1}{i} - \binom{i+1}{1}\binom{n+1+i}{i} + \binom{i+1}{2}\binom{n+i}{i} + \cdots + (-1)^{i+1}\binom{i+1}{i+1}\binom{n+1}{i} = 0$
Using the fact that $\binom{n+1}{i} = \binom{n}{i-1} + \binom{n}{i}$ let's rewrite the terms:
$\left[\binom{n+1+i+1+1}{i+1} - \binom{n+1+i+1}{i+1}\right] -$
$\binom{i+1}{1}\left[\binom{n+1+i+1}{i+1} - \binom{n+1+i}{i+1}\right] +$
$\binom{i+1}{2}\left[\binom{n+1+i}{i+1} - \binom{n+1+i-1}{i+1}\right] + \cdots +$
$(-1)^{i+1}\binom{i+1}{i+1}\left[\binom{n+1+1}{i+1} - \binom{n+1}{i+1}\right] =$
$\binom{n+1+i+1+1}{i+1} -$
$\left[\binom{i+1}{0}\binom{n+1+i+1}{i+1} + \binom{i+1}{1}\binom{n+1+i+1}{i+1}\right] +$
$\left[\binom{i+1}{1}\binom{n+1+i}{i+1} + \binom{i+1}{2}\binom{n+1+i}{i+1}\right] - \cdots +$
$\left[(-1)^{i+1}\binom{i+1}{i+1}\binom{n+1+1}{i+1} + \binom{i+1}{i+1}\binom{n+1+1}{i+1}\right] +$
$(-1)^{i+2}\binom{i+1}{i+1}\binom{n+1}{i+1} =$
$\binom{n+1+i+1+1}{i+1} -$
$\binom{i+2}{1}\binom{n+1+i+1}{i+1} +$
$\binom{i+2}{2}\binom{n+1+i}{i+1} - \cdots +$
$(-1)^{i+2}\binom{i+1}{i+1}\binom{n+1}{i+1} =$
$\sum_{k=0}^{i+2}(-1)^k\binom{i+2}{k}\binom{n+1+i+2-k}{i+1}$
which is 0 by induction hypothesis. $\square$

3.4 Obtaining Closed Form Solutions of Surface areas by Solving Homogeneous Linear Recurrences with Constant Coefficients: An Example in the Star Graph.

A linear homogeneous recurrence relation of degree k with constant coefficients is a recurrence relation of the form

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + ... + c_k a_{n-k}$$

where the $c_1, c_2, ..., c_k \in \mathbb{R}$ and $c_k \neq 0$.
Using characteristic equation of such recurrences, one can solve these functions and obtain a closed form and this reveals the importance of the forward difference property as clearly, a closed form formula is preferred.
In [9] Ke Qiu, *et al.* presented a closed form of surface area for star graphs by showing that the surface area holds the property of $(i+1)$-th forward difference and then by utilizing the characteristic function. For example, the general solution for $i = 4$ is shown to be $B_{S_n}(u, 4) = 19 - 245n/6 + 30n^2 - 55n^3/6 + n^4$.
Based on the approach of this research, the following steps should be taken in order to obtain a closed form formula for $B_{G,u}(i)$ of an arbitrary graph G.

- **Step 1**. Show that the surface area satisfies the forward difference property.

- **Step 2**. Since any function that holds forward difference property can be considered as a homogeneous linear recurrence with constant coefficients, its characteristic equation can be utilized to find a closed form of $B_{G,u}(i)$.

Chapter 4

Obtaining A Recurrence for Surface Area of (n, k)-Arrangement and (n, k)-Star Graphs

4.1 A Recurrence for Surface Area of (n, k)-Arrangement Graph

In this section, we are focusing on finding a recursive relationship for $B_{A(n,k)}(i)$.
As mentioned in Chapter 1, the $A(n, n-1)$ is isomorphic to S_n, therefore, the surface area for $A(n, n-1)$ is known as a result of Ke Qiu *et. al.* work in [9]. However, for other values of k, so far there have been no simple recurrence to compute $B_{A(n,k)}(i)$ in general, other than for some trivial cases such as $k = 1$. In this section, we propose a generalized recursive formula to compute $B_{A(n,k)}(i)$ for any arbitrary values of n and k in their defined range.

As mentioned in Section 1.4.10, $A(n,k)$ has a hierarchical structure and is built of n, $A(n-1,k-1)$ subgraphs. As we mentioned, we consider this partitioning based on the last symbol of nodes, in partition 1 for example, there are only nodes whose representations end with symbol 1, and partition 2 consists of nodes ending with 2 and so forth. Each partition is an $(n-1,k-1)$-arrangement graph since excluding the last symbol, there are $n-1$ symbols to fill $k-1$ positions in the node representation. We denote these subgraphs by $A_i(n-1,k-1)$ for the partition with the last symbol equal to i.

Having this structure, it is very likely that there is a recursive relationship for this graph's properties. Similarly to [9] let $f(n,k,i,j)$ be the number of nodes of the form $*j$ at distance i from e in $A(n,k)$ and $g(n,k,i)$ be the total number of nodes at distance i from e in $A(n,k)$. Clearly, $g(n,k,i) = \sum_{j=1}^{n} f(n,k,i,j)$ and for $A(n,k)$, $\sum_{i=0}^{\lfloor 3k/2 \rfloor} g(n,k,i) = \frac{n!}{(n-k)!}$.

In Tables 4.1, 4.2, 4.3, some values for $f(n,k,i,j)$ when $k=n-1$, $k=n-2$ and $k=n-3$ in (n,k)-arrangement graphs are provided. Note that in these tables, the sum of each row will be the related value of $g(n,k,i) = B_{A(n,k)}(i)$.

Theorem 4.1. In $A(n,k)$, for all n, $1 \leq k < n$, $0 \leq i \leq D(A(n,k))$, we have $f(n,k,i,k) = g(n-1,k-1,i)$.

Proof. Based on the hierarchical structure of (n,k)-arrangement graph that was discussed before, all nodes with the last symbol equal to k, (as well as the source node $e = 12...k$) are in partition $A_k(n-1,k-1)$. Each partition is isomorphic to an $(n-1,k-1)$-arrangement graph. Clearly, $f(n,k,i,j) = g(n-1,k-1,i)$. □

Lemma 4.2. In an (n,k)-arrangement graph,

1) $f(n,k,i,r) = f(n,k,i,s)$ for all $1 \leq r,s \leq k-1$, and

2) $f(n,k,i,j) = f(n,k,i,l)$ for all $k+1 \leq j,l \leq n$.

Table 4.1: Values for $f(n, k, i, j)$ in $A(n, k = n - 1)$

$i = 1$									
j:	1	2	3	4	5	6	7	8	9
(2, 1)	0	1							
(3, 2)	0	1	1						
(4, 3)	0	0	2	1					
(5, 4)	0	0	0	3	1				
(6, 5)	0	0	0	0	4	1			
(7, 6)	0	0	0	0	0	5	1		
(8, 7)	0	0	0	0	0	0	6	1	
(9, 8)	0	0	0	0	0	0	0	7	1

$i = 2$									
j:	1	2	3	4	5	6	7	8	9
(2, 1)									
(3, 2)	1	0	1						
(4, 3)	1	1	2	2					
(5, 4)	1	1	1	6	3				
(6, 5)	1	1	1	1	12	4			
(7, 6)	1	1	1	1	1	20	5		
(8, 7)	1	1	1	1	1	1	30	6	
(9, 8)	1	1	1	1	1	1	1	42	7

$i = 3$									
j:	1	2	3	4	5	6	7	8	9
(3, 2)	1	0	0						
(4, 3)	3	3	1	2					
(5, 4)	5	5	5	9	6				
(6, 5)	7	7	7	7	30	12			
(7, 6)	9	9	9	9	9	70	20		
(8, 7)	11	11	11	11	11	11	135	30	
(9, 8)	13	13	13	13	13	13	13	231	42

$i = 4$									
j:	1	2	3	4	5	6	7	8	9
(3, 2)									
(4, 3)	2	2	0	1					
(5, 4)	10	10	10	5	9				
(6, 5)	24	24	24	24	44	30			
(7, 6)	44	44	44	44	44	170	70		
(8, 7)	70	70	70	70	70	70	460	135	
(9, 8)	102	102	102	102	102	102	102	1015	231

$i = 5$									
j:	1	2	3	4	5	6	7	8	9
(5, 4)	7	7	7	0	5				
(6, 5)	45	45	45	45	26	44			
(7, 6)	138	138	138	138	138	250	170		
(8, 7)	310	310	310	310	310	310	1110	460	
(9, 8)	585	585	585	585	585	585	585	3430	1015

$i = 6$									
j:	1	2	3	4	5	6	7	8	9
(5, 4)	1	1	1	0	0				
(6, 5)	35	35	35	35	3	26			
(7, 6)	254	254	254	254	254	169	250		
(8, 7)	930	930	930	930	930	930	1689	1110	
(9, 8)	2455	2455	2455	2455	2455	2455	2455	8379	3430

Table 4.2: Values for $f(n, k, i, j)$ in $A(n, k = n - 2)$

$i = 1$									
j:	1	2	3	4	5	6	7	8	9
(3, 1)	0	1	1						
(4, 2)	0	2	1	1					
(5, 3)	0	0	4	1	1				
(6, 4)	0	0	0	6	1	1			
(7, 5)	0	0	0	0	8	1	1		
(8, 6)	0	0	0	0	0	10	1	1	
(9, 7)	0	0	0	0	0	0	12	1	1

$i = 2$									
j:	1	2	3	4	5	6	7	8	9
(3, 1)									
(4, 2)	2	0	2	2					
(5, 3)	2	2	6	4	4				
(6, 4)	2	2	2	18	6	6			
(7, 5)	2	2	2	2	36	8	8		
(8, 6)	2	2	2	2	2	60	10	10	
(9, 7)	2	2	2	2	2	2	90	12	12

$i = 3$									
j:	1	2	3	4	5	6	7	8	9
(4, 2)	1	0	0	0					
(5, 3)	7	7	1	6	6				
(6, 4)	13	13	13	27	18	18			
(7, 5)	19	19	19	19	102	36	36		
(8, 6)	25	25	25	25	25	250	60	60	
(9, 7)	31	31	31	31	31	31	495	90	90

$i = 4$									
j:	1	2	3	2	2	6	7	8	9
(4, 2)									
(5, 3)	3	3	0	1	1				
(6, 4)	30	30	30	8	27	27			
(7, 5)	81	81	81	81	152	102	102		
(8, 6)	156	156	156	156	156	680	250	250	
(9, 7)	255	255	255	255	255	255	1960	495	495

$i = 5$									
j:	1	2	3	4	5	6	7	8	9
(6, 4)	14	14	14	0	8	8			
(7, 5)	162	162	162	162	58	152	152		
(8, 6)	564	564	564	564	564	1010	680	680	
(9, 7)	1340	1340	1340	1340	1340	1340	5190	1960	1960

$i = 6$									
j:	1	2	3	4	5	6	7	8	9
(6, 4)	1	1	1	0	0	0			
(7, 5)	85	85	85	3	58	58			
(8, 6)	1054	1054	1054	1054	459	1010	1010		
(9, 7)	4430	4430	4430	4430	4430	4430	7749	5190	5190

Table 4.3: Values for $f(n, k, i, j)$ in $A(n, k = n - 3)$

$i = 1$										$i = 2$									
j:	1	2	3	4	5	6	7	8	9	j:	1	2	3	4	5	6	7	8	9
(4, 1)	0	1	1	1						(4, 1)									
(5, 2)	0	3	1	1	1					(5, 2)	3	0	3	3	3				
(6, 3)	0	0	6	1	1	1				(6, 3)	3	3	12	6	6	6			
(7, 4)	0	0	0	9	1	1	1			(7, 4)	3	3	3	36	9	9	9		
(8, 5)	0	0	0	0	12	1	1	1		(8, 5)	3	3	3	3	72	12	12	12	
(9, 6)	0	0	0	0	0	15	1	1	1	(9, 6)	3	3	3	3	3	120	15	15	15
$i = 3$										$i = 4$									
j:	1	2	3	4	5	6	7	8	9	j:	1	2	3	2	2	6	7	8	9
(5, 2)	1	0	0	0	0					(5, 2)									
(6, 3)	13	13	1	12	12	12				(6, 3)	4	4	0	1	1	1			
(7, 4)	25	25	25	63	36	36	36			(7, 4)	68	68	68	11	63	63	63		
(8, 5)	37	37	37	37	246	72	72	72		(8, 5)	192	192	192	192	404	246	246	246	
(9, 6)	49	49	49	49	49	610	120	120	120	(9, 6)	376	376	376	376	376	1910	610	610	610
$i = 5$										$i = 6$									
j:	1	2	3	4	5	6	7	8	9	j:	1	2	3	4	5	6	7	8	9
(7, 4)	23	23	23	0	11	11	11			(7, 4)	3	3	3	0	0	0	0		
(8, 5)	429	429	429	429	102	404	404	404		(8, 5)	165	165	165	165	3	102	102	102	
(9, 6)	1578	1578	1578	1578	1578	3030	1910	1910	1910	(9, 6)	3174	3174	3174	3174	3174	969	3030	3030	3030

Proof. 1) According to the hierarchical structure that we discussed earlier, we can consider the node $\mathcal{V}_1 = n2...(k-1)1$, $\mathcal{V}_2 = 1n...(k-1)2$, $\cdots$ and node $\mathcal{V}_{k-1} = 12...n(k-1)$ as the source nodes in subgraphs $A_1(n-1, k-1)$, $A_2(n-1, k-1)$, $\cdots$ and $A_{k-1}(n-1, k-1)$ respectively. $\mathcal{V}_1, \mathcal{V}_2, ..., \mathcal{V}_{k-1}$ are at distance 2 from the source node in $A(n, k)$, $e = 12...(k-1)k$. Clearly, $A_i(n-1, k-1)$ for $1 \leq i \leq k-1$ are isomorphic to each other, hence the number of nodes in $A_1(n-1, k-1)$ at distance j from $\mathcal{V}_1$ is equal to the number of nodes in $A_2(n-1, k-1)$ at distance j from $\mathcal{V}_2$ which is equal to the number of nodes in $A_3(n-1, k-1)$ at distance j from $\mathcal{V}_3$ and so on.

2) The proof for part 2 is similar to part one. Note that here the source node in subgraph $A_{k+1}(n-1, k-1)$ is $\mathcal{V}_{k+1} = 12...(k-1)(k+1)$, and in $A_{k+2}(n-1, k-1)$ is $\mathcal{V}_{k+2} = 12...(k-1)(k+2)$ and so forth. These source nodes are all at distance 1 from $e = 12...(k-1)k$. □

Theorem 4.3. For all n, $1 \leq k < n$, $0 \leq i \leq D(A(n, k))$ and $j \in \{k+1, k+2, ..., n\}$ we have $f(n, k, i, j) = g(n-1, k-1, i-1)$.

Proof. Without loss of generality and by using Lemma 4.2, let's count the number of nodes at distance i in $A(n, k)$ with the last symbol equal to n, namely nodes in partition $A_n(n-1, k-1)$. Due to node symmetry in arrangement graphs, consider the node $u = 12...(k-1)n$ as source node in $A_n(n-1,k-1)$ which is adjacent to $e = 12...(k-1)k$ in $A_k(n-1,k-1)$. By the cyclic structure and cycle correction algorithm that was discussed in Section 1.5, if there exist a shortest path of length i from a node of form $(*...n)$ to node $e = 12...(k-1)k$, then at least one of such paths takes the form of

$$\overbrace{(*...n) \to \cdots \to 12...(k-1)n}^{i-1} \to 12...(k-1)k$$

which completely stays in the partition. Therefore, $f(n,k,i,j) = g(n-1,k-1,i-1)$. □

Lemma 4.4. In $A(n,k)$, for $i > 2$, the number of nodes at distance i from e with last symbol 1 that use node $a = n2...(k-1)1$ to reach e is equal to $f(n,k,i,n)$.

Proof. Nodes in partition $A_1(n-1,k-1)$ (the partition where last symbol is 1), that are at distance i from e and use node $a = n2...(k-1)1$ to reach e have a path format as: $\overbrace{u_1 \to u_2 \to ... \to a = n2...(k-1)1}^{i-2} \to n2...(k-1)k \to e = 12...(k-1)k$. On the other hand, nodes in partition $A_n(n-1,k-1)$ (the partition where last symbol is n), that are at distance i from e have a path format as: $\overbrace{u_1 \to u_2 \to ... \to m}^{i-2} \to b = 12...(k-1)n \to e = 12...(k-1)k$ where, $m \in \{$nodes in $A_n(n-1,k-1)$ connected to $b\} \setminus \{e\}$. Then node m is at distance 2 from e and plays a similar role as node a in $A_1(n-1,k-1)$. Since the number of nodes at distance $i-2$ from a in $A_1(n-1,k-1)$ is equal to the number of nodes at distance $i-2$ from m in $A_n(n-1,k-1)$, then the number of nodes at distance $i-2$ from a in $A_1(n-1,k-1)$ is equal to the number of nodes that are at distance $i-1$ from b in $A_n(n-1,k-1)$ ($= f(n,k,i,n)$). □

Theorem 4.5. For all n, $1 \leq k < n$, $0 \leq i \leq D(A(n,k))$ and $j \in \{1,2,...,k-1\}$ we have $f(n,k,i,j) = g(n-1,k-1,i-1) + g(n-2,k-2,i-3) - g(n-2,k-2,i-1)$.

Proof. Using the partitioning technique that we showed before, in $A_1(n-1,k-1)$ the closest node to $e = 12...(k-1)k$ (the source node in $A(n,k)$) is at distance 2 and in the form of $v_1 2...(k-1)1$ where $v_1 \in \{n-k+1,...,n-1,n\}$. Let's assume $a = n2...(k-1)1$ is the source node in $A_1(n-1,k-1)$. We define set A as all nodes in $A_1(n-1,k-1)$ that are at distance $i-2$ from a (therefore, at distance i from e). Using Lemma 4.4 and Theorem 4.5, there are $g(n-1,k-1,i-1)$ many of these nodes and hence, $|A| = g(n-1,k-1,i-1)$.

Set A does not contain all nodes at distance i from e in $A_1(n-1,k-1)$. For instance, in $A(5,3)$ node $b = 241$ is at distance 3 from e ($b = 241 \rightarrow 243 \rightarrow 142 \rightarrow e = 123$), however, b is not connected to any node of the form $v_1 2...(k-1)1$.

Node b does not contain symbol k as if it does, it can not be at distance less than 4 from e. By a recursive approach, let's consider $A(n-2,k-2)$ with b as its source node and define the set B as all nodes in $A(n-2,k-2)$ that are at distance $i-3$ from b (at distance i from e). Note that this subgraph is inside $A_1(n-1,k-1)$.

There are nodes that belong to both sets A and B. For example, in $A(5,3)$, $u = 231$ is adjacent to $b = 241$ and therefore, there exists path p_1 as $u = 231 \rightarrow b = 241 \rightarrow 243 \rightarrow 143 \rightarrow e = 123$ and hence $u \in B$. u is also a neighbor of $v = 531$ which provides a path to e through node a and with the same length as p_1 ($u = 231 \rightarrow v = 531 \rightarrow a = 521 \rightarrow 523 \rightarrow e = 123$), so $u \in A$.

The intersection is the set of all nodes in $A(n-2,k-2)$ that are also at distance $i-1$ from a. Hence, the cardinality of $A \cup B$, all nodes in $A(n,k)$ at distance i from e with last symbol 1 is:

$$|A \cup B| = |A| + |B| - |A \cap B|$$

$$= g(n-1,k-1,i-1) + g(n-2,k-2,i-3) - g(n-2,k-2,i-1)$$

□

Theorem 4.6. In an (n,k)-arrangement graph, for all n, $1 \leq k < n$, for $g(n,k,i)$ ($= B_{A(n,k)}(i)$) we have:

$$\begin{cases} 1 & i=0 \\ k(n-k) & i=1 \\ k(n-k)((k(n-k)-1)-(n-k-1)(1+\frac{k-1}{2})) & i=2 \\ (k-1)\times(k-3)\times\cdots\times 1 & i=\lfloor\frac{3k}{2}\rfloor and \text{ even } k \\ 0 & i>\lfloor\frac{3k}{2}\rfloor \end{cases}$$

and for $3 \leq i \leq D(A(n,k))$, $g(n,k,i)$ is equal to

$$(n-1)g(n-1,k-1,i-1)+(k-1)[g(n-2,k-2,i-3)-g(n-2,k-2,i-1)]+g(n-1,k-1,i).$$

Proof. Since for $i=0$, $i=1$ and $i>\lfloor\frac{3k}{2}\rfloor$ the proof is trivial, we only show the other cases.

When $i=2$:Since $A(n,k)$ is $k(n-k)$-regular, e has $k(n-k)$ neighbors that each of these nodes also has $k(n-k)$ nodes as neighbors. Let's call the set of nodes that are connected to e as L_1 and assume the set L_2 is {all nodes connected to v_i, for all $v_i \in \{L_1 \setminus \{e\}\}$}. Hence, $|L_2| = k(n-k)(k(n-k)-1)$ nodes can possibly be at distance 2 from e. Among nodes in L_2 there are $k(n-k)(n-k-1)$ many nodes that are also in L_1, since for each node in L_1 there is $(n-k-1)$ many symbols left to choose to connect those nodes to e. Therefore, $E_1 = k(n-k)(n-k-1)$ should be excluded from $|L_2|$ as these nodes are at distance 1 from e. From the other hand, there are nodes in L_2 that are connected to the same node in L_1 and hence they are counted twice. More precisely, each node in L_1 is connected to $(k-1)(n-k-1)$ nodes whose are connect to another node in L_1. Since by reducing one bit from the node representation $(k-1)$ and having $(n-k-1)$ symbols to choose from one can connect a node to $(k-1)(n-k-1)$ many nodes. Thus, half of this amount should be deducted in order to count these nodes only once, namely, $E_2 = \dfrac{k(n-k)(k-1)(n-k-1)}{2}$ should also be subtracted from $|L_2|$. This will leave us with

$$k(n-k)(k(n-k)-1)-k(n-k)(n-k-1)-\frac{k(n-k)(k-1)(n-k-1)}{2}.$$

By simplifying this statement the goal is achieved.
<u>When $i = \lfloor \frac{3k}{2} \rfloor$ and k is even</u>: Based on the cyclic routing algorithm that was discussed in Section 1.5, the cyclic representation of nodes that are at distance d (the diameter of $A(n,k)$) from the identity node e, have the maximum value of $c+m-2z$ in which c is the number of cycles, m is the number of misplaced symbols and z is the number of external cycles. If $c = k/2$ for even k, $m = k$ and $z = 0$, $c+m-2z$ will be at its maximum value. Now we count the number of nodes that satisfy these conditions. From $z = 0$ we exclude nodes that contain any external symbol, thus the set of symbols one can choose from for node representation has only k elements ($\{1, 2, ..., k\}$). In addition, these nodes must have $c = k/2$ cycles in a way that there is no invariant symbol as m ought to be maximum. The number of ways to pair k elements with each other (to keep the number of cycles maximum) and no symbol be in its correct position is $(k-1)$ symbols for the first cycle, $(k-3)$ symbols for the second cycle and so on until the last cycle which only has one pair of symbols to take and this is equal to $(k-1) \times (k-3) \times \cdots \times 1$.
<u>When $i \geq 3$</u>: By the definition of two functions $f(n,k,i,j)$ and $g(n,k,i)$,

$$g(n,k,i) = \sum_{j=1}^{n} f(n,k,i,j) =$$

$$\sum_{j=1}^{k-1} f(n,k,i,j) + f(n,k,i,k) + \sum_{j=k+1}^{n} f(n,k,i,j).$$

From Theorems 4.1, 4.2 and 4.5, $g(n,k,i)$ is equal to
$(k-1)[g(n-1,k-1,i-1)+g(n-2,k-2,i-3)-g(n-2,k-2,i-1)]$
$+g(n-1,k-1,i-1)+(n-k)g(n-1,k-1,i-1)$.
By simplifying the same terms the goal is achieved. □
It is worth mentioning that if we define $B_{A(n,k)}(i)=0$ for $i<0$, then $B_{A(n,k)}(2)$ also can be derived from the above recurrence.

4.1.1 The Significance of the Proposed Recurrence

From the computational complexity point of view, surface area can be computed in polynomial time in terms of the graph size [28], however, the number of nodes in an interconnected network in most cases is exponential or factorial regarding its parameters. For example, an S_n graph has $n!$ and (n,k)-arrangement graph has $\frac{n!}{(n-k)!}$ nodes. A naive solution to compute surface area of an interconnection network is to iterate nodes by a Breadth First Search (BFS) which takes $O(N+E)$ where N is the number of nods and E is the number of edges. In terms of the star graph, this would be $O(n!+n!n)=O(n\times n!)$ which is exceptionally expensive.
In general, it is not trivial to find an algorithm to compute surface area of such graphs in polynomial time in terms of its parameters or to find an explicit formula of surface area with polynomially many terms in regards to the parameter n (or n and k).
Despite of many work done to compute the node centered surface areas that include explicit formulas which some was mentioned in Section 2.3, the formula that Portier and Vaughan proposed in [25] gives a simple recurrence that allows us to compute $B_{S_n}(i)$ (which is a special case of $A(n,k)$ and $S(n,k)$ when $k=n-1$) in linear time which is very efficient. The recurrence proposed in this research has the computational complexity of the same order as Portier and Vaughan's, but for a much wider family of graphs, namely $A(n,k)$ for any arbitrary k in its de-

fined range which is a tremendous improvement over the naive solution. From the other hand, the proposed recurrence is a comprehensive generalization of recurrences proposed in [25] and therefore it's as simple as the simplest solution available but for all (n, k)-arrangement graphs. Figure 4.1 provides a 3-dimensional diagram of $B_{A(n,k)}(i)$.

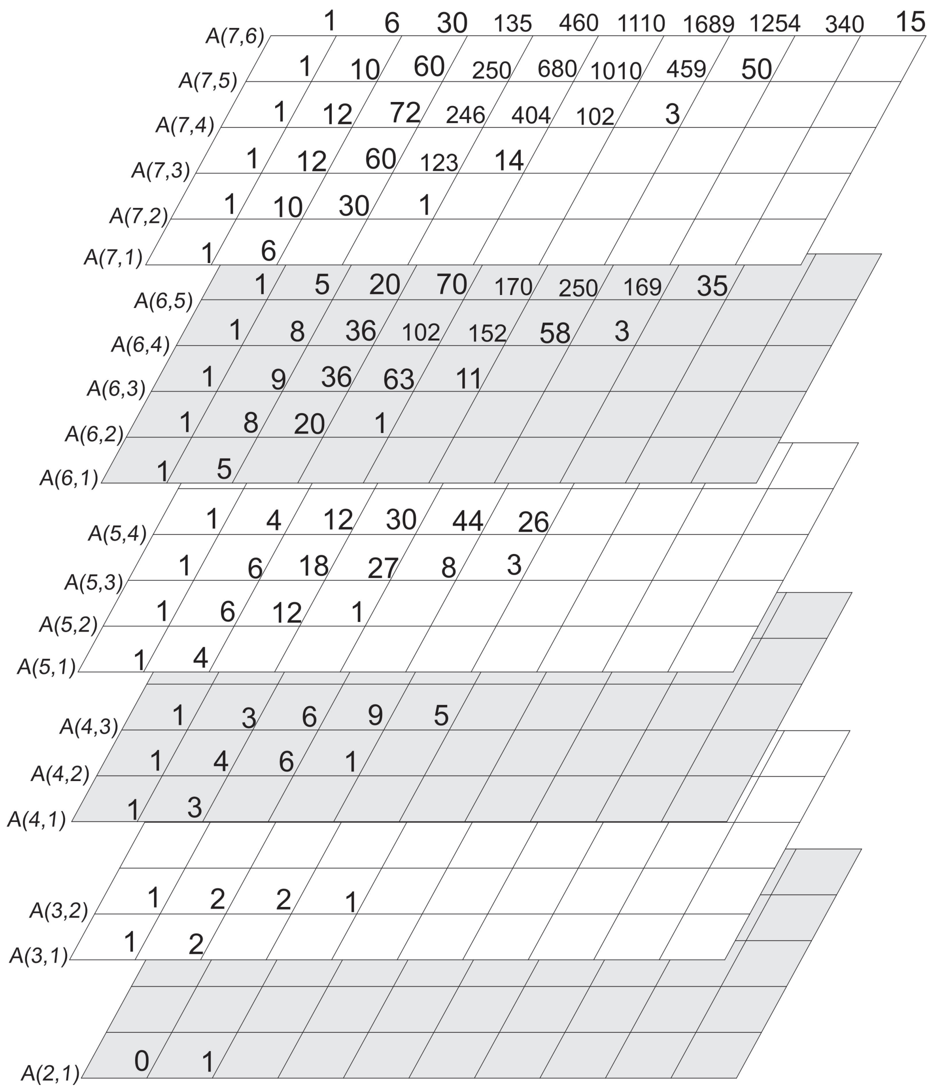

Figure 4.1: A 3D illustration of $B_{A(n,k)}(i)$

4.2 A Recurrence for Surface Area of (n, k)-Star Graph

In this study, we have taken another step to investigate (n, k)-star graphs in order to find a simple recurrence for surface area of this class of graphs which yielded to some partial results. The objective of this section is to explore ideas that will help us to count the number of nodes at distance i from the identity node (e) in the (n, k)-star graph. According to the hierarchical structure of the (n, k)-star graph which was explained in Section 1.4.9, $S(n, k)$ can be partitioned into n isomorphic $S(n-1, k-1)$ subgraphs such that in each subgraph nodes are of the form $*j$ for $1 \leq j \leq n$. In this study, we use $S_i(n-1, k-1)$ to denote the partition whose nodes' last symbol is i. In Tables 4.4, 4.5, 4.6, some values for $f(n, k, i, j)$ when $k = n-1$, $k = n-2$ and $k = n-3$ in the (n, k)-star graphs are provided. Note that in these tables, the sum of each row will be the related value of $g(n, k, i) = B_{S(n,k)}(i)$.

Considering functions f and g that have been defined in Section 4.1, it can be easily derived that:

Theorem 4.7. In an (n, k)-star graph,

(1) $f(n, k, 0, j) = 0$ for $j \neq k$ and $f(n, k, 0, k) = 1$, therefore, $g(n, k, 0) = 1$.

(2) $f(n, k, 1, j) = 0$ for $j \neq 1, k$ and $f(n, k, 1, 1) = 1$, and $f(n, k, 1, k) = n - 2$, therefore, $g(n, k, 1) = n - 1$.

(3) $f(n, k, 2, j) = n - 2$ for $j \neq 1, k$ and $f(n, k, 2, 1) = n - 2$, and

$$f(n, k, 2, k) = \begin{cases} 0 & k \leq 2 \\ (k-2)(2n-k-3) & otherwise \end{cases}$$

therefore, $g(n, k, 2) = 2(n-2) + \begin{cases} 0 & k \leq 2 \\ (k-2)(2n-k-3) & otherwise \end{cases}$.

Table 4.4: Values for $f(n, k, i, j)$ in $S(n, k = n - 1)$

$i = 1$									
j:	1	2	3	4	5	6	7	8	9
(2, 1)	0	1							
(3, 2)	1	1	0						
(4, 3)	1	0	2	0					
(5, 4)	1	0	0	3	0				
(6, 5)	1	0	0	0	4	0			
(7, 6)	1	0	0	0	0	5	0		
(8, 7)	1	0	0	0	0	0	6	0	
(9, 8)	1	0	0	0	0	0	0	7	0

$i = 2$									
j:	1	2	3	4	5	6	7	8	9
(2, 1)									
(3, 2)	1	0	1						
(4, 3)	2	1	2	1					
(5, 4)	3	1	1	6	1				
(6, 5)	4	1	1	1	12	1			
(7, 6)	5	1	1	1	1	20	1		
(8, 7)	6	1	1	1	1	1	30	1	
(9, 8)	7	1	1	1	1	1	1	42	1

$i = 3$									
j:	1	2	3	4	5	6	7	8	9
(3, 2)	0	0	1						
(4, 3)	2	3	1	3					
(5, 4)	6	5	5	9	5				
(6, 5)	12	7	7	7	30	7			
(7, 6)	20	9	9	9	9	70	9		
(8, 7)	30	11	11	11	11	11	135	11	
(9, 8)	42	13	13	13	13	13	13	231	13

$i = 4$									
j:	1	2	3	4	5	6	7	8	9
(3, 2)									
(4, 3)	1	2	0	2					
(5, 4)	9	10	10	5	10				
(6, 5)	30	24	24	24	44	24			
(7, 6)	70	44	44	44	44	170	44		
(8, 7)	135	70	70	70	70	70	460	70	
(9, 8)	231	102	102	102	102	102	102	1015	102

$i = 5$									
j:	1	2	3	4	5	6	7	8	9
(5, 4)	5	7	7	0	7				
(6, 5)	44	45	45	45	26	45			
(7, 6)	170	138	138	138	138	250	138		
(8, 7)	460	310	310	310	310	310	1110	310	
(9, 8)	1015	585	585	585	585	585	585	3460	585

$i = 6$									
j:	1	2	3	4	5	6	7	8	9
(5, 4)	0	1	1	0	1				
(6, 5)	26	35	35	35	3	35			
(7, 6)	250	254	254	254	254	169	254		
(8, 7)	1110	930	930	930	930	930	1689	930	
(9, 8)	3430	2455	2455	2455	2455	2455	2455	8379	2455

Table 4.5: Values for $f(n, k, i, j)$ in $S(n, k = n - 2)$

$i = 1$									
j:	1	2	3	4	5	6	7	8	9
(3, 1)	0	1	1						
(4, 2)	1	2	0	0					
(5, 3)	1	0	3	0	0				
(6, 4)	1	0	0	4	0	0			
(7, 5)	1	0	0	0	5	0	0		
(8, 6)	1	0	0	0	0	6	0	0	
(9, 7)	1	0	0	0	0	0	7	0	0

$i = 2$									
j:	1	2	3	4	5	6	7	8	9
(3, 1)									
(4, 2)	2	0	1	1					
(5, 3)	3	1	4	1	1				
(6, 4)	4	1	1	10	1	1			
(7, 5)	5	1	1	1	18	1	1		
(8, 6)	6	1	1	1	1	28	1	1	
(9, 7)	7	1	1	1	1	1	40	1	1

$i = 3$									
j:	1	2	3	4	5	6	7	8	9
(4, 2)	0	0	2	2					
(5, 3)	4	5	4	4	4				
(6, 4)	10	7	7	21	6	6			
(7, 5)	18	9	9	9	57	8	8		
(8, 6)	28	11	11	11	11	118	10	10	
(9, 7)	40	13	13	13	13	13	210	12	12

$i = 4$									
j:	1	2	3	2	2	6	7	8	9
(4, 2)									
(5, 3)	4	6	0	6	6				
(6, 4)	21	20	20	22	18	18			
(7, 5)	57	40	40	40	119	36	36		
(8, 6)	118	66	66	66	66	368	60	60	
(9, 7)	210	98	98	98	98	98	870	90	90

$i = 5$									
j:	1	2	3	4	5	6	7	8	9
(5, 3)	0	0	0	1	1				
(6, 4)	22	26	26	2	27	27			
(7, 5)	119	107	107	107	130	102	102		
(8, 6)	368	267	267	267	267	774	250	250	
(9, 7)	870	530	530	530	530	530	2710	495	495

$i = 6$									
j:	1	2	3	4	5	6	7	8	9
(5, 3)									
(6, 4)	2	6	6	0	8	8			
(7, 5)	130	149	149	149	30	152	152		
(8, 6)	744	701	701	701	701	911	680	680	
(9, 7)	2710	2054	2054	2054	2054	2054	5849	1960	1960

Table 4.6: Values for $f(n, k, i, j)$ in $S(n, k = n - 3)$

$i = 1$										$i = 2$									
j:	1	2	3	4	5	6	7	8	9	j:	1	2	3	4	5	6	7	8	9
(4, 1)	0	1	1	1						(4, 1)									
(5, 2)	1	3	0	0	0					(5, 2)	3	0	1	1	1				
(6, 3)	1	0	4	0	0	0				(6, 3)	4	1	6	1	1	1			
(7, 4)	1	0	0	5	0	0	0			(7, 4)	5	1	1	14	1	1	1		
(8, 5)	1	0	0	0	6	0	0	0		(8, 5)	6	1	1	1	24	1	1	1	
(9, 6)	1	0	0	0	0	7	0	0	0	(9, 6)	7	1	1	1	1	36	1	1	1
$i = 3$										$i = 4$									
j:	1	2	3	4	5	6	7	8	9	j:	1	2	3	2	2	6	7	8	9
(5, 2)	0	0	3	3	3					(5, 2)									
(6, 3)	6	7	9	5	5	5				(6, 3)	9	12	0	10	10	10			
(7, 4)	14	9	9	37	7	7	7			(7, 4)	37	32	32	51	26	26	26		
(8, 5)	24	11	11	11	90	9	9	9		(8, 5)	90	58	58	58	230	48	48	48	
(9, 6)	36	13	13	13	13	174	11	11	11	(9, 6)	174	90	90	90	90	638	76	76	76
$i = 5$										$i = 6$									
j:	1	2	3	4	5	6	7	8	9	j:	1	2	3	4	5	6	7	8	9
(6, 3)	0	0	0	4	4	4				(6, 3)									
(7, 4)	51	57	57	12	51	51	51			(7, 4)	12	21	21	0	33	33	33		
(8, 5)	230	193	193	193	330	165	165	165		(8, 5)	330	355	355	355	153	323	323	323	
(9, 6)	638	432	432	432	432	1634	370	370	370	(9, 6)	1634	1394	1394	1394	1394	2517	1214	1214	1214

Proof. The proofs for (1) and (2) follow from the definition of two functions f and g and the structure of $S(n, k)$. In case (3), for $f(n, k, 2, 1)$ it's sufficient to focus on subgraph $S(n-1, k-1)$ with nodes of the form $*1$. Node $u = k23...1$ is connected to identity node and thus is at distance 1 from e. Therefore, all its neighbors in the same subgraph are at distance 2 from e with the last symbol 1 and since there is no node of the form $*1$ in other subgraphs, this will be equal to $f(n, k, 2, 1)$. Node u has $n-1$ neighbors of which one of them is e. Thus, $f(n, k, 2, 1) = n - 2$. For $f(n, k, 2, j), j \neq 1, k$, consider $n - 2$ neighbors of e that are connected to one of the $n - 2$ subgraphs that are of the form $*j$ for $j \neq 1, k$. For $f(n, k, 2, k)$ notice that nodes at distance 1 (e's neighbors) of the form $*k$ can be obtained by two ways: first, swapping $v(1)$ and $v(i)$ for $2 \leq i \leq k-1$ and second, replacing $v(1)$ with an external symbol. In the first way, there are $(k-2)$ nodes that can be connected to $(k-3)$ nodes by swapping and to $(n-k)$ nodes by replacing with external nodes which gives $(k-2)((k-3)+(n-k)$ nodes at distance 2. In the second way, there are $(n-k)$ nodes that can be connected to $(k-2)$ different nodes by swapping $v(1)$ and $v(i)$

for $2 \leq i \leq k-1$ which is $(k-2)$ cases. Therefore, all together, $f(n,k,2,k) = (k-2)(n-3) + (n-k)(k-2) = (k-2)(2n-k-3)$. □
It is worth mentioning that derivations for star graphs in [9] can be obtained here too which is obvious as star graphs are special case of (n,k)-stars when $k = n-1$.

Lemma 4.8. In $S(n,k)$, 1) for all $1 < j,h < k$, $f(n,k,i,j) = f(n,k,i,h)$ and 2) for all $k < r,s \leq n$, $f(n,k,i,r) = f(n,k,i,s)$.

Proof. The proof is similar to that for Lemma 4.2. □

Theorem 4.9. For $0 \leq i \leq D(\mathrm{S(n,\ k)})$, $f(n,k,i,1) = g(n-1,k-1,i-1)$.

Proof. Let's count $f(n,k,i,1)$ in $S(n,k)$. Since we only need to count nodes of the form $*1$, we need to count nodes in subgraph $S_1(n-1,k-1)$ that are at distance i from identity node. By cycle correction algorithm that was discussed in Section 1.5, if there exists shortest paths of length i from a node of form $(*...1)$ to node $e = 12...(k-1)k$, then at least one of such paths takes the form of

$$\overbrace{(*...1) \to \cdots \to u = k2...(k-1)1}^{i-1} \to 12...(k-1)k$$

which completely stays in the partition. Notice that due to node symmetry we can consider u as identity node in subgraph $S_1(n-1,k-1)$. By definition and with a recursive approach, the total number of such nodes are $g(n-1,k-1,i-1)$. On the other hand, u is adjacent to $12...(k+1)$ which means $f(n,k,i,1) = g(n-1,k-1,i-1)$. □

Theorem 4.10. In $S(n,k)$, $f(n,k,i,k) = g(n-1,k-1,i)$.

Proof. Based on the hierarchical structure of (n,k)-star graphs that was discussed before, all nodes with the last symbol equal to k, (as well as the source node $e = 12...k$) are in partition $S_k(n-1,k-1)$. Clearly, $f(n,k,i,j) = g(n-1,k-1,i)$. □

Lemma 4.11. In $S(n,k)$, for $i > 2$, the number of nodes at distance i

from e with last symbol 2 that use node $k13\cdots(k-1)2$ to reach to e is equal to $g(n-1,k-1,i-1)$.

Proof. Recall that in the hierarchical approach we used in $S(n,k)$, $S_i(n-1,k-1)$ denotes the partition that is isomorphic to $S(n-1,k-1)$ and contains the nodes with last symbol i. Nodes in $S_2(n-1,k-1)$ that are at distance i from e have a path format as:

$$\overbrace{u_1 \to u_2 \to ... \to a = k1...(k-1)2}^{i-2} \to 21...(k-1)k \to e = 12...(k-1)k$$

On the other hand, nodes in partition $A_1(n-1,k-1)$, that are at distance i from e have a path format as: $\overbrace{u_1 \to u_2 \to ... \to m}^{i-2} \to b = k2...(k-1)1 \to e = 12...(k-1)k$ where, $m \in \{$nodes in $A_1(n-1,k-1)$ connected to $b\}\backslash\{e\}\}$. Then node m is at distance 2 from e and plays a similar role as node a in $A_2(n-1,k-1)$. Since the number of nodes at distance $i-2$ from a in $A_2(n-1,k-1)$ is equal to the number of nodes at distance $i-2$ from m in $A_1(n-1,k-1)$, then the number of nodes at distance $i-2$ from a in $A_2(n-1,k-1)$ is equal to the number of nodes that are at distance $i-1$ from b in $A_1(n-1,k-1)$ which by Theorem 4.9, is equal to $g(n-1,k-1,i-1)$. $\square$

Theorem 4.12. In $S(n,k)$, for all $2 \le j \le k-1$ and $0 \le i \le D(S(n,k))$, $f(n,k,i,j) = g(n-1,k-1,i-1) + g(n-2,k-2,i-3) - g(n-2,k-2,i-1)$.

Proof. The proof is similar to Theorem 4.5. Using the partitioning technique that we showed before, in $S_2(n-1,k-1)$ the closest node to $e = 12...(k-1)k$ is at distance 2. Without loss of generality, let's assume $a = k1...(k-1)2$ is the source node in $S_2(n-1,k-1)$. We define set A as all nodes in $S_2(n-1,k-1)$ that are at distance $i-2$ from a (therefore, at distance i from e). Using Lemma 4.11, there are $g(n-1,k-1,i-1)$ many of these nodes and hence, $|A| = g(n-1,k-1,i-1)$.

Set A does not contain all nodes at distance i from e in $S_2(n-1,k-1)$.

For instance, in $S(5,3)$ node $b = 124$ is at distance 3 from e ($b = 124 \rightarrow 421 \rightarrow 321 \rightarrow e = 123$), however, b is not connected to $a = k1...(k-1)2$. Node b does not contain symbol k as if it does, it can not be at distance less than 4 from e. By a recursive approach, let's consider $S(n-2, k-2)$ with b as its source node and define the set B as all nodes in $S(n-2, k-2)$ that are at distance $i-3$ from b (at distance i from e). Note that this subgraph is inside $S_2(n-1, k-1)$.
There are nodes that belong to both sets A and B. For example, in $S(5,3)$, $u = 214$ is adjacent to $b = 314$ and therefore, there exists path p_1 as $u = 214 \rightarrow b = 314 \rightarrow 413 \rightarrow 213 \rightarrow e = 123$ and hence $u \in B$. u is also a neighbor of $v = 412$ which provides a path to e through node a and with the same length as p_1 ($u = 214 \rightarrow v = 412 \rightarrow a = 312 \rightarrow 213 \rightarrow e = 123$), Thus, $u \in A$.
The intersection is the set of all nodes in $S(n-2, k-2)$ that are also at distance $i-1$ from a. Hence, the cardinality of $A \cup B$, all nodes in $S(n,k)$ at distance i from e with last symbol 2 is:

$$|A \cup B| = |A| + |B| - |A \cap B|$$

$$= g(n-1, k-1, i-1) + g(n-2, k-2, i-3) - g(n-2, k-2, i-1)$$

□

4.3 Future Works

4.3.1 Forward Difference Property of $B_{A(n,k)}(i)$

In the previous section, we have proposed a recursive relationship for $B_{A(n,k)}(i)$. Tables 4.7, 4.8 and 4.9 show the value of $B_{A(n,k)}(i)$ for $n \in \{2, ..., 9\}$ and $k \in \{n-1, n-2, n-3\}$. According to these numbers, it seems that $B_{A(n,k)}(i)$ holds the forward difference property of order $i+1$. For example, in the $(n,n-2)$-arrangement graph, for $i=2$, the third forward difference is equal to

$$\sum_{j=0}^{3}(-1)^j\binom{3}{j}B_{A(n+3-j,k+3-j)}(2) =$$
$$\binom{3}{0}B_{A(7,5)}(2)-\binom{3}{1}B_{A(6,4)}(2)+\binom{3}{2}B_{A(5,3)}(2)-\binom{3}{3}B_{A(4,2)}(2) =$$
$$60-(3)36+(3)18-6 = 114-114 = 0$$

Also, using the recurrence in Theorem 4.6, the forward difference property of order 4 for $B_{A(n,k)}(3)$ is equal to

$$\sum_{j=0}^{4}(-1)^j\binom{4}{j}B_{A(n+4-j,k+4-j)}(3) =$$
$$\binom{4}{0}B_{A(9,7)}(3)-\binom{4}{1}B_{A(8,6)}(3)+\binom{4}{2}B_{A(7,5)}(3)-\binom{4}{3}B_{A(6,3)}(3)+\binom{4}{4}B_{A(5,3)}(3) =$$
$$\binom{4}{0}\left[(9-1)B_{A(8,6)}(2)+(6-1)(B_{A(7,5)}(0)-B_{A(7,5)}(2))+B_{A(8,6)}(3)\right]-$$
$$\binom{4}{1}\left[(8-1)B_{A(7,5)}(2)+(5-1)(B_{A(6,4)}(0)-B_{A(6,4)}(2))+B_{A(7,5)}(3)\right]+$$
$$\binom{4}{2}\left[(7-1)B_{A(6,4)}(2)+(4-1)(B_{A(5,3)}(0)-B_{A(5,3)}(2))+B_{A(6,4)}(3)\right]-$$
$$\binom{4}{3}\left[(6-1)B_{A(5,3)}(2)+(3-1)(B_{A(4,2)}(0)-B_{A(4,2)}(2))+B_{A(5,3)}(3)\right]+$$
$$\binom{4}{4}\left[(5-1)B_{A(4,2)}(2)+(2-1)(B_{A(3,1)}(0)-B_{A(3,1)}(2))+B_{A(4,2)}(3)\right] =$$
$$\left[8\times 90+5(1-60)+B_{A(8,6)}(3)\right]-$$
$$4\left[7\times 60+4(1-36)+B_{A(7,5)}(3)\right]+$$
$$6\left[6\times 36+3(1-18)+B_{A(6,4)}(3)\right]-$$
$$4\left[5\times 18+2(1-6)+B_{A(5,3)}(3)\right]+$$
$$\left[4\times 6+(1-0)+B_{A(4,2)}(3)\right].$$

Using the recurrence for required times, the values for $B_{A(8,6)}(3)$, $B_{A(7,5)}(3)$, $B_{A(6,4)}(3)$, $B_{A(5,3)}(3)$ and $B_{A(4,2)}(3)$ is 495, 250, 102, 27 and 1 respectively. By substituting these values in the above equation:

$$[8\times 90+5(1-60)+495]-4[7\times 60+4(1-36)+250]+$$
$$6[6\times 36+3(1-18)+102]-4[5\times 18+2(1-6)+27]+$$
$$[4\times 6+(1-0)+1] =$$
$$(720-295+495)-4(420-140+250)+6(216-51+102)-4(90-10+27)+(24+1+1) = 920-2120+1602-428+26 = 2548-2548 = 0.$$

Table 4.7: Values for $B_{A(n,n-1)}(i)$

(n, k)	$i = 0$	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$	$i = 7$	$i = 8$	$i = 9$	$i = 10$	$i = 11$	$i = 12$
(2, 1)	1	1											
(3, 2)	1	2	2	1									
(4, 3)	1	3	6	9	5								
(5, 4)	1	4	12	30	44	26	3						
(6, 5)	1	5	20	70	170	250	169	35					
(7, 6)	1	6	30	135	460	1110	1689	1254	340	15			
(8, 7)	1	7	42	231	1015	3430	8379	13083	10408	349	315		
(9, 8)	1	8	56	364	1960	8540	28994	71512	114064	96116	36260	4900	105

Table 4.8: Values for $B_{A(n,n-2)}(i)$

(n, k)	$i = 0$	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$	$i = 7$	$i = 8$	$i = 9$	$i = 10$
(3, 1)	1	2									
(4, 2)	1	4	6	1							
(5, 3)	1	6	18	27	8						
(6, 4)	1	8	36	102	152	58	3				
(7, 5)	1	10	60	250	680	1010	459	50			
(8, 6)	1	12	90	495	1960	5190	7749	4008	640	15	
(9, 7)	1	14	126	861	4480	17150	44709	67326	38464	7889	420

In fact, we have not found a counter example in the surface area of $A(n, k)$ graphs that does not hold the forward difference property of order $(i + 1)$. Thus, as an interesting topic to investigate, one can take the proposed recurrence for surface area of $A(n, k)$ (as one of the simplest and most generalized formula available so far) and verify the forward difference property of order $(i + 1)$ for $B_{A(n,k)}(i)$. This is an important topic since as we mentioned before, in the case that the property holds for $B_{A(n,k)}(i)$, the recurrence is a linear homogeneous equation with constant coefficients and order $(i+1)$ for which a known algorithm exists to solve them, resulting in a closed form solution.

4.3.2 Obtaining a Simple Recurrence for $B_{S(n,k)}(i)$

In Section 4.2, we have proved some properties for surface area of (n, k)-star graphs. The functions g and f that we defined before have the following relationship with each other:

Table 4.9: Values for $B_{A(n,n-3)}(i)$

(n,k)	$i=0$	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$	$i=7$	$i=8$	$i=9$
(4, 1)	1	3								
(5, 2)	1	6	12	1						
(6, 3)	1	9	36	63	11					
(7, 4)	1	12	72	246	404	102	3			
(8, 5)	1	15	120	610	1910	3030	969	65		
(9, 6)	1	18	180	1215	5620	16650	25929	9822	1030	15

Table 4.10: Values for $B_{S(n,n-1)}(i)$

(n,k)	$i=0$	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$	$i=7$	$i=8$	$i=9$	$i=10$	$i=11$	$i=12$
(2, 1)	1	1											
(3, 2)	1	2	2	1									
(4, 3)	1	3	6	9	5								
(5, 4)	1	4	12	30	44	26	3						
(6, 5)	1	5	20	70	170	250	169	35					
(7, 6)	1	6	30	135	460	1110	1689	1254	340	15			
(8, 7)	1	7	42	231	1015	3430	8379	13083	10408	3409	315		
(9, 8)	1	8	56	364	1960	8540	28994	71512	114064	96116	36260	4900	105

$g(n,k,i) = \sum_{j=1}^{n} f(n,k,i,j)$.

Therefore,

$g(n,k,i) = f(n,k,i,1) + \sum_{j=2}^{k-1} f(n,k,i,j) + f(n,k,i,k) + \sum_{j=k+1}^{n} f(n,k,i,j)$.

From Theorems 4.9, 4.10 and 4.12, $f(n,k,i,1)$, $f(n,k,i,k)$ and $\sum_{j=2}^{k-1} f(n,k,i,j)$ is already known. By Lemma 4.8, if a formula for $f(n,k,i,j)$ for any $k+1 \le j \le n$ can be found, a recurrence for $B_{S(n,k)}(i)$ will be obtained. Furthermore, based on values of $B_{S(n,k)}(i)$ for some n, k and i which is provided in Tables 4.10, 4.11 and 4.12, it appears that $B_{S(n,k)}(i)$ holds the forward difference property of order $(i+1)$. The next step after finding the recurrence then, could be verifying this property for (n,k)-star graphs.

Table 4.11: Values for $B_{S(n,n-2)}(i)$

(n,k)	$i=0$	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$	$i=7$	$i=8$	$i=9$	$i=10$	$i=11$
(3, 1)	1	2										
(4, 2)	1	3	4	4								
(5, 3)	1	4	10	21	22	2						
(6, 4)	1	5	18	57	119	130	30					
(7, 5)	1	6	28	118	368	774	911	302	12			
(8, 6)	1	7	40	210	870	2710	5849	7233	2960	280		
(9, 7)	1	8	54	339	1750	7220	22749	49992	64216	30551	4470	90

Table 4.12: Values for $B_{S(n,n-3)}(i)$

(n,k)	$i=0$	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$	$i=7$	$i=8$	$i=9$	$i=10$
(4, 1)	1	3									
(5, 2)	1	4	6	9							
(6, 3)	1	5	14	37	51	12					
(7, 4)	1	6	24	90	230	330	153	6			
(8, 5)	1	7	36	174	638	1634	2517	1545	168		
(9, 6)	1	8	50	295	1400	5110	13369	21632	15720	2835	60

Chapter 5

Conclusion

In this thesis, we have studied the node centered surface area of (n, k)-arrangement graphs and (n, k)-star graphs. These classes of graphs are interesting as an interconnection network since they address the scalability problem of star graph, $S(n)$ which itself is an attractive topology. In addition, they have other classes of graphs within their structure as for instance, when $k = n - 1$ it is the class of star graphs, when $k = n - 2$ it's the class of alternating group graphs and when $k = 1$, it is the class of complete graphs. Therefore, study of this generalized class of graphs is very beneficial as proving a property for these implies the proof for those special cases as well.

Obtaining a formula to compute the surface area of (n, k)-arrangement is not a trivial task. Despite numerous studies that have been conducted for this purpose, a closed form formula or a simple recurrence has not been proposed yet to compute the surface area of (n, k)-arrangement graphs. One of the remarkable works in this area is the recurrence that Portier and Vaughan have proposed which computes the surface area of $S(n)$ graphs in polynomial time in terms of the parameter n. $S(n)$ graph is a special case of (n, k)-arrangement graph when k is $n - 1$. In this research, we proposed a simple recurrence for surface area of (n, k)-arrangement graphs with linear computational complexity in regards to the parameters n and k which is a prodigious improvement over the naive solution. From the other hand, this recurrence is a generalization of the one that Portier and Vaughan proposed which works for all (n, k)-arrangement graphs for any arbitrary n and k in their defined range.

Moreover, we studied the surface area of (n, k)-star graphs and we achieved partial results that are useful in obtaining a simple recurrence for surface area of this graph. Also, throughout this research we have proved some properties of other interconnection networks that can be used in future studies. For example, in Chapter 3, we proved that the surface area of folded hypercubes holds the forward difference property of order $(i + 1)$.

One of the future directions would be verifying the forward difference property of order $(i + 1)$ for surface area of (n, k)-arrangement graphs using the proposed recurrence in this study. This is an important topic as if the surface area of (n, k)-arrangement graph holds this property, then it will be a homogeneous linear equation with constant coefficients for which a known algorithm exists to solve it and obtain the closed form.

As another trend of this thesis, one can take advantage of counting method used in Chapter 4 to count the number of nodes in partition $S_n(n-1, k-1)$ at distance i from the identity node in order to complete a simple recurrence for (n, k)-star graphs.

Bibliography

[1] J.M. Mellor-Crummey, M.L. Scott, Algorithms for scalable synchronization on shared-memory multiprocessors, ACM Transactions on Computer Systems (TOCS), Vol. 9, No. 1, (Feb. 1991), 21-65.

[2] S.V. Adve, K. Gharachorloo, Shared memory consistency models: A tutorial, Computer, Vol. 29, No. 12, (1996), 66-76.

[3] L.N. Bhuyan, D.P. Agrawal, Generalized hypercube and hyperbus structures for a computer network, IEEE Transactions on Computers, Vol. 33, No. 4, (1988), 393-400.

[4] D.S. Lee, J.L. Kalb , Network topology analysis, Sandia National Laboratories, (Jan. 2008).

[5] S.A. Choudum, V. Sunitha, Augmented cubes, Networks, Vol. 40, No. 2, (2002), 7184.

[6] M. Ma, G. Liu, J.M. Xu, Panconnectivity and edge-fault-tolerant pancyclicity of augmented cubes, Parallel Computing Archive, Vol. 33, No. 1, (2007), 36-42.

[7] S.B. Akers, D. Harel, B. Krishnamurthy, The star graph: An attractive alternative to the n-cube, Proceedings of the International Conference on Parallel Processing, (1987), 393-400.

[8] E. Cheng, K. Qiu, A generating function approach to the surface areas of some interconnection networks, Journal of Interconnection Networks, Vol. 10, No. 3, (2009), 189-204.

[9] K. Qiu, S.G. Akl, On some properties of the star graph, VLSI Design, Vol. 2, No. 4, (1995), 389-396.

[10] Q.P. Gu, S. Peng, Cluster faulttolerant routing in star graphs, Networks, Vol. 35, No. 1, (2000), 83-90.

[11] S. Sur, P.K. Srimani, A depth-first search routing algorithm for star graphs and its performance evaluation, Mathematical and Computer Modelling, Vol. 19, No. 9, (1994), 35-52.

[12] W.K. Chiang, R.J. Chen, On the arrangement graph, Information Processing Letters, Vol. 66, (1998), 215-219.

[13] K. Day, A. Tripathi, Embedding of cycles in arrangement graphs, IEEE Transactions on Computers, Vol. 42, (1993), 1002-1006.

[14] D.E. Knuth, The art of computer programming, Vol. 1, Massachusetts: Addison-Wesley, (1973).

[15] E. Cheng, K. Qiu, Z. Shen, On the surface area of the augmented cubes, The Journal of Supercomputing, Vol. 61, No. 3, (Sept. 2012), 856868.

[16] A. El-Amawy, S. Latifi, Properties and performance of folded hypercubes, IEEE Transactions on Parallel and Distributed Systems, Vol. 2, No. 1, (Jan 1991).

[17] E. Cheng, K. Qiu, Z. Shen, A proposed scheme of finding closed form solutions to surface areas of graphs, Congressus Numeratium, (2018).

[18] S.P. Dandamudi, D.L. Eager, Hierarchical interconnection networks for multicomputer systems, IEEE Transactions on Computers, Vol. 39, No. 6, (Jun. 1990), 786-797.

[19] P.W. Dowd, K. Jabbour, Performance evaluation of spanning multiaccess channel hypercube interconnection network, IEEE Proceedings (Computers and Digital Techniques), Vol. 134, No. 6, (Nov. 1987), 295302.

[20] S.P. Dandamudi, D.L. Eager, On hypercube-based hierarchical interconnection network design, Journal of Parallel and Distributed Computing, Vol. 12, No. 3, (Jul. 1991), 283-289.

[21] W.K. Chiang, R.J. Chen, The (n, k)-star graph: A generalized star graph, Information Processing Letters Vol. 56, (1995), 259 - 264.

[22] G. Fertin, A. Raspaud, k-Neighborhood broadcasting, International Colloquium on Structural Information and Communication Complexity, (2001), Spain.

[23] M. Chakraborty, S. Chowdhury, J. Chakraborty, et al., Algorithms for generating all possible spanning trees of a simple undirected connected graph: an extensive review, Complex Intelligent Systems, (2018), 117.

[24] K. Day, A. Tripathi, Arrangement graphs: a class of generalized star graphs, Information Processing Letters, Vol. 42, (1992), 235-241.

[25] F. Portier, T. Vaughan, Whitney numbers of the second kind for the star poset, European Journal of Combinatorics, Vol. 11, (1990), 277288.

[26] L.Wang, S. Subrammanian, S. Lati, et al., Distance distribution of nodes in star graphs, Applied Mathematics Letters, Vol. 19, (2006), 780784.

[27] N. Imani, H. Sarbazi-Azad1, S.G. Akl, Some combinatorial properties of the star graphs: the surface area and volume, Discrete Mathematics, Vol. 309, (2009), 560569.

[28] E. Cheng, K. Qiu, Z. Shen, Length two path centered surface areas of the (n, k)-star graph, Information Sciences, Vol. 332, (2016), 115-130.

[29] G. Fertin, A. Labarre, I. Rusu, et al., Combinatorics of Genome Rearrangements, Vol. 1, Cambridge, Massachusetts: The MIT Press, (2009).

[30] E. Cheng, K, Qiu, Z. Shen, The edge-centered surface area of the arrangement graph, Combinatorial Optimization and Applications, vol 7402, (2012), 49-60.

[31] E. Cheng, K. Qiu, Z. Shen, On deriving explicit formulas of the surface areas for the arrangement graphs and some of the related graphs, International Journal of Computer Mathematics, Vol. 87, No. 87, (2010), 29032914.

[32] Z. Shen, K. Qiu, An explicit formula of the surface area for the star graph and a proof of its correctness, Congressus Numerantium, Vol. 192, (2008), 115127.

[33] Z. Shen, K. Qiu, On the Whitney numbers of the second kind for the star poset, European Journal of Combinatorics, Vol. 29, (2008), 1585-1586.

[34] E. Cheng, K. Qiu, Z. Shen, On deriving explicit formulas of the surface areas for the arrangement graphs and some of the related

graphs, International Journal of Computer Mathematics, Vol. 87, No. 13, (2010), 2903-2914.

[35] Z. Shen, K. Qiu, E. Cheng, On the surface area of (n, k)-star graph, Theoretical Computer Science, Vol. 410, No. 52, (2009), 5481-5490.