Study of Cubelike Graphs for Parallel and Quantum Computation

by

RISHIKANT RAJDEEPAK
201521006

A Thesis Submitted in Partial Fulfilment of the Requirements for the Degree of

DOCTOR OF PHILOSOPHY
to

DHIRUBHAI AMBANI INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGY

April, 2022

Declaration

I hereby declare that

i) the thesis comprises of my original work towards the degree of Doctor of Philosophy at Dhirubhai Ambani Institute of Information and Communication Technology and has not been submitted elsewhere for a degree,

ii) due acknowledgement has been made in the text to all the reference material used.

Rishikant Rajdeepak

Rishikant Rajdeepak

Certificate

This is to certify that the thesis work entitled STUDY OF CUBELIKE GRAPHS FOR PARALLEL AND QUANTUM COMPUTATION has been carried out by RISHIKANT RAJDEEPAK for the degree of Doctor of Philosophy at *Dhirubhai Ambani Institute of Information and Communication Technology* under our supervision.

V. Sunitha

Dr. V. Sunitha
Thesis Supervisor (joint)

Dr. Jaideep Mulherkar
Thesis Supervisor (joint)

Acknowledgements

I am very grateful to my supervisors, Prof. V. Sunitha and Prof. Jaideep Mulherkar, for their guidance and motivation all these years. I want to thank Prof. Rahul Muthu and Prof. Gautam Dutta for patiently reviewing my research work and suggesting a few ideas from time to time. I want to thank all members of DA-IICT, both academic and non-academic staff, for providing commending environment for the smooth flow of my Ph.D. courses. I thank my colleagues for boosting my interest in research.

I thank my friends Rahul, Pankaj, Madhu, Nikita, and Purvi for sharing their time and stories and creating memorable moments. I thank another set of close friends, Jignesh, Kamal, Sumukh, Prashant, and Parth, for their presence in almost every moment, like cooking food, sports, and movies. All these moments kept me going all these years.

Last but not least, I would like to thank my mummy and papa, bhaiya and bhabhi, didi, and my wife (Nisha Kumari), who support and believe in me in pursuing my dreams.

Contents

Abstract

In this thesis, we study Cayley graphs over $\mathbb{Z}_r^n$ for their utility in multiprocessor computing and quantum computing. For multiprocessor computing, we investigate problems of graph embedding on cubelike interconnection networks such as hypercubes and augmented cubes. These embeddings are required for efficient simulation of divide-and-conquer based algorithms on multiprocessor systems built on such interconnection networks. In particular, we discuss Havel's conjecture which states that an equibipartite binary tree on 2^n vertices spans the n-dimensional hypercube. We develop an efficient embedding technique to prove the conjecture for a subfamily of equibipartite binary tree. We also worked on a related conjecture which states that a binary tree on 2^n vertices spans the n-dimensional augmented cube. For this, we propose a recursive technique to embed and a method to count the spanning binary trees of the augmented cube. For exploring the utility of cubelike graphs for quantum computation, we study quantum walks that can be used to develop quantum algorithms for searching and communication. We study both theoretical and experimental aspects of quantum walks on cubelike graphs as well as Cayley graphs over $\mathbb{Z}_r^n$. For discrete-time quantum walk, our work gives a closed form for the quantum state of the system associated with cubelike graphs, after finitely many time steps; this work is the key to studying hitting times on the graphs which is a measure of how quickly the walker reaches a specific target node. We also decompose the quantum circuits of the corresponding evolution operators that were run on IBM's quantum computing platform. We conjecture that there is a linear relation between the quantum hitting times and dimension of cubelike graphs. For continuous-time quantum walk, we investigate weighted Cayley graphs over $\mathbb{Z}_r^n$ in order to classify them

into three categories of graphs - those that admit PST, those that do not admit PST but are periodic, and those that are not periodic. In continuation to this work, we constructed quantum circuits of the evolution operators for CTQW on weighted cubelike graphs.

List of Tables

List of Figures

CHAPTER 1

Introduction and Motivation

Graphs are among the most powerful mathematical tools used to model many practical problems in basic sciences and engineering, computer science, social science, network science, and linguistics. In the world of computing, it gives a visual interpretation of algorithms and helps in analyzing them. Effective use of properties of graphs improves the design of network structure, which speeds up computation and optimizes memory. A well-known method of using graphs is via the stochastic or random process called random walks on graphs. Random walks on graphs are used to analyze and simulate the randomness of objects and describe their statistical properties. Some typical applications of random walks are diffusion of information, generating random samples, measuring the characteristics of the world wide web and social networks, and searching problems in a database. One of the recent applications of graphs is in quantum computation, which is based on quantum walks on graphs. A quantum walk is a quantum counterpart of (classical) random walk, where the underlying graph describes the evolution of the associated quantum system. Its algorithmic applications are recent, and their implementations are very challenging. The central idea in quantum computing is to build quantum algorithms that may outperform their classical counterparts. A comparison of algorithms based on quantum walks and random walks are discussed in [68].

In this thesis, our work is closely related to hypercubes and its variants. We study Cayley graphs over the direct product of n copies of a cyclic group, $\mathbb{Z}_r^n$, for their utility in multiprocessor computing and quantum computing. For multiprocessor computing, we investigate problems of graph embedding on cubelike

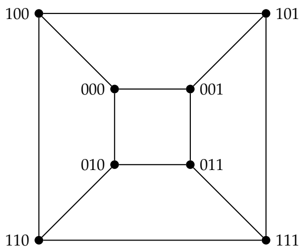


Figure 1.1: The 3-dimensional hypercube.

interconnection networks such as hypercubes and augmented cubes. In quantum computation, we study about quantum walks on hypercubes. We also study quantum walks on Cayley graphs which is a generalization of cubelike graphs.

1.1 Hypercube multiprocessor systems and graph embeddings

An n-dimensional hypercube is a graph whose vertices are represented by binary strings of length n and two vertices are adjacent if their binary representations differ at exactly one position (see Fig. 1.1). A multiprocessor system whose architecture is based on hypercube topology is widely used on parallel computing machines. Hypercubes serve as efficient interconnection networks because of their low degree of regularity, logarithmic distance, and the existence of many edge-disjoint paths between a pair of nodes, which help develop adaptive and fault-tolerant parallel algorithms. The vertex-symmetry (and edge-symmetry) of hypercubes allow interchanging the roles of a vertex (or an edge) with proper permutations of vertices (or edges). Some of the typical computational problems solved by hypercube algorithms are sorting, merging, parallel prefix computation, and matrix multiplication.

Hypercube machines are capable of emulating other interconnection networks such as trees, rings, torus, and meshes with minimum overhead. This simulating problem is a graph embedding problem that deals with the portability of algorithms from one architecture to another. The recursive structure of hypercubes is appropriate for running recursive or divide-and-conquer algorithms. Many par-

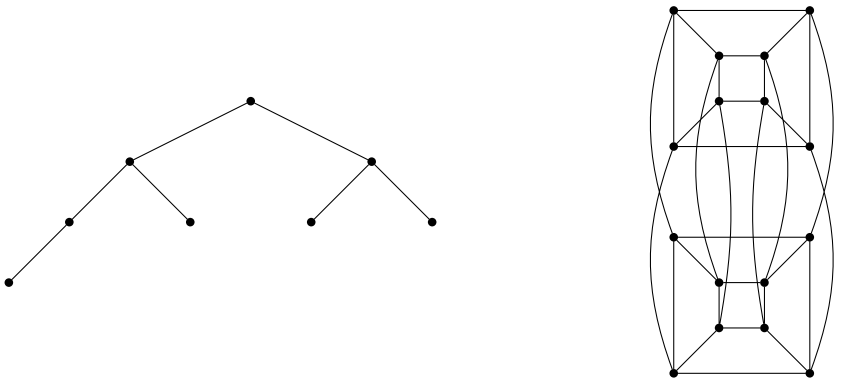


Figure 1.2: A complete binary tree (left) and 4-dimensional hypercube (right).

allel algorithms adopt binary tree structures for communication, and therefore it is important to study their embeddings into hypercubes. In general, graph embeddings do not respect the structure of a guest graph into the host graph, i.e., an edge in the guest graph can be mapped to a path in the host graph. In this thesis, an embedding is restricted to an injective graph homomorphism.

Each binary tree is embeddable into a hypercube of dimension n, for some n, where the n-dimensional hypercube has 2^n nodes. However, the optimum benefit of an embedding is obtained if the dimension of the hypercube is minimum. For instance, the complete binary tree (see Fig. 1.2) on 8 vertices is embeddable into the n-dimentional hypercube for $n \geq 4$. A better approach would be finding a subclass of binary trees on m vertices, with $2^{n-1} < m \leq 2^n$, that covers most of the computational problems based on binary trees and are embeddable into the hypercube of dimension n. One such subfamily is the equibipartite binary trees, i.e., the binary trees whose vertex-set can be partitioned into two sets of equal size such that each part contains non-adjacent vertices, i.e., the bipartition of these binary trees are of equal size.

1.2 Random walks and quantum walks on hypercubes

A random walk on a graph is described by a Markov chain, where nodes denote possible states of the associated state space and the probability of the walker to move from a given node to any of its neighbors is non-zero and to a non-adjacent node is zero. It has revolutionized the study of classical algorithms by providing techniques to create better approximation algorithms for PageRank and 2-SAT problems. A quantum generalization to a random walk is a quantum walk on a graph. Quantum walks based algorithms have the potential to speed up computational problems compared to random walks based algorithms because of the unique properties of quantum parallelism equipped with quantum interference and entanglement.

Quantum walks are broadly of two types, discrete-time coined quantum walk (DTQW) and continuous-time quantum walk (CTQW). In [62], Shenvi et al. describe a quantum search algorithm based on DTQW on hypercubes that perform an oracle search on a database of $N = 2^n$ items with $\mathcal{O}(\sqrt{N})$ calls to the oracle. Routing based on DTQW on regular graphs has potential applications in quantum information theory [70]. A CTQW based algorithm is developed by Campos et al. that solves a hard k-SAT problem [19], where the underlying graph is a hypercube. Another important feature is mixing time of random walk, whose quantum counterpart is discussed in [4, 33, 57].

In this thesis, we focus on well studied properties of quantum walks on graphs, quantum hitting times and perfect state transfer. The quantum hitting time of the DTQW on the n-dimensional hypercube from a node to its antipodal node is linearly proportional to n [49], whereas in the classical case the hitting time is approximately 2^n [55]. A good study on perfect state transfer of CTQW on cube-like graphs can be found in [14, 22]. An implementation of quantum walks on hypercubes are discussed in [32, 64].

1.3 Cayley graph structure of hypercubes

The n-dimensional hypercube represents the abstract structure of the Boolean group $\mathbb{Z}_2^n$, viz., two vertices are adjacent if their XOR sum has exactly one nonzero bit. A Cayley graph is a generalization to this notion that represents the abstract structure of a group. A Cayley graph is a graph defined over a pair (G, Ω), where G is a finite group and Ω is a generating set of G, with the properties: (1) Ω does not contain the identity element, and (2) Ω is closed under the group in-

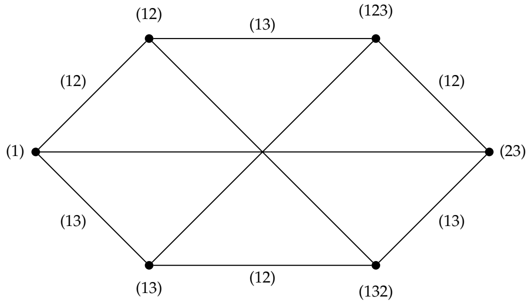


Figure 1.3: $Cay(Sym(3), \{(12), (13), (23)\})$.

verse, i.e., $x^{-1} \in \Omega$ for all $x \in \Omega$. The Cayley graph, denoted by $Cay(G, \Omega)$, is a graph whose vertices are the elements of G and the edge set is given by $\{(x, y) : xy^{-1} \in \Omega\}$.

The Cayley graph is connected because the generating set Ω defines each edge. If it is assumed that Ω does not generate G, then the Cayley type graph constructed as above is disconnected, and each component is a coset of the subgroup generated by Ω. For any vertex x, the cardinality of $\{xa : a \in \Omega\}$ is equal to that of Ω; hence the graph is regular with $|\Omega|$ being the degree of regularity. The graph is vertex-transitive, i.e., for every pair of vertices $\{x, y\}$ there is a graph isomorphism that maps one to the other. The Cayley graph defined over the symmetric group of order three, $Sym(3)$, with the generating set consisting of three permutations $\{(12), (13), (23)\}$ is a connected 3-regular graph (see Fig 1.3).

An essential subfamily of Cayley graphs is defined over the direct product of n copies of a cyclic group $\mathbb{Z}_r^n$, where r and n are positive integers. The case $r = 2$ defines the cubelike graph $Cay(\mathbb{Z}_2^n, \Omega)$ over the Boolean group $\mathbb{Z}_2^n$. The n-dimensional hypercube is a cubelike graph with the generating set $\Omega = \{0^i10^j : i + j = n - 1\}$. An important variant to the n-dimensional hypercube is the n-dimensional augmented cube whose generating set consists of $\Omega = \{0^i10^j : i + j = n - 1\} \cup \{0^{n-i}1^i : 1 \leq i \leq n\}$.

1.4 Contribution of the thesis

This thesis is divided into two parts; we study Cayley graphs from the perspective of multiprocessor computing and quantum computing in these parts. In particular, Part I discusses embedding problems in interconnection networks based on hypercube topology or augmented cube, and Part II describes quantum walks on cubelike graphs. More specifically;

- Part I is divided into two chapters. In chapter 2, we study some properties of spanning trees and their relation with linear algebra. We enumerate spanning binary trees of a graph using a couting method and a dynamic data structure. We have used these approaches to verify a conjecture about spanning binary trees of augmented cubes. In chapter 3, we present an efficient embedding algorithm that maps a subfamily of equibipartite binary trees on 2^n vertices into the n-dimensional hypercube via an injective graph homomorphism.

- Part II is divided into two chapters. Chapter 4 addresses theoretical and experimental aspects of discrete-time coined quantum (DTQW) walks on cubelike graphs. In section 4.1, we give an expression for a generic quantum state after finitely many evolutions that is essential to compute hitting times in DTQW. In section 4.3, we decompose quantum circuits for DTQW on cubelike graphs that were run on IBM's quantum computing platform, and based on observations, we conjecture about the linear relation between hitting times and dimension of cubelike graphs. This work is an extension of the work by Kempe [49]. In chapter 5, we study continuous-time quantum walks (CTQW) on Cayley graphs over $\mathbb{Z}_r^n$. In section 5.2, we study properties of normal matrices, which are used to define a family of graphs having the same eigenvectors; in particular, weighted Cayley graphs having the same eigenvectors are constructed. In section 5.3, we investigate weighted Cayley graphs over $\mathbb{Z}_r^n$ in order to classify them into three categories of graphs - those that admit PST, those that do not admit PST but are periodic, and those that are not periodic. This work generalizes the work by Cheung and Godsil [22]. In section 5.5, we decompose quantum circuits for the evolution operator of CTQW on weighted cubelike graphs and verified the existence of perfect state transfer in them.

Part I

Embedding Binary Trees into Cubelike Interconnection Networks

CHAPTER 2

Spanning trees of a graph

A spanning tree of a connected graph Γ is a subgraph containing $N-1$ edges and no cycles. If an edge e is not present in a spanning tree T of Γ, then there is a unique cycle containing the edge e and edges in T only. On the other hand, if e is an edge in T, then there is a unique cut of Γ containing e and edges not in T.

2.1 The incidence matrix

Let Γ be a connected graph on N vertices. We assign an orientation to Γ by assigning arbitrary directions to the edges of the graph. The incidence matrix of the oriented graph Γ is an $N \times M$ matrix, where rows correspond to N vertices and columns correspond to M edges. The incidence matrix $\mathcal{I}$ is defined by;

$$\mathcal{I}_{xy} = \begin{cases} +1 & ;\ \text{if } x \text{ is the head of the edge } xy \\ -1 & ;\ \text{if } y \text{ is the tail of the edge } xy \\ 0 & ;\ \text{if } xy \text{ is not an edge.} \end{cases} \tag{2.1}$$

The determinant of any square submatrix of $\mathcal{I}$ is 0 or -1 or $+1$. Notice that each column has exactly two non-zero entries corresponding to an edge and the sum of all rows in $\mathcal{I}$ is the zero row vector $\mathbf{0}$. Suppose the linear combination of rows of $\mathcal{I}$ is zero, i.e., $\sum_x \lambda_x r_x = \mathbf{0}$, then $\lambda_x = \lambda_y$ if (x, y) is an edge. Thus, if there exists a path from a vertex x to another vertex z, then $\lambda_x = \lambda_z$. Since the graph Γ is connected, the linear combination $\sum_x \lambda_x r_x$ is the multiple of $\sum_x r_x$ and hence the rank of $\mathcal{I}$ is $N-1$. The incidence matrix and spanning trees of the graph Γ are related by the following result.

Lemma 1. *[17] Let Γ be a connected graph on N vertices, and U be a subgraph containing $N-1$ edges. Let $\mathcal{I}_U$ be a $(N-1)\times(N-1)$ submatrix of D whose columns correspond to edges in U and contains any $(N-1)$ rows. Then, $\mathcal{I}_U$ is invertible if and only if the subgraph U is a spanning tree of Γ.*

Proof. If U is a spanning tree, then it is a connected graph on N vertices, which implies its incidence matrix has rank $N-1$; thus $\mathcal{I}_U$ is invertible. Conversely, if $\mathcal{I}_U$ is invertible, then the incidence matrix of U has rank $N-1$, which implies U is connected. Thus, U is a tree on $N-1$ vertices and it spans the graph. □

2.1.1 Generating spanning trees

Each set of $N-1$ linearly independent columns of $\mathcal{I}$ corresponds to a unique spanning tree. The following algorithm generates a spanning tree of a graph.

Algorithm 1. *Let the input be an incidence matrix $\mathcal{I}$ of a connected graph Γ. Then, the following steps output maximal linearly independent column vectors.*

(i) Select a column.

(ii) For each selected column c mark the two rows of $\mathcal{I}$ corresponding to non-zero entries of c.

(iii) If there is a column c that has a marked row corresponding to its non-zero entry and the other is not marked, then select the column and go to step (ii); else go to step (iv).

(iv) Output all selected columns and end the algorithm.

Proof. We prove the correctness of the algorithm by induction on the number m of columns selected. If $N=1$, then $m=0$, and if $N=2$, then $m=1$. Assume that $N>2$. For $m=1\leq N-1$, a single column vector is linearly independent. Suppose, a set of $m>1$ columns are linearly independent and $m<N-1$, then, since the graph is connected there exists a required column in step (iii), which is mutually linearly independent of the selected columns. Thus, the step (iii) gives

$m+1$ linearly independent set of columns. Since, the number of rows are finite the algorithm terminates after a finite number of steps. □

Let C be the set of column vectors of $\mathcal{I}$ with the lexicographic order. Let $\mathcal{T}$ be the set of ordered subset of C, each of fixed size $N-1$. Under the lexicographic order of $\mathcal{T}$, spanning trees of the graph can be obtained linearly without duplication. This approach will enumerate all spanning trees.

Example 1. *Consider the Cayley graph on the symmetric group over three elements, with the generating set $\{(12),(13),(23)\}$, see Fig. 1.3. Let the incidence matrix of the graph be*

$$\mathcal{I} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 1 \\ 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & -1 \end{bmatrix}$$

Applying algorithm 1, the sequence of $N-1$ linearly independent column vectors obtained is $\{(1,2),(1,4),(1,6),(2,3),(2,5)\}$.

2.1.2 The tree-number

Sometimes we are interested in counting total number of spanning trees of a graph without knowing the exact form of spanning trees. The matrix-tree theorem gives the tree number $\kappa(\Gamma)$ of the graph Γ, where the tree number is the total number of spanning trees. The Laplacian matrix L is defined by $L = \mathcal{I}\mathcal{I}^T$, which is also equal to $D - A$, where D is the diagonal matrix whose diagonal entry corresponding to a vertex is its degree and A is the adjacency matrix.

Theorem 1. *[17] Let L be the Laplacian matrix of a graph. Then, every cofactor of L is equal to the tree-number of the graph.*

Theorem 2. *[17] The tree-number of a graph on N vertices is given by*

$$\kappa = \frac{\mid J + L \mid}{N^2},$$

where L is the Laplacian matrix and J is the matrix with all entries equal to 1.

If we know the list of eigenvalues of the Laplacian matrix, then we can calculate the tree-number using the following result.

Corollary 1. *[17] Let* $0 \leq \mu_1 \leq \cdots \leq \mu_{N-1}$ *be the eigenvalues of the Laplacian matrix of a graph with N vertices. Then, the tree-number is given by*

$$\kappa = \frac{\mu_1 \mu_2 \cdots \mu_{N-1}}{N}. \tag{2.2}$$

Notice that if the graph is not-connected then $\mu_1 = 0$ and it has zero spanning tree, and if the graph is connected then $\mu_1 > 0$. The following result relates the eigenvalues of the adjacency matrix (called the spectra of the graph) with the eigenvalues of the Laplacian matrix.

Theorem 3. *[17] Let* Γ *be a r-regular graph on N vertices with the Laplacian matrix L and the adjacency matrix A. Suppose* $\mu_0 \leq \mu_1 \leq \cdots \leq \mu_{n-1}$ *are the eigenvalues of L and* $\lambda_0 \geq \lambda_1 \geq \cdots \geq \lambda_{N-1}$ *are the eigenvalues of A, then*

$$\mu_i + \lambda_i = r.$$

Remarks

We have seen a method to generate spanning trees of a graph, sequentially, using an incidence matrix. If the graph is not connected then algorithm 1 generates a tree that spans a component of the graph. Applying the algorithm 1 to each component generates a spanning forest. The tree-number of an unconnected graph is zero; we can instead, compute the number of spanning forests by computing tree-number for each component and multiplying them.

2.2 Graph homomorphism

A graph homomorphism is a function ϕ from a guest graph G to a host graph H that preserves the adjacency structure of G, i.e., if (x, y) is an edge in G, then $f(x)$

is adjacent to $f(y)$. In this section, we will study the application of graph homomorphism to degree constraint spanning trees (DCST) problem, i.e., we answer the question of whether for a given degree sequence s there exists a spanning tree of G. We use $hom(G, H)$, $inj(G, H)$, and $sub(G, H)$ to denote the set of homomorphisms from G to H, the set of injective homomorphisms from G to H, and the set of distinct copies of G in H, respectively.

Definition 1. *A tree decomposition of a graph G is a mapping of G into a tree D_G such that*

(i) the vertex-set of D_G is a collection of subsets of vertex-set of G,

(ii) each vertex of G is present in all vertices of a subtree of D_G,

(iii) each pair of adjacent vertices are present in a vertex of D_G.

It is clear that the union of all vertices of D_G equals the vertex-set of G. The width of a tree decomposition D_G is one less than the maximum cardinality of a vertex, i.e., $max\{\mid X \mid -1 \ : \ X \in V(D_G)\}$. The treewidth of a graph G is the minimum width over all the tree decompositions of G, denoted by $tw(G)$. For example, the treewidth of a tree, a cycle, and a complete graph on N vertices have treewidth one, two, and $N - 1$, respectively. A chordal graph has treewidth equal to the size of the largest clique minus one.

2.2.1 Counting subgraphs

We will discuss a method to count injective homomorphism from a graph G to another graph H that will allow us to compute the number of subgraphs given by

$$\mid sub(G, H) \mid = \frac{\mid inj(G, H) \mid}{\mid aut(G, H) \mid}, \tag{2.3}$$

where $aut(G, G)$ is the set of automorphism (bijective isomorphism from a graph to itslef) of G. The size of $aut(G, G)$ can be computed in time $2^{\sqrt{\mathcal{N} \log \mathcal{N}}}$ [10, 21].

Theorem 4. *[7] Let G and H be two graphs on N vertices. Then,*

$$| \, inj(G, H) \, |= \sum_{W \subset V(G)} (-1)^{|W|} \, | \, hom(G, H[W']) \, |, \tag{2.4}$$

where $H[W']$ is the subgraph of of H induced by the complement of W.

Theorem 5. *[34] Let G and H be two grapsh on N and M vertices, respectively. If there is a tree decomposition of G with width t, then $| \, hom(G, H) \, |$ can be computed in time $\mathcal{O}(N \cdot M^{t+1} min\{t, M\})$ and space $\mathcal{O}(log \; N \cdot M^{t+1})$.*

Theorem 6. *[60] The number of unlabelled trees on N vertices is asymptotically $C\alpha^n N^{-\frac{5}{2}}$, where $C = 0.53495\ldots$ and $\alpha = 2.95576\ldots$.*

Theorem 7. *[15] All unlabelled rooted trees of size N can be generated in time $C\alpha^n N^{-\frac{5}{2}}$, where $C = 0.53495\ldots$ and $\alpha = 2.95576\ldots$.*

Theorem 8. *[7] Let G be a graph on N vertices and $s = (s_1, s_2, \ldots, s_N)$ be a sequence of N positive values. Then, the number of spanning trees of G with i-th vertex of degree at most s_i for all i, can be computed in time $\mathcal{O}(5.912^N)$.*

Remarks

The counting method via graph homomorphism is applicable to embedding binary trees into an n-dimensional cubelike graph Γ. Since the treewidth of a tree T on $N = 2^n$ vertices is one, the size of $hom(T, \Gamma)$ is computable in time $\mathcal{O}(2^{3n})$, by Theorem 5. Consider the hypercube $\mathcal{Q}_n$ or the augmented cube $\mathcal{AQ}_n$, and enumerate all unordered trees on $N = 2^n$ by using Theorem 7. Then after, we compute $| \, sub(T, \mathcal{Q}_n) \, |$ or $| \, sub(T, \mathcal{AQ}_n) \, |$ by using Theorem 8, where the degree sequence is bounded by $s = (3, 3, \ldots, 3)$.

2.3 Dynamic data structures

A dynamic graph algorithm is useful in answering the question of connectivity, bipartiteness, 2-edge connectivity and minimum spanning trees. It works on the framework of dynamic data structures that aid graph operations such as insertion and deletion of an edge or a vertex. Various data structures are suitable for efficient graph operations such as Biased search trees, Euler tour trees (ET-trees) and Top trees [13, 45, 47]. To generate spanning trees of a connected and unweighted graph Γ, we require three types of graph operations, viz., (1) deletion of an edge, (2) finding a replacement edge, and (3) insertion of an edge. Suppose T is a spanning tree, then its edges are called tree edges and other edges are called non-tree edges. We can choose any of the data structures mentioned before to store the spanning tree T along with additional information about edges which are useful in finding a replacement edge efficiently.

2.3.1 Enumeration of spanning trees

Suppose Γ is a connected and unweighted graph on N vertices. Define a graph $\mathcal{T}(\Gamma)$, where the node-set is the set of all spanning trees and two spanning trees are adjacent if one can be obtained from the other by exchaning exactly one edge. The main idea to enumerate all spanning trees is to find the spanning tree of the graph $\mathcal{T}(\Gamma)$. We obtain the first spanning tree T_0 by applying depth-first search on the graph Γ and label the vertices and edges in the same order they are visited during the search, viz., the vertices are labeled as $v_1 < v_2 < \cdots < v_N$ and the edges are labeled as $e_1 < e_2 < \cdots < e_{N-1}$. The remaining edges are labeled lexicographically and hence the node-set of $\mathcal{T}(\Gamma)$ can be ordered in increasing lexicographic order. Beginning at T_0 we apply breath-first search on $\mathcal{T}(\Gamma)$ to obtain its spanning tree $\mathcal{D}$. During the search the following operations are performed (see Fig. 2.1 and Fig. 2.2);

(i) For the current node T the next least child T_c of T is obtained by deleting the highest index edge $e_k \in T \cap T_0$ for which the replacement edge e_l exists such that $e_l \in T_0^c$ is the next least index edge, i.e. if e_i was the previous replacement edge for e_k then $e_i < e_l$. Visit T_c and repeat the process.

(ii) If we do not get any replacement edge then we delete the next highest index edge $e_j \in T \cap T_0$ such that $e_j < e_k$ and continue the search.

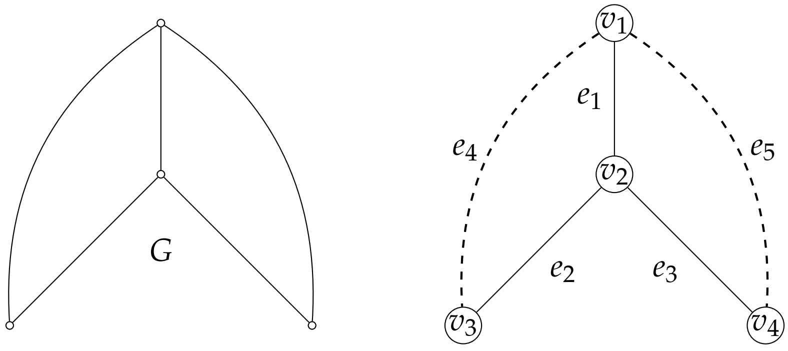


Figure 2.1: T^0 obtained by depth-first search

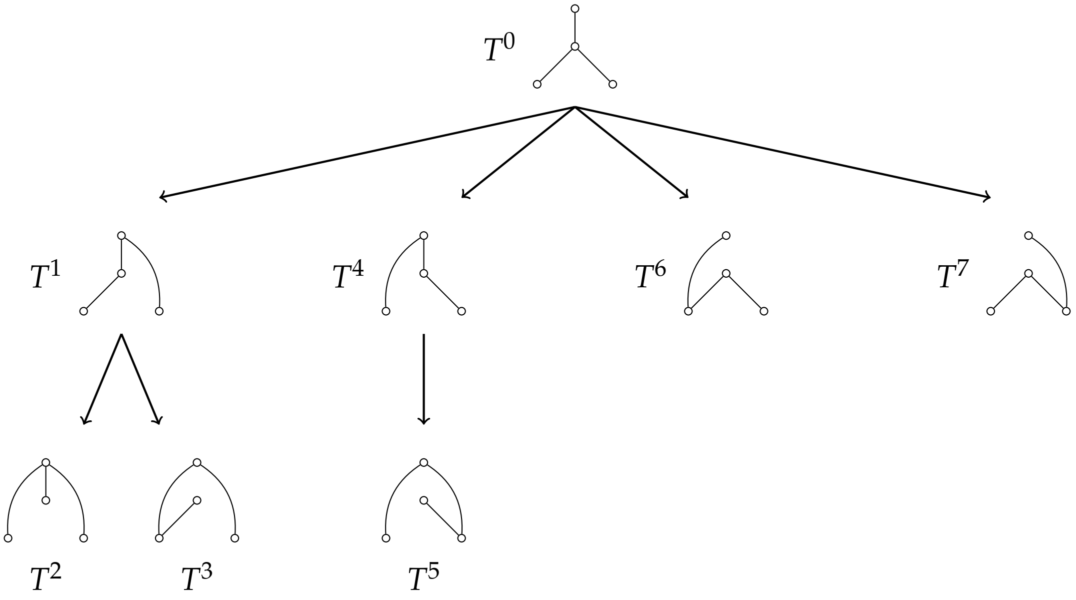


Figure 2.2: Depth-first search in $\mathcal{T}(G)$.

(iii) If no replacement edge for any edge in $T \cap T_0$ is found then we visit its parent T^p and continue the process such that if $e \in T^p$ was replaced with $f \in T$ then the search starts by deleting $e \in T^p$ and finding the least index replacement edge $g > f$ in T_0^c.

(iv) If the current node is T_0 and no child is found then we stop algorithm.

Remarks

The enumerating technique is efficient, discussed in [63] by Shioura et al., which can be improved if we consider poly-logarithmic deterministic dynamic algorithms from [47]. Out objective was to use these algorithms to identify the number of spanning binary trees of the n-dimensional augmented cubes and compare it with the number of binary trees on 2^n vertices. For small values of n, we could run the algorithm and found that the two numbers coincide; thereby verifying for these values of n the conjecture that binary trees span augmented cubes.

CHAPTER 3

Embedding binary trees into hypercubes and augmented

Graph embeddings over hypercubes are among the most studied problems in the interconnection network. In this chapter, we discuss about embedding a subfamily of binary trees into the hypercubes, where the recursive property of the hypercubes is used. An n-dimensional hypercube, denoted by $\mathcal{Q}_n$, is defined recursively, as follows; Q_1 is the complete graph on two vertices, with vertices labeled by 0 and 1, and Q_n is defined as the Cartesian product of Q_{n-1} and Q_1, denoted by $Q_{n-1} \times Q_1$, where,

- a vertex a in Q_1 and a vertex x in Q_{n-1} concatenate to form the vertex ax in Q_n,
- two vertices, ax and by, are adjacent in Q_n if either $a = b$ and (x, y) is an edge in Q_{n-1} or $x = y$ and (a, b) is an edge in Q_1.

This definition helps in breaking down a task graph into two subtasks and assigning them to two disjoint subcubes Q_{n-1} in such a way that the communication overhead is low. Augmented cubes have the same potential as hypercubes and have similar recursive definition. The 1-dimensional augmented cube $\mathcal{AQ}_1$ has two vertices $\mathbb{Z}_2 = \{0, 1\}$ and one edge $(0, 1)$. For $n > 1$, the n-dimensional augmented cube $\mathcal{AQ}_n$ is obtained from the $(n-1)$-dimensional augmented cube $\mathcal{AQ}_{n-1}$ as follows;

(i) Take two copies of $\mathcal{AQ}_{n-1}$ and label them by $\mathcal{AQ}_{n-1}^0$ and $\mathcal{AQ}_{n-1}^1$. Relabel each vertex $x = x_{n-1} \dots x_2 x_1$ in $\mathcal{AQ}_{n-1}^0$ by $0x = 0x_{n-1} \dots x_2 x_1$, and each

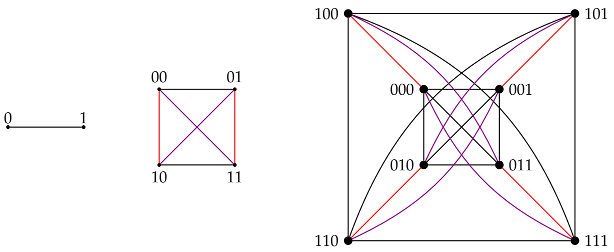


Figure 3.1: A recursive construction of $\mathcal{AQ}_3$.

vertex x in $\mathcal{AQ}^1_{n-1}$ by $1x = 1x_{n-1} \dots x_2x_1$.

(ii) For each vertex $0x \in \mathbb{Z}^n_2$, $\mathcal{AQ}_n$ has two additional edges $(0x, 1x)$ and $(0x, 1\bar{x})$, see Fig. 3.1.

In this chapter, we talk about an embedding technique based on recursive method.

3.1 An embedding conjecture

In 1984, Havel [43] conjectured that an equibipartite binary tree on 2^n $(n \geq 1)$ vertices is a spanning tree of the n-dimensional hypercube. It attained attention of researchers after Havel and Liebl [44] proved the result for equibipartite binary caterpillars, wherein a caterpillar is a tree such that if all leaves are removed then the remaining subgraph is a path. Such caterpillars are also called one-legged caterpillars where legs are its leaves. In [16] it is shown that binary caterpillars with each leg of same parity is a subgraph of its optimal hypercube, where, for a graph on m ($2^{n-1} < m \leq 2^n$) vertices, the hypercube of dimension n is its optimal hypercube. This was generalized in [56] for equibipartite binary caterpillars with legs of arbitrary length. A variation to Havel's conjecture is that a binary tree is a subgraph of its optimal augmeted cube.

3.2 A technique for embedding in augmeted cubes

We adopt a recursive approach to define an embedding of binary trees into augmented cubes. Let T be a binary tree on 2^n vertices. We partition T into atmost

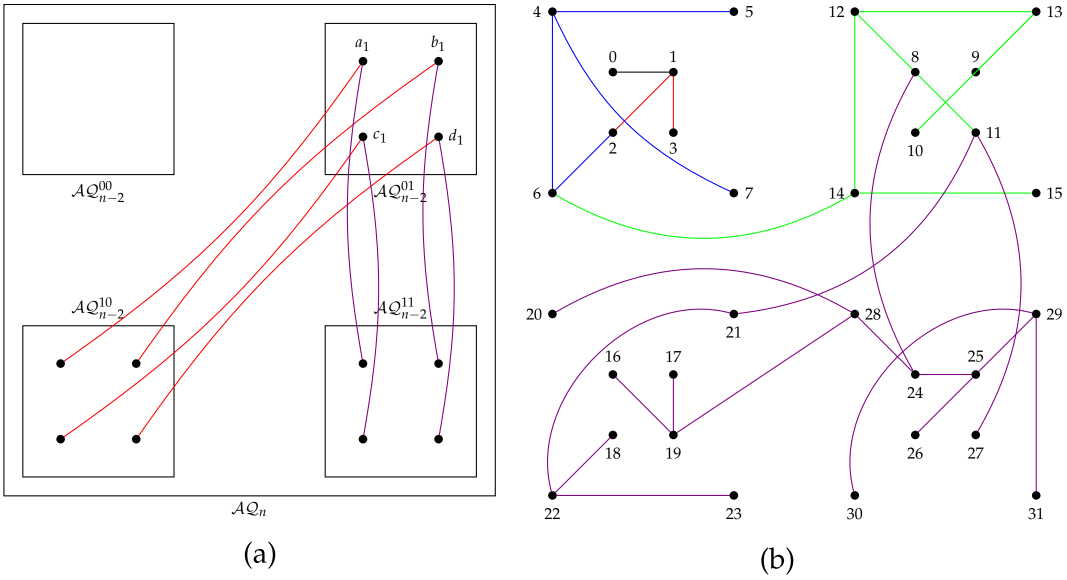

a_1
b_1
c_1
d_1
$\mathcal{AQ}^{00}_{n-2}$
$\mathcal{AQ}^{01}_{n-2}$
$\mathcal{AQ}^{10}_{n-2}$
$\mathcal{AQ}^{11}_{n-2}$
$\mathcal{AQ}_n$
(a)
4
5
12
13
0
1
8
9
2
3
10
11
6
7
14
15
20
21
28
29
16
17
25
24
18
19
26
27
22
23
30
31
(b)

Figure 3.2: (a) A partition of $\mathcal{AQ}_n$ into four $\mathcal{AQ}_{n-2}$. (b) An embedding of a binary tree into $\mathcal{AQ}_5$.

five subtrees $X, X_1, X_2, X_3, X_4, X_5$, by removing at most 4 edges, such that X has 2^{n-1} vertices. Assume that the deleted edges are (a_1, a_2), (b_1, b_2), (c_1, c_2) and (d_1, d_2) with $P = \{a_1, b_1, c_1, d_1\} \in$ X. The n-dimensional augmented cube $\mathcal{AQ}_n$ is partitioned into four $(n-2)$-dimensional augmented cubes, denoted by $\mathcal{AQ}^{00}_{n-2}$, $\mathcal{AQ}^{01}_{n-2}$, $\mathcal{AQ}^{10}_{n-2}$ and $\mathcal{AQ}^{11}_{n-2}$, such that the first two bits of vertices in $\mathcal{AQ}^{ab}_{n-2}$ is the constant ab, where $a, b \in \{0, 1\}$. Then, the embedding is so defined that X is mapped to $\mathcal{AQ}^{00}_{n-2}$ and $\mathcal{AQ}^{01}_{n-2}$ such that P lies in the second part $\mathcal{AQ}^{01}_{n-2}$. Since each element in $\mathcal{AQ}^{01}_{n-2}$ has two adjacent vertices in $\mathcal{AQ}^{1}_{n-1}$, one in $\mathcal{AQ}^{10}_{n-2}$ and the other in $\mathcal{AQ}^{11}_{n-2}$, and no two vertices in $\mathcal{AQ}^{01}_{n-2}$ has a common adjacent vertex in $\mathcal{AQ}^{1}_{n-1}$, the mapping conditions seem feasible (see Fig. 3.2a). As seen in Fig. 3.2b, we can apply this embedding technique on the pair $\{\mathcal{AQ}^{0}_{n-1}, X\}$, i.e., $\mathcal{AQ}^{0}_{n-1}$ is further divided into four sub-augmented cubes of dimension $(n-3)$ and X can be partitioned into at most five subtrees, and so on. In this embedding method, we need to show that given a binary tree on m vertices, there exists a partition of the tree into at most five subtrees, such that one subtree has $\frac{m}{2}$ vertices.

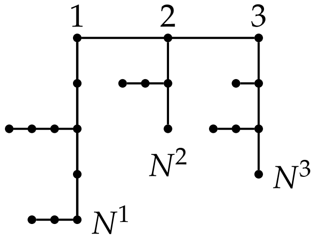


Figure 3.3: A 2-caterpillar with a backbone of order 3. It has three legs, each of which is a 1-caterpillar.

3.3 Embedding k-caterpillars into hypercubes

A 0-caterpillar is a path. For $k \geq 1$, a k-caterpillar is a binary tree consisting of a path with j-caterpillars $(0 \leq j \leq k-1)$ emanating from some of the vertices on the path. The path is called the backbone and its vertices the backbone vertices of the k-caterpillar. A leg of the k-caterpillar is a j-caterpillar, $0 \leq j \leq k-1$, originating from a backbone vertex, including the backbone vertex (see Fig. 3.3).

The order of a graph is its number of vertices, and its size is the number of edges. Let C be a k-caterpillar on m vertices and N be the order of its backbone. We denote the q^{th} leg of C by C^q and the order of the backbone of C^q by N^q (see Fig. 3.3). Similarly, the j^{th} leg in C^q will be denoted by $C^{q,j}$ and the order of the backbone of $C^{q,j}$ by $N^{q,j}$.

3.3.1 Properties of k-caterpillars and hypercubes

A perfectly balanced graph is a graph with a perfect matching, i.e., the vertex set can be partitioned into pairs such that each pair is an edge. A tree has at most one perfect matching. A path of odd length is perfectly balanced. Deleting a non-matching edge in a perfectly balanced tree partitions the tree into two perfectly balanced subtrees. A path connecting two distinct vertices x and y, denoted as $[x, y]$-path, is unique in a tree.

A k-caterpillar is also a j-caterpillar for all $j \geq k$. A strictly k-caterpillar is a k-caterpillar which is not a j-caterpillar, for any $j \leq k-1$. A backbone of a k-caterpillar is not unique. It can be extended to another backbone of higher order by including the backbone of the first leg or that of the last leg (see Fig. 3.4). Such

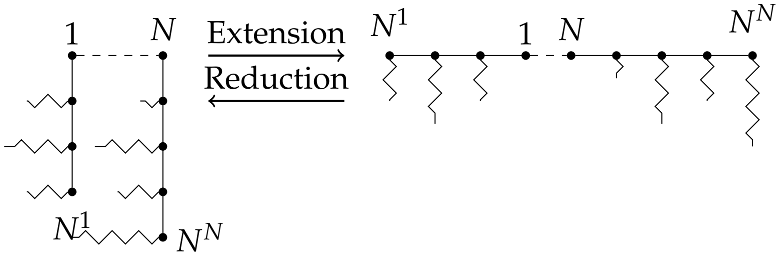


Figure 3.4: Extension and reduction of a backbone in a 2-caterpillar.

extension reduces the order of the first leg or that of the last leg. The following result is imminent.

Proposition 1. *If C is a k-caterpillar, with $k \geq 1$, on m vertices, then*

(a) a backbone of C may not be unique,

(b) there is a backbone with the first leg and the last leg, each of order strictly less than $\frac{m}{2}$ (for $m > 2$),

(c) if there is a backbone of order 2 then C is a $(k-1)$-caterpillar.

Proof. (a) In Fig. 3.3, the backbone of C can be extended by including the backbone of its first leg. Thus, the backbone is not unique.

(b) Let B_0 be a backbone of order n_0. Since C is a k-caterpillar, its first leg C^1 is a $(k-1)$-caterpillar. If C^1 has one vertex, then part (b) of the proposition is true. If C^1 has more than one vertex, then by adding the backbone of C^1 to B_0, we obtain a backbone B_1 containing B_0. If B_1 has n_1 vertices, then $n_1 > n_0$. Since C^1 is a $(k-1)$-caterpillar, the first leg of C with the backbone B_1 is a $(k-2)$-caterpillar. By extending the backbone in this way, we obtain a sequence of backbones $B_0 \subset B_1 \subset B_2 \subset \ldots$. Since, C is finite there exists a maximal backbone B_r, for some $r \geq 1$. Clearly, the order of the first leg in C with the backbone B_r is one. Hence, part (b) of the proposition holds true.

(c) Suppose B is a backbone of C of order 2, then by including the backbone of its only two legs, viz., C^1 and C^2, into B, we obtain a new backbone B' of C. Since legs of C^1 and C^2 are $(k-2)$-caterpillars, the legs of C with respect

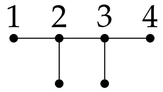


Figure 3.5: A strictly 1-caterpillar with unique backbone upto isomorphism.

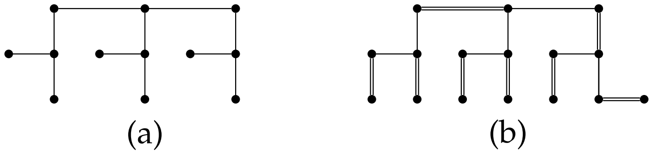


Figure 3.6: (a) A strictly 2-caterpillar and (b) a strictly perfectly balanced 2-caterpillar. A matching edge is drawn as double line segment.

to the new backbone B' are $(k-2)$-caterpillars. Therefore, C is a $(k-1)$-caterpillar.

□

Remark 1. *Consider a strictly 1-caterpillar C on 6 vertices with the degree sequence (2,2,1,1,1,1). C has a unique backbone of order 4, upto graph isomorphism.*

A backbone of a k-caterpillar can also be reduced to a backbone of smaller order if there is a backbone vertex of degree 2 and all legs before or after this vertex are at most $(k-2)$-caterpillars, in which case, the vertex becomes the first or the last backbone vertex (see Fig. 3.4). A strictly 2-caterpillar has order at least 12 (see Fig. 3.6 (a)), and if, in addition, it is perfectly balanced then the order is at least 16 (see Fig. 3.6 (b)).

Consider a k-caterpillar on m vertices with a backbone B of order N. Suppose x is the first backbone vertex and y is any backbone vertex, we define $f_B(y)$ to be the order of the first l legs of C, where l is the order of the $[x, y]$-path. If the backbone vertices are labeled from 1 to N, then we simply write $f_B(l)$. Clearly, f_B is a strictly increasing function. We use this function to prove the following results. Unless explicitly specified, we assume that the backbone vertices are labeled from 1 to N.

Proposition 2. *Let C be a k-caterpillar on $m > 2$ vertices with a backbone B of order N such that C^1 and C^N are both of order strictly less than $\frac{m}{2}$. Then, $\exists q$, with $1 < q < N$, such that $f_B(q) \geq \frac{m}{2}$ and $f_B(q-1) < \frac{m}{2}$.*

Proof. Since f_N is a strictly increasing function with $f_B(1) < \frac{m}{2}$ and $f_B(N) = m > \frac{m}{2}$, we get the required result. □

Proposition 3. *Let C be a perfectly balanced k-caterpillar on $m > 2$ vertices with a backbone B of order N. If there exists q, with $1 < q < N$, such that $f_B(q) > \frac{m}{2}$ and $f_B(q-1) < \frac{m}{2}$, then we can deduce that $(q-1, q)$ is not a matching edge.*

Proof. If $f_B(q) > \frac{m}{2}$ then $f_B(N) - f_B(q) < \frac{m}{2}$. So, if $(q-1, q)$ is not a matching edge, we are done, otherwise by reversing the labels of the backbone vertices, i.e. the backbone vertex i $(1 \leq i \leq N)$ is relabeled by $N - i + 1$, we get the required result. □

A maximal backbone B of a k-caterpillar C is a backbone which can not be extended to a larger backbone of C containing B. The first and the last legs in a maximal backbone are each of order one. If C is of order $m = 2$, then it has exactly two backbones, viz., (a) a backbone with one leg of order two and (b) a backbone with two legs, each of order one. In either case the first leg or the last leg can not have order strictly less than $\frac{m}{2}$, therefore, in the previous propositions we assumed $m > 2$.

Suppose q is a backbone vertex of a perfectly balanced k-caterpillar C. If the order of the leg C^q is even then it is perfectly balanced, else $C^q \backslash \{q\}$, i.e., q is removed from C^q, is perfectly balanced. Moreover, if C^q is of odd order then either $(q, q+1)$ or $(q-1, q)$ is a matching edge, in which case, C^{q+1} or C^{q-1} is of odd order.

Proposition 4. *Let C be a perfectly balanced k-caterpillar with N backbone vertices and M be its perfect matching. Suppose $(i, j) \in M$ is a backbone edge with $1 \leq i < j < N$, then, either the first leg of even order lies at an odd distance from j or $N - 1$ is at odd distance from j with $(N-1, N) \in M$.*

Proof. Let C^q be the first leg of even order from j, i.e., $q > j$ is the minimum integer for which C^q is of even order. Then C^q is perfectly balanced and alternate edges on $[i, q]$-path are matching edges, with $(q-1, q) \notin M$, so $q - j$ is odd (see Fig. 3.7). If no such path exists then $N - i$ is odd with $(N-1, N) \in M$. □

Proposition 5. *Let C be a perfectly balanced 1-caterpillar on m $(m \geq 2)$ vertices with a backbone B of order N. Let x be the first backbone vertex on B, then there exist backbones*

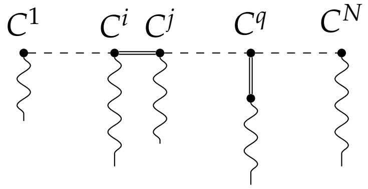


Figure 3.7: (i, j) is a matching edge drawn as double line segment. C^q is the first leg of odd length from the backbone vertex j.

B' and B'' of orders N' and N'', respectively, both having x as its first backbone vertex, such that

(a) the $[x, N']$-path is of even length and,

(b) the $[x, N'']$-path is of odd length.

Proof. The result can be obtained by extending and reducing the backbone B. Suppose the order of B is odd. If the order of the last leg C^N is greater than 1, then B can be extended to a backbone of even order by including the second vertex on C^N. If the order of C^N is one, then the order of C^{N-1} is odd. Exchange the edge $(N-1, N)$ by the path C^{N-1} to form a new backbone of even order. By the similar approach we get a backbone of odd order, if B were of even order. □

Remark 2. *We can apply Propositon 5 on a leg of a 2-caterpillar. However, if a leg is of order 2, then the leg has unique backbone, which is of odd length.*

A hypercube $\mathcal{Q}_n$ of dimension n is a graph with the vertex set $\mathbb{Z}_2^n$ and two vertices being adjacent if and only if the Hamming distance between them is exactly one. We use the following properties of the hypercube in this paper.

Lemma 2. *[27] A hypercube of dimension $n \geq 1$, is*

1. *K_1 symmetric, i.e., vertex symmetric,*

2. *K_2 symmetric, i.e., edge symmetric,*

3. *$K_{1,2}$ symmetric for $n \geq 2$, i.e., P_3 symmetric,*

4. *$K_{1,3}$ symmetric for $n \geq 3$, i.e., claw symmetric,*

5. *C_4 symmetric for $n \geq 2$.*

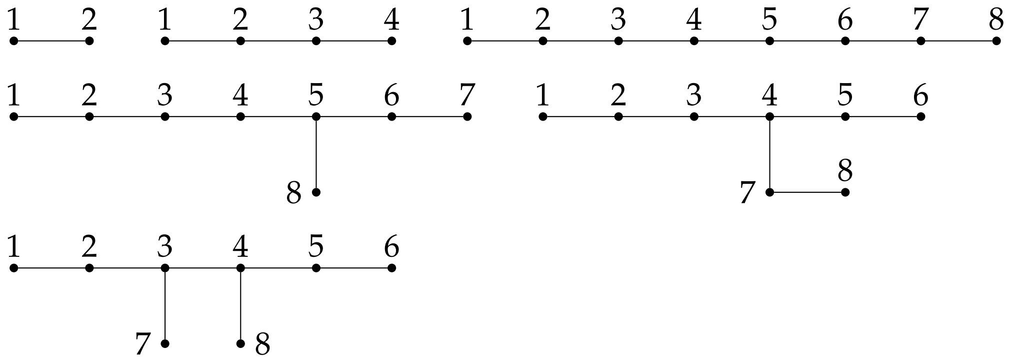


Figure 3.8: Perfectly balanced 2-caterpillars on 2, 4 and 8 vertices.

where K_n is a complete graph, P_n is a path and C_n is a cycle, all on n vertices, and $K_{p,q}$ is a complete bi-partite graph with parts of order p and q.

3.3.2 The embedding theorem

An embedding is an injective graph homomorphism. In this paper, we show that any perfectly balanced 2-caterpillar on m vertices, where $2^{n-1} < m \leq 2^n$ ($n \geq 1$), is embeddable into a hypercube of dimension n. It is sufficient to show that a perfectly balanced 2-caterpillar on 2^n vertices span the hypercube of dimension n.

Theorem 9. *Let C be a perfectly balanced 2-caterpillar on 2^n $(n \geq 1)$ vertices. Then C is a subgraph of the hypercube Q_n of dimension n.*

Proof. For $n \leq 3$, there are exactly 6 perfectly balanced 2-caterpillars as shown in Fig. 3.8, viz., two perfectly balanced 0-caterpillars on 2 and 4 vertices, and four perfectly balanced 1-caterpillars on 8 vertices. Each of them are embeddable into the respective optimal hypercubes. For $n = 4$, we have, using the brute force method, verified that the theorem holds. For $n \geq 5$, we prove a stronger result as given in the following theorem. □

The following theorem states that a perfectly balanced 2-caterpillar, on at least 32 vertices, is embeddable into its optimal hypercube, with at most four fixed vertices being mapped to some fixed graph patterns, where K_1, K_2, $K_{1,2}$, $K_{1,3}$ and C_4 are among the fixed graph patterns.

Theorem 10. *For $n \geq 5$, let C be a perfectly balanced 2-caterpillar on 2^n $(n \geq 1)$ vertices with a backbone B on N vertices. Let x be the first backbone vertex on C, y be the j^{th} $(2 \leq j \leq N^1)$ backbone vertex on C^1, z be the end vertex on the path $C^{1,j}$ and α be the $(j-1)^{th}$ vertex, if it exists, on C^1 (see Fig. 3.9 (a)). Then, there exists an embedding ϕ of C into $\mathcal{Q}_n$ such that $\phi(\{x, y, z\})$ induces some fixed graph patterns (see Fig. 3.9 (b)), viz.,*

1. *if $[x, y]$-path is of odd length and*

   (i) if $[y, z]$-path is of odd length then the sequence $[\phi(x), \phi(y), \phi(z)]$ is P_3.

   (ii) else, if $[y, z]$-path is of even length then the sequence $[\phi(y), \phi(x), \phi(z)]$ is P_3.

2. *else, $[x, y]$-path is of even length, in which case,*

   (i) if $[y, z]$-path is of even length then $\{\phi(x), \phi(y), \phi(\alpha), \phi(z)\}$ induces a claw $K_{1,3}$, with $\phi(\alpha)$ as its central vertex.

   (ii) else if $[y, z]$-path is of odd length then the sequence $[\phi(x), \phi(\alpha), \phi(y), \phi(z), \phi(x)]$ is C_4.

Proof. We prove by induction on n. For the base case of $n = 5$, the result can be verified by the brute force method[1]. We now proceed with the induction step.

Assume that any perfectly balanced 2-caterpillar on 2^n $(n \geq 5)$ vertices is embeddable into $\mathcal{Q}_n$ with an embedding satisfying the conditions 1 or 2, as given in the theorem. Let C be a perfectly balanced 2-caterpillar on 2^{n+1} vertices. Let M be the perfect matching of C. Without loss of generality, assume that the order of the backbone of C is $N \geq 3$ and the order of C^1 and C^N are both strictly less than 2^n (by Proposition 1). The proof that C has an embedding ϕ into $\mathcal{Q}_{n+1}$, as required, is exhibited by performing the following two steps.

I. *Partition of the 2-caterpillar:* C is partitioned into at most four subtrees, say X, X_2, Y_2 and Z_2, such that if $\{(x_1, x_2), (y_1, y_2), (z_1, z_2)\}$ are deleted edges then, as seen in Fig. 3.10,

[1] GitHub link: https://github.com/rishikantrajdeepak/EmbeddingBinaryTreeIntoHypercube

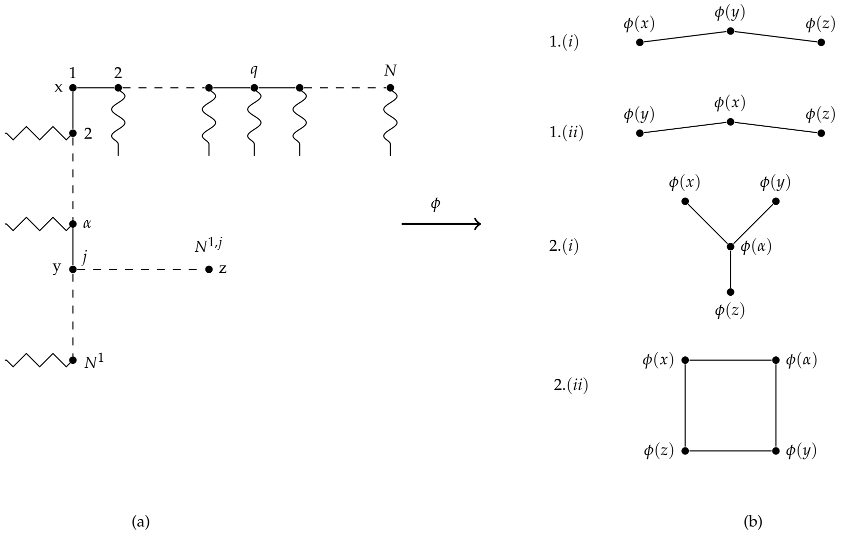


Figure 3.9: (a) A perfectly balanced 2-caterpillar with three fixed vertices x, y and z on its first leg and (b) An embedding ϕ mapping x, y and z into some fixed patterns in $\mathcal{Q}_n$.

(a) (x_1, x_2) lie on the backbone of C, (y_1, y_2) lie on the backbone of the leg C^{x_1} and (z_1, z_2) lie on the leg C^{x_1,y_1}, and

(b) x_1, y_1 and z_1 are contained in X and x_2, y_2 and z_2 are contained in X_2, Y_2 and Z_2, respectively,

such that X is perfectly balanced 2-caterpillar of order 2. The remaining subtrees, i.e. X_2, Y_2 and Z_2, are joined by some new edges to form a perfectly balanced 2-caterpillar, say Y, of order 2^n, such that $\{x_2, y_2, z_2\}$ lie on one of the fixed patterns.

II. *Extension of embeddings:* By the induction hypothesis, there exists an embedding $\phi_1 : X \to \mathcal{Q}_n$ such that $\phi_1(\{x_1, y_1, z_1\})$ lie on one of the fixed patterns satisfying one of the four conditions, as mentioned in the theorem. By construction, $\{x_2, y_2, z_2\}$ lie on one of the fixed pattern, so any embedding $\phi_2 : Y \to \mathcal{Q}_n$, which exists by the induction hypothesis, will preserve the pattern. By Lemma 2, there exists automorphisms π_1 and π_2 on $\mathcal{Q}_n$ such that $\pi_1 \circ \phi_1(x_1) = \pi_2 \circ \phi_2(x_2)$, $\pi_1 \circ \phi_1(y_1) = \pi_2 \circ \phi_2(y_2)$ and

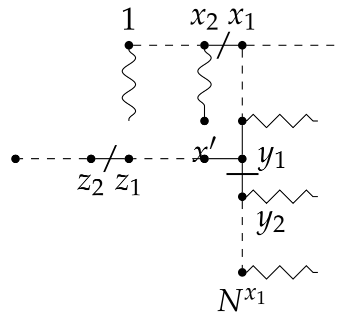


Figure 3.10: Partition of a perfectly balanced 2-caterpillar.

$\pi_1 \circ \phi_1(z_1) = \pi_2 \circ \phi_2(z_2)$. Define an embedding $\phi : C \to \mathcal{Q}_{n+1}$ by

$$\phi(x) = \begin{cases} 0\pi_1 \circ \phi_1(x); & \text{if } x \in X, \\ 1\pi_2 \circ \phi_2(x); & \text{if } x \in Y. \end{cases} \tag{3.1}$$

It then follows that ϕ is an embedding and $(\phi(x_1), \phi(x_2))$, $(\phi(y_1), \phi(y_2))$ and $(\phi(z_1), \phi(z_2))$ form edges in $\mathcal{Q}_{n+1}$. Thus, once a partition, $\{X, Y\}$ of C, is obtained, embeddings ϕ_1 and ϕ_2 exist by the induction hypothesis. So, we only need to show that C can be partitioned as required in step I.

We adopt some notations to be used in the proof. The order of a graph G is denoted by $o(G)$. A subgraph induced by X, where X is a subgraph of G, is denoted by $\langle X \rangle$. Recall that $f_B(y)$ is the order of the first l legs of a k-caterpillar with a backbone B of N vertices, where y is the l-th backbone vertex. We discuss the proof in cases, as follows.

By Proposition 2, there exists an integer q, with $1 < q < N$, such that $f_B(q) \geq 2^n$ and $f_B(q-1) < 2^n$. As seen in Fig. 3.10, the value of $f_B(q)$ determines the following three cases.

1. If $f_B(q) = 2^n$, then $(q, q+1) \notin M$. Put $x_1 = q+1$ and $x_2 = q$. Delete (x_1, x_2) to get $X = \langle C^{x_1}, \ldots, C^N \rangle$ and $Y = \langle C^1, \ldots, C^{x_2} \rangle$. By the induction hypothesis, X and Y can be embedded into $\mathcal{Q}_n$ via maps ϕ_1 and ϕ_2, respectively. By vertex-symmetry of $\mathcal{Q}_n$, we get $\phi_1(x_1) = \phi_2(x_2)$. Hence, the extended embedding ϕ, as defined in Eq. 3.1, maps $\{x_1, x_2\}$ into an edge $(\phi(x_1), \phi(x_2))$ of $\mathcal{Q}_{n+1}$.

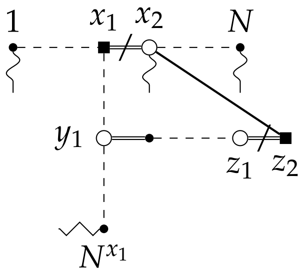


Figure 3.11: Case 2. $f_B(q) = 2^n + 1$ and two matching edges are deleted.

2. If $f_B(q) = 2^n + 1$, then $(q, q+1) \in M$. Put $x_1 = q$ and $x_2 = q + 1$. Let C^{x_1,y_1} be the first path of odd length from x_1, on the leg C^{x_1}, and (z_1, z_2) be the last edge on this path (see Fig. 3.11). Then, $[x_1, z_1]$-path is of odd length. Delete (x_1, x_2) and (z_1, z_2) to obtain $X = \langle C^1, \ldots, C^{q-1}, C^{x_1} \backslash z_2 \rangle$. All the matching edges edges along the $[x_1, z_1]$-path in C become non-matching edges in X and vice-versa. Add (x_2, z_2), which becomes a new matching edge, to obtain $Y = \langle (x_2, z_2), C^{x_2}, \ldots, C^N \rangle$. By the induction hypothesis 1.(ii), we get $(\phi_1(x_1), \phi_1(z_1))$ as an edge in $\mathcal{Q}_n$. By construction (x_2, z_2) is an edge in Y, so $(\phi_2(x_2), \phi_2(z_2))$ is an edge in $\mathcal{Q}_n$. By edge-symmetry of $\mathcal{Q}_n$, we get $\phi_1(x_1) = \phi_2(x_2)$ and $\phi_1(z_1) = \phi_2(z_2)$. Thus, $(\phi(x_1), \phi(x_2))$ and $(\phi(z_1), \phi(z_2))$ form edges in $\mathcal{Q}_{n+1}$, via the map ϕ.

3. If $f_B(q) > 2^n + 1$ then put $x_1 = q$ and $x_2 = q - 1$. Without loss of generality, assume (x_1, x_2) is a non-matching edge, by Proposition 1. Delete (x_1, x_2) to get a part $X_2 = \langle C^1, \ldots, C^{x_2} \rangle$. This case is further divided into two subcases 3.1 and 3.2.

3.1. If $\exists (y_1, y_2)$, with $y_1 \neq x_1$, on the backbone B^{x_1} of C^{x_1}, such that

$$f_B(x_2) + f_{B^{x_1}}(N^{x_1}) - f_{B^{x_1}}(y_1) = 2^n,$$

then delete (y_1, y_2) to get $Y_2 = \langle C^{x_1,y_2}, \ldots, C^{x_1,N^{x_1}} \rangle$. The remaining part is $X = \langle C \backslash X_2 \cup Y_2 \rangle$. To join the parts X_2 and Y_2, we add a new edge, as described in the following sub-cases 3.1.1 and 3.1.2.

3.1.1. If $[x_1, y_1]$-path is of odd length, then add the edge (x_2, y_2) to get second part

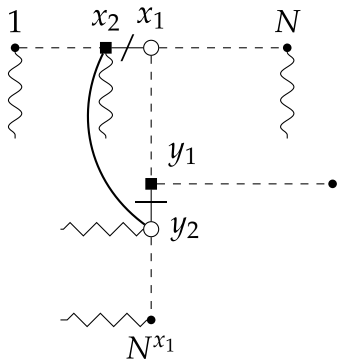


Figure 3.12: Case 3.1.1. Two non-matching edges are deleted and one new edge is added to construct two perfectly balanced 2-caterpillars, each of order 2^n.

$Y = \langle X_2, (x_2, y_2), Y_2\rangle$, as shown in Fig. 3.12. By the induction hypothesis 1, $\{x_1, y_1\}$ and $\{x_2, y_2\}$ are mapped to an edge in $\mathcal{Q}_n$ via maps ϕ_1 and ϕ_2, respectively. By edge-symmetry of $\mathcal{Q}_n$, we get $\phi_1(x_1) = \phi_2(x_2)$ and $\phi_1(y_1) = \phi_2(y_2)$. Thus, $(\phi(x_1), \phi(x_2))$ and $(\phi(y_1), \phi(y_2))$ are edges in $\mathcal{Q}_{n+1}$.

3.1.2. If $[x_1, y_1]$-path is of even length, then by Proposition 5 $[y_2, N^{x_1}]$-path is of odd length. By the induction hypothesis 2, $\{x_1, y_1\}$ is mapped to end vertices of a path P_3 in $\mathcal{Q}_n$. As seen in Fig. 3.13, we further have two subcases.

(i) If $o(C^{x_2}) = 1$ then add edge (x_2, N^{x_1}) to get $Y = \langle X_2, (x_2, N^{x_1}), Y_2\rangle$.

(ii) If $o(C^{x_2}) > 1$ then $[x_2, N^{x_2}]$-path is of odd length (by Proposition 5). Add (N^{x_2}, y_2) to get $Y = \langle X_2, (N^{x_2}, y_2), Y_2\rangle$.

In both the subcases, by the induction hypothesis 2, $\{x_2, y_2\}$ is mapped to end vertices of a path P_3 in $\mathcal{Q}_n$, via map ϕ_2. By P_3-symmetry of $\mathcal{Q}_n$, we see that $\phi_1(x_1) = \phi_2(x_2)$ and $\phi_1(y_1) = \phi_2(y_2)$. Thus, the extended map ϕ form edges $(\phi(x_1), \phi(x_2))$ and $(\phi(y_1), \phi(y_2))$ in $\mathcal{Q}_{n+1}$.

3.2. If $\exists$ (y_1, y_2) on the backbone of C^{x_1} and (z_1, z_2) on C^{x_1, y_1} such that

$$f_B(x_2) + (f_{N^{x_1}}(N^{x_1}) - f_{N^{x_1}}(y_1)) + (f_{N^{x_1,y_1}}(N^{x_1,y_1}) - f_{N^{x_1,y_1}}(z_1)) = 2^n,$$

then these two edges are unique. Delete (y_1, y_2) and (z_1, z_2) to get the parts $Y_2 = \langle C^{x_1,y_2}, \ldots, C^{x_1,N^{x_1}}\rangle$ and $Z_2 = [z_2, N^{x_1,y_1}]$-path. The first part obtained

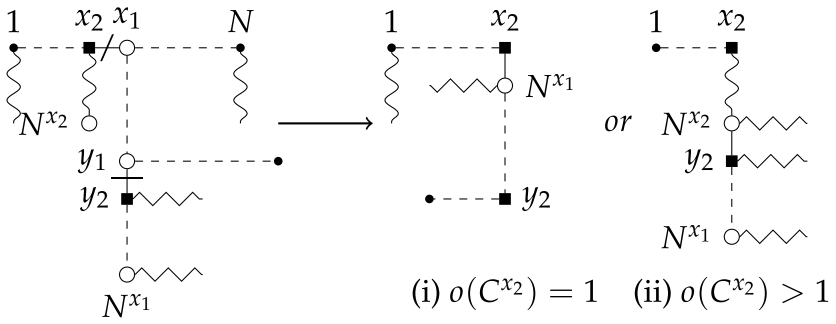


Figure 3.13: Case 3.1.2.Two non-matching edges are deleted and one new edge is added to construct two perfectly balanced 2-caterpillars, each of order 2^n.

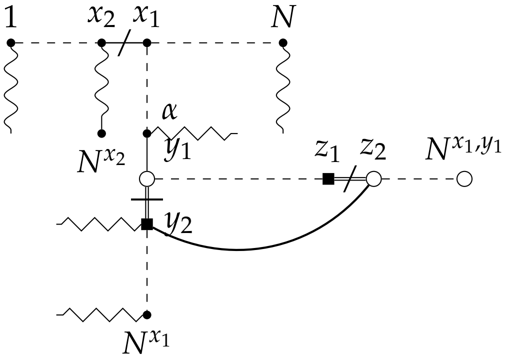


Figure 3.14: Case 3.2.1. Two matching edges (y_1, y_2) and (z_1, z_2) are deleted and compensated by adding one matching edge (y_2, z_2).

is $X = \langle C \backslash X_2 \cup Y_2 \cup Z_2 \rangle$. Since the degree of y_1 is 3, so there are three possible matching edges it can be incident to, as discussed below.

3.2.1. If $(y_1, y_2) \in M$ then $(z_1, z_2) \in M$ and $[y_1, z_1]$-path is of odd length. Add the new matching edge (y_2, z_2) (see Fig. 3.14). The non-matching edges along the $[y_1, z_1]$-path become matching edges and vice-versa. Furthermore,

(i) if $[x_1, y_1]$-path is of even length, add (x_2, z_2) to get the required 2- caterpillar $Y = \langle X_2, (x_2, z_2), Z_2, (y_2, z_2), Y_2 \rangle$ (Fig. 3.15). Since the sequence $[x_2, z_2, y_2]$ is P_3, so its image $[\phi_2(x_2), \phi_2(z_2), \phi_2(y_2)]$ is P_3 in $\mathcal{Q}_n$. By induction hypothesis $2(ii)$, the sequence $[\phi_1(x_1), \phi_1(z_1), \phi_1(y_1)]$ form P_3 in $\mathcal{Q}_n$. By P_3-symmetry of $\mathcal{Q}_n$, we get $\phi_1(x_1) = \phi_2(x_2)$, $\phi_1(y_1) = \phi_2(y_2)$ and $\phi_1(z_1) = \phi_2(z_2)$. Thus, the extended map *phi* form edges $(\phi(x_1), \phi(x_2))$, $(\phi(y_1), \phi(y_2))$ and $(\phi(z_1), \phi(z_2))$ in $\mathcal{Q}_{n+1}$, as required.

(ii) if $[x_1, y_1]$-path is of odd length, then, as seen in Fig. 3.16,

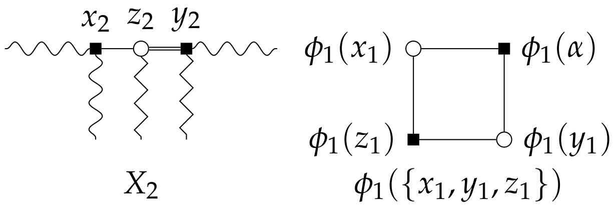


Figure 3.15: Case 3.2.1. Adding edges (x_2, y_2) and (y_2, z_2) to form perfectly balanced 2-caterpillar on 2^n vertices. Thus, (i) $\phi_2(\{x_2, z_2, y_2\})$ and $\phi_1(\{x_1, z_1, y_1\})$ are both $K_{1,2}$.

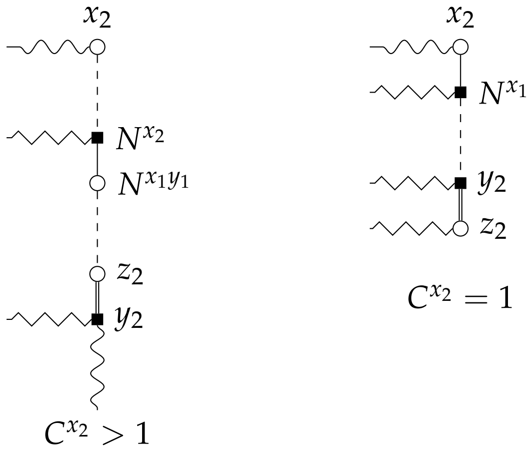


Figure 3.16: Case 3.2.1. (ii) Adding an edge (y_2, z_2) to form perfectly balanced 2-caterpillar on 2^n vertices .

- if $o(C^{x_2}) > 1$ then $[x_2, N^{x_2}]$-path is of odd length (by Proposition 5). Add (N^{x_2}, N^{x_1, y_1}) to get $Y = \langle X_2, (N^{x_2}, N^{x_1, y_1}), Z_2, (y_2, z_2), Y_2 \rangle$. Here, it is possible that $z_2 = N^{x_1 y_1}$.
- if $o(C^{x_2}) = 1$ then $[y_1, N^{x_1}]$-path is of odd length (by Proposition 5). Add (x_2, N^{x_1}) to get $Y = \langle X_2, (x_2, N^{x_1}), Y_2, (y_2, z_2), Z_2 \rangle$. Here, $y_2 = N^{x_1}$ is possible.

In both the cases, by the induction hypothesis 2 and since (y_2, z_2) is an edge, the sequence $[\phi_2(x_2), \phi_2(y_2), \phi_2(z_2)]$ form P_3 in $\mathcal{Q}_n$. By the induction hypothesis $1(i)$, the sequence $[\phi_1(x_1), \phi_1(y_1), \phi_1(z_1)]$ form P_3 in $\mathcal{Q}_n$. Using P_3-symmetry of $\mathcal{Q}_n$, we get $\phi_1(x_1) = \phi_2(x_2)$, $\phi_1(y_1) = \phi_2(y_2)$ and $\phi_1(z_1) = \phi_2(z_2)$. Thus, the extended map *phi* form edges $(\phi(x_1), \phi(x_2))$, $(\phi(y_1), \phi(y_2))$ and $(\phi(z_1), \phi(z_2))$ in $\mathcal{Q}_{n+1}$, as required.

3.2.2. If (y_1, y') is a matching edge on C^{x_1, y_1} then $[y_1, z_1]$-path is of odd length. Also

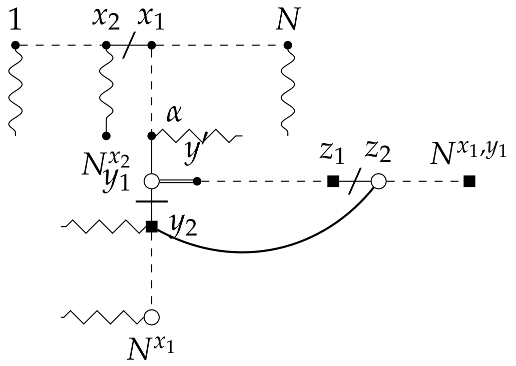


Figure 3.17: Case 3.2.2. Deleting three non-matching edges.

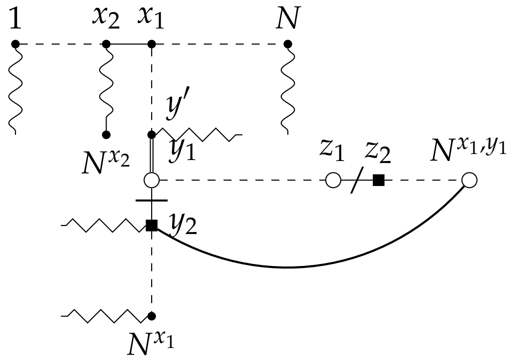


Figure 3.18: Case 3.2.3. Deleting three non-matching edges.

$[y_2, N^{x_1}]$-path is of odd length (by Proposition 5). Add (y_2, z_2). As seen in Fig. 3.17,

(i) If $[x_1, y_1]$-path is of even length then add (x_2, z_2) to get the second part $Y = \langle X_2, (x_2, z_2), Z_2, (y_2, z_2), Y_2 \rangle$.

(ii) If $[x_1, y_1]$-path is of odd length then,

- if $s(C^{x_2}) > 1$ then $[x_2, N^{x_2}]$-path is of odd length (by Proposition 5). Add (N^{x_2}, N^{x_1}) to get $Y = \langle X_2, (N^{x_2}, N^{x_1}), Y_2, (y_2, z_2), Z_2 \rangle$.
- if $s(C^{x_2}) = 1$ and since $[z_2, N^{x_1,y_1}]$-path is of odd length, (x_2, N^{x_1,y_1}) is added to get $Y = \langle X_2, (x_2, N^{x_1,y_1}), Y_2, (y_2, z_2), Z_2 \rangle$.

3.2.3. If $(y', y_1) \in M$ on the backbone of C^{x_1}, with $y' \neq y_2$, then $[y_1, z_1]$-path and $[z_2, N^{x_1,y_1}]$-path are both of even length. Add (y_2, N^{x_1,y_1}) (see Fig 3.18). We further discuss two sub-cases.

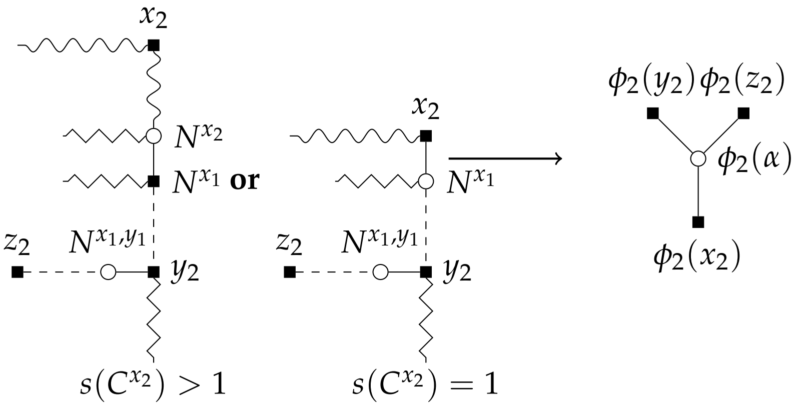


Figure 3.19: Case 3.2.3. (i). Joining subtrees containing x_2, y_2 and z_2.

(i) If $[x_1, y_1]$-path is of even length then, as seen in the Fig. 3.19,

- if $o(C^{x_2}) > 1$ then $[x_2, N^{x_2}]$-path is of odd length (by Proposition 5). By Proposition 5, $[y_2, N^{x_1}]$-path is of even length. If $o(Y_2) > 2$ then add (N^{x_2}, N^{x_1}) to get $Y = \langle X_2, (N^{x_2}, N^{x_1}), Y_2, (y_2, N^{x_1,y_1}), Z_2\rangle$, otherwise add (N^{x_2}, y_2) to get $Y = \langle X_2, (N^{x_2}, y_2), Y_2, (y_2, N^{x_1,y_1}), Z_2\rangle$.
- if $o(C^{x_2}) = 1$ then add (x_2, N^{x_1}), since $[y_2, N^{x_1}]$-path is of odd length by Proposition 5, to get $Y = \langle X_2, (x_2, N^{x_1}), Y_2, (y_2, N^{x_1,y_1}), Z_2\rangle$.

(ii) If $[x_1, y_1]$-path is of odd length then,

- If $o(C^{x_2}) > 1$ then $[x_2, N^{x_2}]$-path is of odd length (by Proposition 5). Add (N^{x_2}, N^{x_1}) to get $Y = \langle X_2, (N^{x_2}, N^{x_1}), Y_2, (y_2, N^{x_1,y_1}), Z_2\rangle$.
- If $o(C^{x_2}) = 1$ then add (x_2, z_2) to get $Y = \langle X_2, (x_2, z_2), Z_2, (y_2, N^{x_1,y_1}), Y_2\rangle$.

□

Remarks

As a next improvement of the result presented in this chapter, we can consider embedding perfectly balanced k-caterpillars ($k \geq 3$). However, the proof technique of creating sub-caterpillars may not be appropriate for these caterpillars. This is because, we will have to then delete more than 3 edges from the k-caterpillars to obtain a perfectly balanced sub-caterpillar. But then, there do not exist any more path symmetries in $\mathcal{Q}_n$ to embed the sub-caterpillar as desired.

Part II

Quantum walks on Cubelike Structures

CHAPTER 4

Discrete-time coined quantum walks on regular graphs

Discrete-time quantum walks are quantum counterpart of random walks on graphs. The first quantum random walk models were proposed in [5]. It has since been observed that there are some startling differences between classical and quantum walks. For instance, it has been shown that the hitting time from one vertex to the antipodal vertex is linear in the dimension of the hypercube, which is exponentially fast in quantum walks compared to classical walks [49]. Moreover, there is also a quadratic speed-up obtained in the mixing time of quantum versus the classical walks [57]. These differences between classical and quantum walk led to a search for quantum walk-based algorithms that can outperform classical counterparts. Some of the quantum algorithms that have been developed based on quantum walks are searching a marked element on a grid, the element distinctness problem, and the triangle finding problem in a graph [6].

In this chapter, we investigate quantum hitting times on cubelike graphs in two different ways; one by giving a closed form for the system's evolution after T steps, and the other by constructing quantum circuits for the evolution operator and implementing on IBM's computing platform.

4.1 Quantum walk evolution

A discrete-time coined quantum walk (DTQW) on a graph Γ on N vertices is described by the evolution of an associated quantum system in a Hilbert space. The position state of the system is a unit vector in the Hilbert space. The Hilbert space is spanned by N orthonormal vectors where each basis vector is represented by a node in the graph. The evolution is assisted by an auxiliary Hilbert space of dimension Δ called the coin space, where Δ is the maximum degree of the graph. The coin space is spanned by edges, where each edge corresponds to a normalized directional vector along which the quantum system evolves. A coin state is a normalized linear combination of all directions that determines the pathway of the quantum system. The total Hilbert space of the associated graph is the tensor product of the coin space and the position space, and the state of the system is a unit vector in the joint Hilbert space. This can be paraphrased classically as; if the walker is at node v of degree d_v, then we toss a d_v-dimensional coin whose output is an adjacent edge (direction) along which the walker moves. Equivalently, in the quantum case, we say that there is an operator, called the coin operator, that operates on the coin space and transforms a coin state $|\psi\rangle$ to another coin state $|\psi'\rangle$. Then after another operator, called the shift operator, shifts the quantum walker from one position state to another position state along the direction $|\psi'\rangle$.

4.1.1 Regular graphs

Suppose Γ is Δ-regular with N vertices, then the associated Hilbert space is the joint Hilbert space $\mathcal{H} = \mathcal{H}_{\mathcal{C}} \otimes \mathcal{H}_{\mathcal{P}}$, where $\mathcal{H}_{\mathcal{C}} \cong \mathbb{C}^{\Delta}$ is known as the coin space, and $\mathcal{H}_{\mathcal{P}} \cong \mathbb{C}^{N}$ is known as the position space. Suppose vertices and edges are labeled by $\{v_0, v_1, \ldots, v_{N-1}\}$ and $\{\alpha_0, \alpha_1, \ldots, \alpha_{\Delta-1}\}$, respectively, then the computational basis of $\mathcal{H}$ is given by

$$\{|\alpha_k\rangle\, |v_a\rangle : 0 \leq k \leq \Delta - 1,\ 0 \leq a \leq N-1\}. \tag{4.1}$$

This association between Γ and $\mathcal{H}$ is such that vertex v_a represents the position state $|v_a\rangle$ in $\mathcal{H}_{\mathcal{P}}$, and edge α_k represents the coin state $|\alpha_k\rangle$ in $\mathcal{H}_{\mathcal{C}}$. The evolution of the system is described by a unitary operator $\mathcal{U}$, called the evolution operator, defined by

$$\mathcal{U} = \mathcal{S}(\mathcal{C} \otimes I), \tag{4.2}$$

where, $\mathcal{C}$ is the coin operator analogous to a Δ-dimensional classical coin and $\mathcal{S}$ is the shift operator that shifts the quantum state $|\alpha_k\rangle\,|v_a\rangle$ to a neighboring quantum state $|\alpha_k\rangle\,|v_b\rangle$ such that α_k is the label of the edge (v_a, v_b).

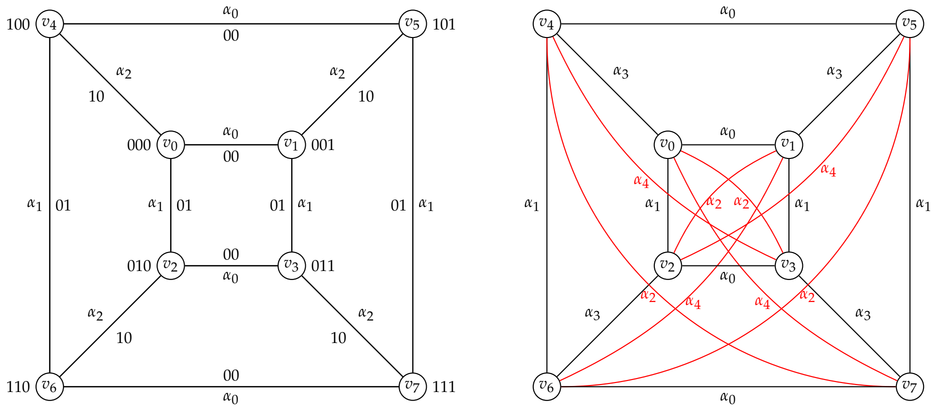
4.1.2 Cubelike graphs

The Cayley graph $Cay(\mathbb{Z}_2^n, \Omega)$ defined over the Boolean group $\mathbb{Z}_2^n$ is called cubelike graph of dimension n. For a cubelike graph $Cay(\mathbb{Z}_2^n, \Omega)$ when the ordering on Ω is not specified then we assume the lexicographical ordering, i.e., we assume that the cubelike graph is given in its canonical form. A vertex $(x_{n-1}, \ldots, x_1, x_0) \in \mathbb{Z}_2^n$ is denoted by the binary string $x_{n-1} \cdots x_1 x_0$, and is given the label v_a, where $a = x_{n-1}2^{n-1} + \cdots + x_1 2^1 + x_0 2^0$. An edge (v_a, v_b) is labeled by α_{k-1}, $1 \leq k \leq \Delta$, if the XOR operation of v_a and v_b, i.e. $v_a \oplus v_b$, is the k-th element $\Omega(k)$ of Ω. Suppose $2^{m-1} < \Delta \leq 2^m$, for some integer m, then the edge labeled by α_{k-1} can be denoted by the k-th binary string $y_{m-1} \cdots y_1 y_0$ in $\mathbb{Z}_2^m$, i.e., $k - 1 = y_{m-1}2^{m-1} + \cdots + y_1 2^1 + y_0 2^0$.

Some special subfamily of cubelike graphs of interest are (a) Hypercubes $\mathcal{Q}_n$ with $\Omega = \{0^i 1 0^j : i + j = n - 1\}$, and (b) Augmented cubes [28] $\mathcal{AQ}_n$ with $\Omega = \{0^i 1 0^j : i + j = n - 1\} \cup \{0^{n-i} 1^i : 1 \leq i \leq n\}$. In Fig. 4.1a, edge (v_0, v_4) is labeled by α_2 because $v_0 \oplus v_4 = 100$ is the third element of its generating set $\{001, 010, 100\}$, while in Fig. 4.1b it is labeled by α_3 as 100 is the fourth element of its generating set $\{001, 010, 011, 100, 111\}$. The binary representations of α_0, α_1, α_2 in $\mathcal{Q}_3$ (Fig. 4.1a) are 00, 01, 10, respectively, while in $\mathcal{AQ}_3$ (Fig. 4.1b), the binary representations of α_0, α_1, α_2, α_3, α_4 are 000, 001, 010, 011, 100, respectively.

The evolution operator

Any unitary operator on $\mathcal{H}_{\mathcal{C}} = \mathbb{C}^{\Delta}$ can be chosen as the coin operator. A popular coin operator is the Grover coin which is invariant under permutations of all



(a) $\mathcal{Q}_3$ (b) $\mathcal{AQ}_3$

Figure 4.1: A pictorial representation of (a) $\mathcal{Q}_3 = Cay(\mathbb{Z}_2^3, \{001, 010, 100\})$ and (b) $\mathcal{AQ}_3 = Cay(\mathbb{Z}_2^3, \{001, 010, 011, 100, 111\})$. In both the cases, a vertex $x_2x_1x_0$ is represented by v_a, where $a = x_2 2^2 + x_1 2^1 + x_0 2^0$, and an edge (v_a, v_b) is represented by α_{k-1}, if $v_a \oplus v_b$ is the k-element of the corresponding lexicographically ordered generating set. The edge α_{k-1} is also represented by the binary string (of length 2 in (a)) of the integer $k-1$.

directions. The Grover coin, denoted by $\mathcal{C}$, is defined by $2\left|D\right\rangle\left\langle D\right| - I$, where,

$$\left|D\right\rangle = \frac{1}{\sqrt{\Delta}}\sum_{k=1}^{\Delta}\left|\alpha_{k-1}\right\rangle \implies \mathcal{C} = \begin{bmatrix} \frac{2}{\Delta}-1 & \frac{2}{\Delta} & \frac{2}{\Delta} & \cdots & \frac{2}{\Delta} \\ \frac{2}{\Delta} & \frac{2}{\Delta}-1 & \frac{2}{\Delta} & \cdots & \frac{2}{\Delta} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{2}{\Delta} & \frac{2}{\Delta} & \frac{2}{\Delta} & \cdots & \frac{2}{\Delta}-1 \end{bmatrix}. \tag{4.3}$$

The shift operator $\mathcal{S}$ is defined by

$$\mathcal{S} = \sum_{k=1}^{\Delta}\sum_{a=0}^{2^n-1}\left|\alpha_{k-1}, v_a \oplus \Omega(k)\right\rangle\left\langle\alpha_{k-1}, v_a\right|, \tag{4.4}$$

The following example illustrates the DTQW on the 2-dimensional hypercube.

Example 2 (DTQW on $\mathcal{Q}_2 = Cay(\mathbb{Z}_2^2, \{01, 10\})$). *The Hilbert space associated with the graph (see Fig. 4.2) is $\mathbb{C}^2 \otimes \mathbb{C}^4$, and the computational basis is*

$$\{\left|0\right\rangle\left|00\right\rangle, \left|1\right\rangle\left|00\right\rangle, \left|0\right\rangle\left|01\right\rangle, \left|1\right\rangle\left|01\right\rangle, \left|0\right\rangle\left|10\right\rangle, \left|1\right\rangle\left|10\right\rangle, \left|0\right\rangle\left|11\right\rangle, \left|1\right\rangle\left|11\right\rangle\}.$$

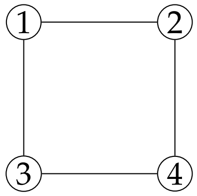


Figure 4.2: The 2-dimensional hypercube.

The evolution operator $\mathcal{U}$ is $\mathcal{S}(\mathcal{C} \otimes I)$, where,

$$\mathcal{C} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix},$$

and,

$$\mathcal{S} = \sum_{k=1}^{2} \sum_{a=0}^{3} |\alpha_{k-1}\rangle \, |v_a \oplus \Omega(k)\rangle \, \langle \alpha_{k-1}| \, \langle v_a| \, .$$

The evolution, with the initial state $|D\rangle \, |00\rangle = \frac{|0\rangle+|1\rangle}{\sqrt{2}} \, |00\rangle$, occurs as:

1. *The first time step;*

$$\begin{aligned} \mathcal{U} \, |D\rangle \, |00\rangle &= \mathcal{S}(\mathcal{C} \otimes I) \left(\frac{1}{\sqrt{2}} (|0\rangle \, |00\rangle + |1\rangle \, |00\rangle) \right) \\ &= \frac{1}{\sqrt{2}} \left(|1\rangle \, |10\rangle + |0\rangle \, |01\rangle \right). \end{aligned}$$

2. *The second time step;*

$$\begin{aligned} \mathcal{U}^2 \, |D\rangle \, |00\rangle &= \mathcal{S} \frac{1}{\sqrt{2}} \left[\left(2 \frac{|0\rangle + |1\rangle}{\sqrt{2}} \frac{1}{\sqrt{2}} - |1\rangle \right) |10\rangle \right. \\ &\quad \left. + \left(2 \frac{|0\rangle + |1\rangle}{\sqrt{2}} \frac{1}{\sqrt{2}} - |0\rangle \right) |01\rangle \right] \\ &= \mathcal{S} \frac{1}{\sqrt{2}} \left(|0\rangle \, |10\rangle + |1\rangle \, |01\rangle \right) = \frac{|0\rangle \, |11\rangle + |1\rangle \, |11\rangle}{\sqrt{2}} \\ &= |D\rangle \, |11\rangle \, . \end{aligned}$$

4.1.3 Hypercubes

We compare the classical random walk with the discrete-time coined quantum walk on the n-dimensional hypercube. The classical transition matrix M for the random walk on a r-regular graph is given by $M = \frac{1}{r}A$, where A is the adjacency matrix, viz.,

$$M_{i,j} = \begin{cases} \frac{1}{d} & ; \, i = j \\ 0 & ; \text{ otherwise.} \end{cases}.$$

Suppose Π_0 is the probability distribution of vertices at the initial time $t = 0$, then after time t the probability distribution of vertices is given by $\Pi_t = M^t \Pi_0$. The hitting time $H(i,j)$ is the expected number of steps taken by the walker to reach the vertex v_j from the vertex v_i. The following result computes the hitting time in a regular graph.

Theorem 11. *[55] If Γ is a r-regular graph with the spectral decomposition of the transition matrix M given by*

$$M = \sum_{k=1}^{N} \lambda_k \left|P_k\right\rangle \left\langle P_k\right|,$$

then the hitting time is given by

$$H(i,j) = N \sum_{k=1}^{N} \frac{1}{1-\lambda_k} (P_k(j)^2 - P_k(i)P_k(j)). \tag{4.5}$$

Theorem 12. *[42] The spectrum of the n-dimensional hypercube is $(n-2k), 0 \leq k \leq n$, with the multiplicity $\binom{n}{k}$.*

Using Theorem 11 and Theorem 12, the hitting time between a pair of antipodal vertices, say v_0 and v_{N-1}, is rewritten as

$$H(0, N-1) = n \sum_{k=1}^{n} \binom{n}{k} \frac{1}{2k} (1 - (-1)^k). \tag{4.6}$$

Thus, we get the following result.

Theorem 13. *The hitting time between antipodal vertices in the n-dimensional hypercube is asymptotically 2^n.*

The corresponding quantum hitting time of the DTQW on the hypercube is a measure of how quickly the quantum walker reach the target vertex, starting from an initial vertex. Among the various definitions of a quantum hitting time, we adopt the definition given by Kempe [49].

Definition 2 ([49])**.** *The discrete-time quantum walk is said to admit (T, p) one-shot $(|v_0\rangle, |v\rangle)$ hitting time if $| \langle v| \mathcal{U}^T |v_0\rangle |^2 \geq p$.*

In Example 2, $\mathcal{Q}_2$ has $(2, 1)$ one-shot $(|00\rangle, |11\rangle)$ hitting time. The following result stated by Kempe [49] shows that the quantum hitting time is linear in n which is exponentially faster than its classical counterpart.

Theorem 14 ([49])**.** *The discrete-time quantum walk with Grover coin on the hypercube of dimension n has a (T, p) one-shot $(|x\rangle, |\bar{x}\rangle)$ hitting time, i.e. $| \langle \bar{x}| \mathcal{U}^T |x\rangle |^2 \geq p$, where T is either $\lfloor \frac{\pi}{2} n \rfloor$ or $\lceil \frac{\pi}{2} n \rceil$ and $p = 1 - \mathcal{O}(\frac{\log^3 n}{n})$.*

Remarks

In Example 2, we get the probability distribution of the quantum states as; after the first step, the quantum system is at state $|01\rangle$ or $|10\rangle$ with probability $\frac{1}{2}$, and after two steps the probability of the quantum system to be at $|11\rangle$ is 1. The value 1 is remarkable and we want to find other cubelike graphs in which such probability is attained at least asymptotically. Kempe [49] generalizes the result to hypercubes where he states that the one-shot hitting time is asymptotically proportional to the dimension of the hypercube with the probability ≈ 1.

4.2 Hitting times

The DTQW starts from a particular starting vertex v_0 and evolves under the dynamics of the evolution operator given in Eq. (4.4) and (4.3). Letting the system evolve for a finite number of steps T is equivalent to applying $\mathcal{U}^T$ to an initial state $|v_0\rangle$ of the system. A measurement is then performed in the position space basis to see if the walker is in a specific target vertex. At time T, if we find the walker in the state $|v\rangle$ corresponding to the target vertex v with probability at least p, then we say (T, p) is the quantum hitting time for v, starting from vertex v_0. This hitting time, defined by Kempe in [49], is the one-shot hitting time.

In this section, we compute the closed form for the quantum state of the system after applying the evolution operator finitely many times. As a result we calculate the hitting times on the complete graphs on 2^n vertices.

4.2.1 An analysis of the evolution operator

We want to compute the quantum state of the system after applying the evolution operator finitely many times at a given initial state. This will allow us to get the coefficients of all computational basis vectors and hence the amplitude square can be calculated to get the probability of the quantum system to be at a particular vertex. If the probability p_v attained at any vertex v for any time step T is higher than the fixed value p, then we say the vertex v is a target vertex with a target probability p_v and a hitting time T.

Notice that the action of the Grover coin operator $\mathcal{C}$ on a coin state $|\alpha_k\rangle$ and the diagonal state $|D\rangle$ is described as

$$\begin{aligned} \mathcal{C}\,|\alpha_k\rangle &= \frac{2}{\sqrt{\Delta}}\,|D\rangle - |\alpha_k\rangle\,, \\ \mathcal{C}\,|D\rangle &= |D\rangle\,, \end{aligned} \tag{4.7}$$

and the action of $\mathcal{S}$ on a quantum state of the system is given by

$$\begin{aligned} \mathcal{S}\,|\alpha_k\rangle\,|v_a\rangle &= |\alpha_k\rangle\,|v_a \oplus \Omega(k)\rangle\,, \\ \mathcal{S}\,|D\rangle\,|v_a\rangle &= \frac{1}{\sqrt{\Delta}}\sum_{k=1}^{\Delta}|\alpha_k\rangle\,|v_a \oplus \Omega(k)\rangle\,. \end{aligned} \tag{4.8}$$

Assume that the DTQW begins at the origin $|v_0\rangle \equiv |0\rangle^{\otimes n}$. We use the following notations for further discussions; let $\mathcal{X}_k$ and $\mathcal{Y}_k$ be defined as

$$\begin{aligned} \mathcal{X}_k &\equiv \sum_{i_1=1}^{\Delta}\cdots\sum_{i_k}^{\Delta}|\alpha_{i_k}\rangle\,|v_0 \oplus \Omega(i_1) \oplus \cdots \oplus \Omega(i_k)\rangle \\ \mathcal{Y}_k &\equiv \sum_{i_1=1}^{\Delta}\cdots\sum_{i_k}^{\Delta}|D\rangle\,|v_0 \oplus \Omega(i_1) \oplus \cdots \oplus \Omega(i_k)\rangle \end{aligned} \tag{4.9}$$

With these notations, the action of the evolution operator $\mathcal{U}$ on $\mathcal{X}_k$ and $\mathcal{Y}_k$ is written as;

$$\begin{aligned} \mathcal{U}(\mathcal{X}_k) &= \frac{2}{\Delta}\mathcal{X}_{k+1} - \sqrt{\Delta}\mathcal{Y}_{k-1}, \\ \mathcal{U}(\mathcal{Y}_k) &= \frac{1}{\sqrt{\Delta}}\mathcal{X}_{k+1}. \end{aligned} \tag{4.10}$$

To understand the above expressions see the first few iterations of the evolution;

$$\begin{aligned} \mathcal{U}\mathcal{Y}_0 &= \mathcal{U}\,|D\rangle\,|v_0\rangle = \frac{1}{\sqrt{\Delta}}\mathcal{X}_1 \\ \mathcal{U}^2\mathcal{Y}_0 &= \frac{2}{\Delta\sqrt{\Delta}}\mathcal{X}_2 - \mathcal{Y}_0 \\ \mathcal{U}^3\mathcal{Y}_0 &= \frac{2^2}{\Delta^2\sqrt{\Delta}}\mathcal{X}_3 - \frac{2}{\Delta}\mathcal{Y}_1 - \frac{1}{\sqrt{\Delta}}\mathcal{X}_1. \\ \mathcal{U}^4\mathcal{Y}_0 &= \frac{2^3}{\Delta^3\sqrt{\Delta}}\mathcal{X}_4 - \frac{2^2}{\Delta^2}\mathcal{Y}_2 - \frac{2\cdot 2}{\Delta\sqrt{\Delta}}\mathcal{X}_2 + \mathcal{Y}_0 \\ \mathcal{U}^5\mathcal{Y}_0 &= \frac{2^4}{\Delta^4\sqrt{\Delta}}\mathcal{X}_5 - \frac{2^3}{\Delta^3}\mathcal{Y}_3 - \frac{3\cdot 2^2}{\Delta^2\sqrt{\Delta}}\mathcal{X}_3 + \frac{2\cdot 2}{\Delta}\mathcal{Y}_1 + \frac{1}{\sqrt{\Delta}}\mathcal{X}_1. \end{aligned} \tag{4.11}$$

Notice the pattern in Eq 4.11 that encourages us to define a recursive set that will determine each term in a particular evolutionary step. Define a recursive set $\mathbf{S}_T$, for the step number $T \geq 1$, by;

$$\begin{aligned} \mathbf{S}_T(T-2k) &= \frac{1}{\sqrt{\Delta}}S_{T-1}(T-2k) + \frac{2}{\Delta}S_{T-1}(T-2k-1) \\ \mathbf{S}_T(T-2k-1) &= -\sqrt{\Delta}S_{T-1}(T-2k-1), \end{aligned} \tag{4.12}$$

where $0 \leq k \leq \lfloor \frac{T-1}{2} \rfloor$, with the initial condition $\mathbf{S}_1 = \{\frac{1}{\sqrt{\Delta}}\}$. Notice that if $l > T$ or $l < 1$ then $\mathbf{S}_T(l)$ is undefined; assume $\mathbf{S}_T(l) = 0$ for those values of l. The following lemma shows that $\mathbf{S}_T$ consists of coefficients of $\mathcal{X}_k$ and $\mathcal{Y}_k$ in $\mathcal{U}^T\mathcal{Y}_0$.

Lemma 3. *For $T \geq 1$, let $\mathbf{S}_T$ be the recursive set defined by Eq. 4.12. Then, the terms in $\mathcal{U}^T\mathcal{Y}_0$ are given by*

$$\mathcal{U}^T\mathcal{Y}_0 = \sum_{k=0}^{\lfloor \frac{T-1}{2} \rfloor} \Big(\mathbf{S}_T(T-2k)\mathcal{X}_{T-2k} + \mathbf{S}_T(T-2k-1)\mathcal{Y}_{T-2k-2} \Big). \tag{4.13}$$

Proof. We prove by using the mathematical induction on T. For $T = 2$, use Eq. 4.12 to compute $\mathbf{S}_2$ as

$$\begin{aligned}\mathbf{S}_2(2) &= \frac{1}{\sqrt{\Delta}}\mathbf{S}_1(2) + \frac{2}{\Delta}\mathbf{S}_1(1) = \frac{2}{\Delta\sqrt{\Delta}},\\ \mathbf{S}_2(1) &= -\sqrt{\Delta}\mathbf{S}_1(1) = -1.\end{aligned}$$

Thus, we get $\mathcal{U}^2\mathcal{Y}_0$ as required. Assume that the expression in Eq. 4.13 is true for $T > 2$. Using the expression for $\mathcal{U}^T\mathcal{Y}_0$, compute the coefficients of $\mathcal{X}_{T+1-2k}$ and $\mathcal{Y}_{T+1-2k-2}$ in $\mathcal{U}^{T+1}\mathcal{Y}_0$ as;

$$\begin{aligned}\mathcal{U}(\mathbf{S}_T(T-2k)\mathcal{X}_{T-2k}) &= \mathbf{S}_T(T-2k)\Big(\frac{2}{\Delta}\mathcal{X}_{T-2k+1} - \sqrt{\Delta}\mathcal{Y}_{T-2k-1}\Big)\\ \mathcal{U}(\mathbf{S}_T(T-2k+1)\mathcal{Y}_{T-2k}) &= \mathbf{S}_T(T-2k+1)\Big(\frac{1}{\sqrt{\Delta}}\mathcal{X}_{T-2k+1}\Big)\end{aligned} \tag{4.14}$$

It can be seen that the coefficients of $\mathcal{X}_{T+1-2k}$ and $\mathcal{Y}_{T+1-2k-2}$ in $\mathcal{U}^{T+1}\mathcal{Y}_0$ are computed by using the coefficients of $\mathcal{X}_{T-2k}$ and $\mathcal{Y}_{T-2k}$ in $\mathcal{U}^T\mathcal{Y}_0$, viz.,

$$\begin{aligned}\Big(\frac{1}{\sqrt{\Delta}}\mathbf{S}_T(T-2k+1) + \frac{2}{\Delta}\mathbf{S}_T(T-2k)\Big)\mathcal{X}_{T-2k+1} &= \mathbf{S}_{T+1}(T+1-2k)\mathcal{X}_{T+1-2k}\\ -\sqrt{\Delta}\mathbf{S}_T(T-2k)\mathcal{Y}_{T-2k-1} &= \mathbf{S}_{T+1}(T-2k)\mathcal{Y}_{T+1-2k-2}.\end{aligned} \tag{4.15}$$

This proves that $\mathbf{S}_T$ defines all terms in $\mathcal{U}^T\mathcal{Y}_0$, which is given by Eq. 4.13. □

The previous lemma tells that the recursive set $\mathbf{S}_T$ determines all terms in $\mathcal{U}^T\mathcal{Y}_0$. The recursive set consists of two subsets, one corresponding to the coefficients of $\mathcal{X}_k$ and the other to $\mathcal{Y}_l$, where $0 \leq k, l \leq \lfloor\frac{T-1}{2}\rfloor$. Substitute the second equation of Eq. 4.12 into the first equation to get;

$$\mathbf{S}_T(T-2k) = -\mathbf{S}_{T-2}(T-2k) + \frac{2}{\Delta}\mathbf{S}_{T-1}(T-2k-1). \tag{4.16}$$

Thus, the recursive set $\mathbf{S}_T$ is decomposed into two recursive subsets $\mathbf{S}_T^{(1)}$ and $\mathbf{S}_T^{(2)}$ corresponding to the coefficients of $\mathcal{X}_k$ and $\mathcal{Y}_l$, respectively. Since the two recursive subsets are related by the second part of Eq. 4.12, the solution for the first part

(Eq. 4.16) gives the solution for the other. The following lemma gives the required solution set.

Lemma 4. *Given $T \geq 1$, the evolution after T steps is given by*

$$\begin{aligned}\mathcal{U}^T\mathcal{Y}_0 = \sum_{k=0}^{\lfloor \frac{T-1}{2} \rfloor} \Bigg[& (-1)^k \binom{T-k-1}{k} \frac{1}{\sqrt{\Delta}} \left(\frac{2}{\Delta}\right)^{T-2k-1} \mathcal{X}_{T-2k} \\ & + (-1)^{k+1} \binom{T-k-2}{k} \left(\frac{2}{\Delta}\right)^{T-2k-2} \mathcal{Y}_{T-2k-2}\end{aligned} \tag{4.17}$$

The expression given in Eq. 4.17 can be obtained by using generating function on two variables. Alternatively, we can use mathematical induction on T to prove the above result; see the inductive steps for $\mathbf{S}_T$ in Eq. 4.14 and 4.15.

4.2.2 Hitting times on complete graphs

The complete graph on 2^n vertices is the Cayley graph $Cay(\mathbb{Z}_2^n, \Omega)$, where $\Omega = \mathbb{Z}_2^n \backslash \{\mathbf{0}\}$.

Lemma 5. *The complete graph on 2^n vertices is periodic with minimum period 4.*

Proof. The first three steps in Eq. 4.11 do not allow periodicity in the complete graph. At the fourth step, we compute the coefficients of the final state $|D\rangle\,|v_0\rangle$, which is the initial state in the current scenario, whose amplitude square gives the return probability of the quantum walker. We do so by calculating the coefficient of $|D\rangle\,|v_0\rangle$ in each term of the summation

$$\mathcal{U}^4\mathcal{Y}_0 = \frac{2^3}{\Delta^3\sqrt{\Delta}}\mathcal{X}_4 - \frac{2^2}{\Delta^2}\mathcal{Y}_2 - \frac{2\cdot 2}{\Delta\sqrt{\Delta}}\mathcal{X}_2 + \mathcal{Y}_0.$$

1. Rewrite the first term $\frac{2^3}{\Delta^3\sqrt{\Delta}}\mathcal{X}_4$ in the expanded form (use Eq. 4.9) as

$$\frac{2^3}{\Delta^3\sqrt{\Delta}} \sum_{i_1=1}^{\Delta}\sum_{i_2=1}^{\Delta}\sum_{i_3=1}^{\Delta}\sum_{i_4=1}^{\Delta} |\alpha_{i_k}\rangle\,|\Omega(i_1)\oplus\Omega(i_2)\oplus\Omega(i_3)\oplus\Omega(i_4)\rangle\,,$$

in which the subterms where $\Omega(i_1)\oplus\cdots\oplus\Omega(i_4) = \mathbf{0} = v_0$ are grouped together as

$$\begin{aligned}
&\frac{2^3}{\Delta^3\sqrt{\Delta}}\sum_{i_4=1}^{\Delta}\left(\sum_{\Omega(i_1)\oplus\Omega(i_2)\oplus\Omega(i_3)=\Omega(i_4)} |\alpha_{i_4}\rangle\,|v_0\rangle\right)\\
&=\frac{2^3}{\Delta^3\sqrt{\Delta}}\sum_{i_k=1}^{\Delta}\Delta^2\,|\alpha_{i_k}\rangle\,|v_0\rangle\\
&=\frac{2^3}{\Delta}\,|D\rangle\,|v_0\rangle\,,
\end{aligned} \tag{4.18}$$

where we have used two facts (1) $\frac{1}{\sqrt{D}}\sum_{i_4=1}^{\Delta}|\alpha_{i_k}\rangle\,|v_0\rangle = |D\rangle\,|v_0\rangle$, and (2) the number of terms satisfying $\Omega(i_1)\oplus\cdots\oplus\Omega(i_4)=\mathbf{0}$ is Δ^2.

2. In the second term $-\frac{2^2}{\Delta^2}\mathcal{Y}_2$ and the third term $-\frac{2^2}{\Delta\sqrt{\Delta}}\mathcal{X}_2$ we require subterms satisfying $\Omega(i_1)\oplus\Omega(i_2)=|v_0\rangle$, which gives $-\frac{2^2}{\Delta}$ as the coefficient of $|D\rangle\,|v_0\rangle$ in both the cases.

3. In the fourth term $|D\rangle\,|v_0\rangle$, the coefficient is 1.

Adding all the coefficients obtained above, we get

$$\frac{2^3}{\Delta}-\frac{2^2}{\Delta}-\frac{2^2}{\Delta}+1=1.$$

Thus, the return probability after steps $T=4$ is 1, which implies that all complete graphs on 2^n vertices are periodic with minimum period 4. □

Remarks

The closed form of the state of the quantum system corresponding to a regular graph can be deduced using similar method as given in Lemma 4, with proper expression for $\mathcal{X}_k$ and $\mathcal{Y}_k$ in Eq. 4.9. The exact form for $\mathcal{X}_k$ and $\mathcal{Y}_k$ depends on the graph under consideration and the time step T as described in the special case of the complete graph on 2^n vertices. In general, it is hard to retrieve a target vertex using the given expression in Eq. 4.17 because for each value of T, we need to compute the coefficients of all vertices untill the probability for a vertex is larger than a fixed value p. Therefore, Lemma 4 is not useful unless we find a way to predict a limited number of possible values for T and target vertices. Next section gives a partial solution to this problem.

4.3 Implementation of discrete-time coined quantum walks on cubelike graphs

Because of their universality and potential use in developing algorithms, there is a need for exemplary physical implementations of DTQW. Physical implementations based on ion-traps, optical cavities, and optical lattices have been suggested in [52, 61, 69]. Circuit implementations of various standard quantum algorithms using IBM's Qiskit platform were done in [1]. Recently in [3] implementation of staggered quantum walks on cycles, two-dimensional lattices, and complete graph was studied on IBM quantum computers. A comparison on two different implementation approaches of quantum walks is given in [38]. An implementation of DTQW on hypercubes, complete graphs, complete bipartite graphs, and 2-d lattice is recently given in [67].

In this section, we decompose the unitary operator for the DTQW on cubelike graphs and construct the corresponding quantum circuits. We run these circuits on IBM's quantum simulators and quantum computers, and explore the hitting times and target vertices.

4.3.1 Quantum circuits

A general quantum circuit representing the DTQW on $Cay(\mathbb{Z}_2^n, \Omega)$, with degree $\Delta =\mid \Omega \mid$, is depicted in Fig. 4.3. An unitary operator that can be appended to a quantum circuit has dimension 2^m, for some positive integer m. Since the coin operator acting on the coin space is of dimension Δ, we first study the case where $\Delta = 2^m$, for some positive integer m. All the qubits are initialized to state $|0\rangle$. The position state is represented by the first n qubits, which is initially $|v_0\rangle = |0\rangle^{\otimes n}$. The coin state, represented by the last m qubits, is initialized to diagonal state $|D\rangle = \frac{1}{\sqrt{\Delta}} \sum_{k=1}^{\Delta} |\alpha_{k-1}\rangle$ by applying the Hadamard gate H to each coin qubit.

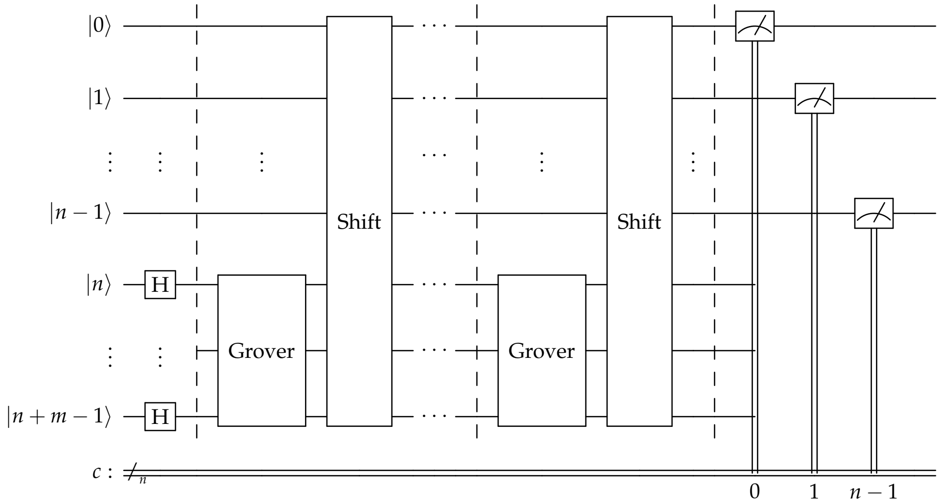


Figure 4.3: A quantum circuit for DTQW on cubelike graph. H is applied to each coin qubit to initialize the coin state to $|D\rangle$. The evolution operator $\mathcal{U} = \mathcal{S}(\mathcal{C} \otimes I)$ is applied for a finite number of times. Each application of $\mathcal{U}$ is separated by a barrier. Finally, measurement operators are applied to the position qubits.

After the initialization, the evolution operator $\mathcal{U}$, which is the composition of the Grover operator $\mathcal{C}$ and the shift operator $\mathcal{S}$, is applied to the circuit for a specific number of times before the measurement is taken. It is important to decompose the evolution operator to run the circuit more efficiently.

Decomposition of the Grover operator

The decomposition of the Grover operator $\mathcal{C}$ has been discussed in [12, 53] (see Fig. 4.4), which is;

$$\mathcal{C} = H^{\otimes m} X^{\otimes m} H \otimes I^{\otimes m-1} C_X^{(m-1)}([0, 1, \ldots, m-2], m-1) H \otimes I^{\otimes m-1} X^{\otimes m} H^{\otimes m},$$

where, $C_X^{(m-1)}(c, t)$ is the generalized Toffoli gate with a set of $m-1$ control qubits, denoted by c, and a target qubit t, i.e., it flips the value of the target only if each qubit in c is at state $|1\rangle$.

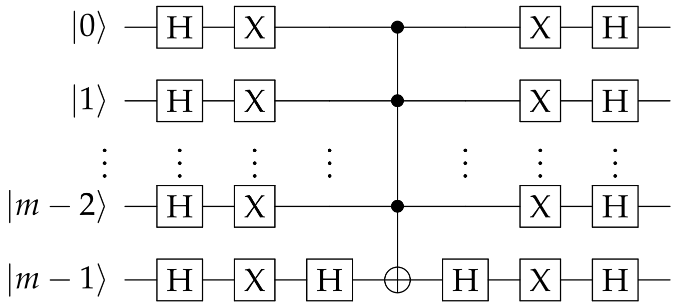


Figure 4.4: A decomposition of the Grover operator.

Decomposition of the Shift operator

The shift operator $\mathcal{S}$ can be expressed as

$$\mathcal{S} = \sum_{k=1}^{\Delta} S_{k-1}, \text{ where } S_{k-1} = \sum_{a=0}^{2^n-1} |\alpha_{k-1}\rangle\, |v_a \oplus \Omega(k)\rangle\, \langle\alpha_{k-1}|\, \langle v_a|\,. \tag{4.19}$$

The operator S_{k-1} shifts the walker only along the edge α_{k-1}. Suppose $\Omega(k)$ has non-zero entries at positions $p_1, p_2, \ldots, p_l$, then we append the extended Toffoli gates $C_X^{(m)}(c, p_j - 1)$, $1 \leq j \leq l$, to the quantum circuit corresponding to S_{k-1}, where, c is the control consisting of all coin qubits and p_j-th position qubit is the target. These gates do not flip the values of the target qubits unless the coin state $|\alpha_{k-1}\rangle$ is $|1^m\rangle$. Therefore, we need to first apply X gate to each coin qubit with value zero in $|\alpha_{k-1}\rangle$ and then apply the extended Toffoli gates. Thus, the sequence of gates appended to the quantum circuit corresponding to S_{k-1} is given by

$$\tilde{S}_{k-1} = (f(k))(C_X^{(m)}(c, p_l - 1) \ldots C_X^{(m)}(c, p_2 - 1) C_X^{(m)}(c, p_1 - 1)), \tag{4.20}$$

where, $f(k)$ is an operator equivalent to appended X gates that transforms $|\alpha_{k-1}\rangle$ to $|1^m\rangle$. Thus, the sequence of operators appended to the quantum circuit corresponding to $\mathcal{S}$ is

$$\mathcal{S} \equiv \tilde{S}_0 \tilde{S}_1 \cdots \tilde{S}_{\Delta-1}. \tag{4.21}$$

where, $S_{k-1}, 1 \leq k \leq \Delta$, is expressed by Eq. (4.20).

Example 3. *We can understand the decomposition of the shift operator through an example of DTQW on Cay($\mathbb{Z}_2^4, \Omega$), with $\Omega = \{0101, 0111, 1001, 1010\}$. Suppose the walker*

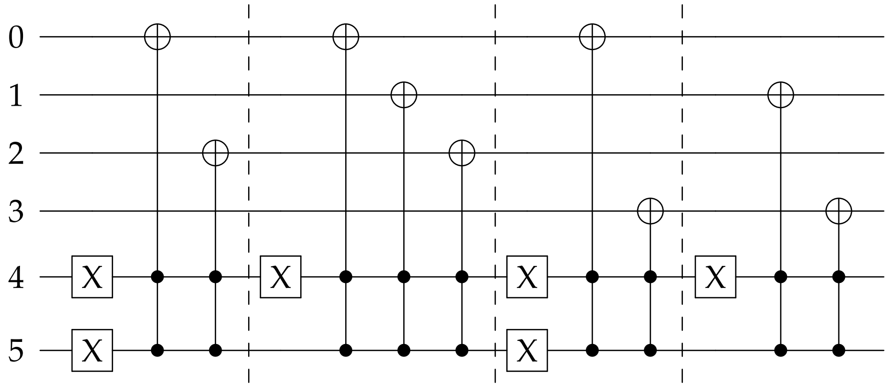


Figure 4.5: A quantum circuit for the shift operator used in DTQW on $Cay(\mathbb{Z}_2^4, \{0101, 0111, 1001, 1010\})$.

is at vertex $v = 1101$, then the shift operation at position state $|v\rangle = |1101\rangle$ is described as;

$$
\begin{aligned}
S\,|\alpha_0\rangle\,|1101\rangle &= |\alpha_0\rangle\,|1101 \oplus 0101\rangle = |\alpha_0\rangle\,|1000\rangle \\
S\,|\alpha_1\rangle\,|1101\rangle &= |\alpha_1\rangle\,|1101 \oplus 0111\rangle = |\alpha_1\rangle\,|1010\rangle \\
S\,|\alpha_2\rangle\,|1101\rangle &= |\alpha_2\rangle\,|1101 \oplus 1001\rangle = |\alpha_2\rangle\,|0100\rangle \\
S\,|\alpha_3\rangle\,|1101\rangle &= |\alpha_2\rangle\,|1101 \oplus 1010\rangle = |\alpha_3\rangle\,|0110\rangle\,.
\end{aligned}
\tag{4.22}
$$

The shift operator, mentioned in Eq. (4.22), can be decomposed as shown in Fig. 4.5. For each edge α_{k-1}, $1 \leq k \leq 4$, we transform $|\alpha_{k-1}\rangle$ to $|\alpha_3\rangle \equiv |11\rangle$ by applying NOT gates, and then apply Toffoli gates to flip the value of position qubits corresponding to non-zero entries in $\Omega(k)$. See Table 4.1, where we have illustrated the transformation of coin qubits. The shift along each direction is given by;

$$
\begin{aligned}
S_0 &= (X \otimes X)(C_X^{(2)}([4,5],0)C_X^{(2)}([4,5],2)) \\
S_1 &= (I \otimes X)(C_X^{(2)}([4,5],0)C_X^{(2)}([4,5],1)C_X^{(2)}([4,5],2)) \\
S_2 &= (X \otimes X)(C_X^{(2)}([4,5],0)C_X^{(2)}([4,5],3)) \\
S_3 &= (I \otimes X)(C_X^{(2)}([4,5],1)C_X^{(2)}([4,5],3))
\end{aligned}
\tag{4.23}
$$

We now discuss how we keep track of changes made to each coin state while

	$X \otimes X$	$I \otimes X$	$X \otimes X$	$I \otimes X$
00	**11**	10	01	00
01	10	**11**	00	01
10	01	00	**11**	10
11	00	01	10	**11**

Table 4.1: Transforming each coin state $|\alpha_{k-1}\rangle, 1 \leq k \leq 4$, to $|11\rangle$ by applying NOT gates.

implementing the shift operation. For $y \in \mathbb{Z}_2^m$, define permutations P_y by

$$P_y(x) = y \oplus x, \ \forall x \in \mathbb{Z}_2^m.$$

We construct a sequence of binary strings $B(\alpha_{k-1}) = 0^{m-r_k}1^{r_k}$, $1 \leq k \leq 2^m$, where $m - r_k$ is the position of the last non-zero bit of the k-th binary string α_{k-1} in $\mathbb{Z}_2^m$. Note that $B(0^m) = 1^m$.

Claim:

$$P_{B(\alpha_{k-1})}(P_{B(\alpha_{k-2})}(\cdots(P_{B(\alpha_0)}(\alpha_{k-1}))\cdots)) = 1^m. \tag{4.24}$$

If the claim is true, then $f(k) = I^{m-r_k}X^{r_k}$, $1 \leq k \leq 2^m$, transforms each coin state $|\alpha_{k-1}\rangle$ to $|1^m\rangle$, one at a time and therefore the operator given by Eq. (4.20) is correct.

We now prove the claim by induction on k. For $k = 1$, we have $\alpha_0 = 0^m$ and $B(\alpha_0) = 1^m$. Thus,

$$P_{B(\alpha_0)}(\alpha_0) = 1^m \oplus 0^m = 1^m.$$

Assume that at k-th iteration, α_{k-1} gets transformed to 1^m. In lexicographical order of binary strings, the next binary string α_k is obtained by

1. finding the last bit which is zero,

2. changing this bit to 1 and all the following bits to 0s.

Thus, α_{k-1} and α_k differ at the last r positions, for some $1 \leq r \leq m$, and $B(\alpha_k) = 0^{m-r}1^r$. Therefore, α_k gets transformed to $1^{m-r}0^r$ at k-th iteration, and at $(k+1)$-th iteration we get $B(\alpha_k) \oplus \alpha_k = 1^m$. Hence, the claim is True.

Notice that, the transformation of a coin state corresponding to an edge alters other coin states corresponding to other edges. At the end of the shift operation in

Example 3 we retrieve the original generic coin state as shown in the last column of Table 4.1. Indeed, after the 2^m-th iterations, notice that the i-bit of α_{k-1}, $1 \leq i \leq m$ and $1 \leq k \leq 2^m$, flips only if $B(\alpha_{j-1})$, $1 \leq j \leq 2^m$, is of the form $0^{m-s}1^s$, with $s \geq i$, which corresponds to binary strings of the form $x_{m-1} \cdots x_i 0^i$. Since the number of such binary strings are 2^{m-i}, the i-th bit of α_{k-1} flips even number of times. Therefore, each coin state $|\alpha_{k-1}\rangle$, $1 \leq k \leq 2^m$, remains unchanged after the 2^m-th iterations. Therefore, the shift operation is performed successfully without altering the coin state.

Analysis of the quantum circuit for the shift operator

The number of X gates $Num(X)$ used in the decomposition of the shift operator is equal to

$$Num(X) = \sum_{x \in \mathbb{Z}_2^m} wt(B(x)) = 2^{m+1} - 2, \tag{4.25}$$

where $wt(B(x))$ is the Hamming weight of $B(x)$. This can be proved by induction on m. For $m = 1$, $Num(X) = 2$ because $B(0) = 1$ and $B(1) = 1$. Assume by way of induction that, for $m = k$, $Num(X) = 2^{k+1} - 2$. Then, for $m = k + 1$, notice that if $x \in \mathbb{Z}_2^m$ then $0x, 1x \in \mathbb{Z}_2^{m+1}$ and,

$$B(bx) = \begin{cases} B(x), & \text{if } x \neq 0^m \\ B(x) + 1, & \text{if } x = 0^m. \end{cases} \tag{4.26}$$

where $b = 0$ or 1. Therefore, $Num(X)$ for $m = k + 1$ is equal to

$$2 \times (2^{k+1} - 2) + 2 = 2^{k+2} - 2.$$

$Num(X)$ is of order $\mathcal{O}(\mid \Omega \mid)$. For most practical purposes sparse Cubelike graphs are used. In particular, if we take $m = \mathcal{O}(log\ n)$, then $Num(X)$ is of order $\mathcal{O}(n)$. This implies that the circuit is efficient for most cubelike graphs of interest. The number of generalized Toffoli gates used is equal to the sum of Hamming weights of all elements of the generating set Ω, which is

$$Num(C_X^{(m)}) = \sum_{x \in \Omega} wt(x), \qquad wt(x) = \text{number of 1s in } x. \tag{4.27}$$

Quantum circuits for cubelike graphs of arbitrary degree

If the degree of the cubelike graph is not a power of 2 (for example $\mathcal{Q}_3$) then we have to make certain modifications to implement our circuit. Consider a cubelike graph $Cay(\mathbb{Z}_2^n, \Omega)$, with $\Delta =\mid \Omega \mid$. Let m be a positive integer satisfying $2^{m-1} < \Delta \leq 2^m$. The Grover coin $\mathcal{C} = 2\left|D\right\rangle\left\langle D\right| - I$ is a $\Delta \times \Delta$ matrix, where Δ may not be equal to 2^m. We, therefore, define a new $2^m \times 2^m$ operator $\mathcal{C}'$ by

$$\mathcal{C}' = \begin{bmatrix} 2\left|D\right\rangle\left\langle D\right| & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} - I = 2\left|D'\right\rangle\left\langle D'\right| - I,$$

where,

$$\left|D'\right\rangle = \frac{1}{\sqrt{\Delta}} \sum_{k=1}^{\Delta} \left|\alpha_{k-1}\right\rangle \in \mathbb{C}^{2^m}$$

is a projection of the diagonal state $\left|D\right\rangle$ in the higher dimensional Hilbert space $\mathcal{H}_{\mathcal{C}'} = \mathbb{C}^{2^m}$ that contains the coin space $\mathcal{H}_{\mathcal{C}} = \mathbb{C}^{\Delta}$ as a subspace. In other words, $\mathcal{H}_{\mathcal{C}'}$ is a new coin space with the computational basis

$$\{\left|\alpha_{k-1}\right\rangle : 1 \leq k \leq 2^m\},$$

and the quantum walk occurs in $Cay(\mathbb{Z}_2^n, \Omega')$, where Ω' has 2^m elements containing Ω, i.e., $Cay(\mathbb{Z}_2^n, \Omega)$ is a subgraph of $Cay(\mathbb{Z}_2^n, \Omega')$. The new coin operator $\mathcal{C}'$ changes the coefficients of the initialized vector, which is $\left|D'\right\rangle$, and therefore, the coefficents of coin states $\left|\alpha_{k-1}\right\rangle$, with $k \geq \Delta + 1$, remain 0 throughout the evolution, i.e. the generic coin state $\left|c\right\rangle$, in our case, is

$$\left|c\right\rangle = \sum_{k=1}^{\Delta} \lambda_k \left|\alpha_{k-1}\right\rangle + \sum_{k=\Delta+1}^{2^m} 0 \left|\alpha_{k-1}\right\rangle.$$

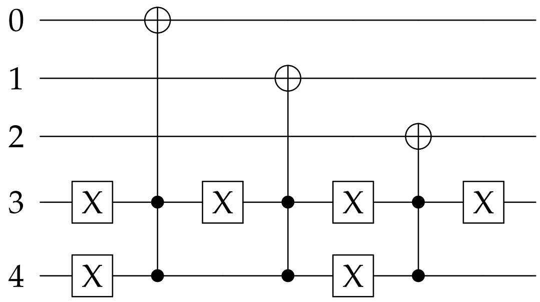


Figure 4.6: A quantum circuit for the shift operator on $\mathcal{Q}_3$.

We define a new shift operator $\mathcal{S}'$ by

$$\begin{aligned}\mathcal{S}' &= \sum_{k=1}^{\Delta} \sum_{a=0}^{2^n-1} |\alpha_{k-1}\rangle |v_a \oplus \Omega(k)\rangle \langle \alpha_{k-1}| \langle v_a| \\ &+ \sum_{k=\Delta+1}^{2^m} \sum_{a=0}^{2^n-1} |\alpha_{k-1}\rangle |v_a\rangle \langle \alpha_{k-1}| \langle v_a| \, .\end{aligned} \tag{4.28}$$

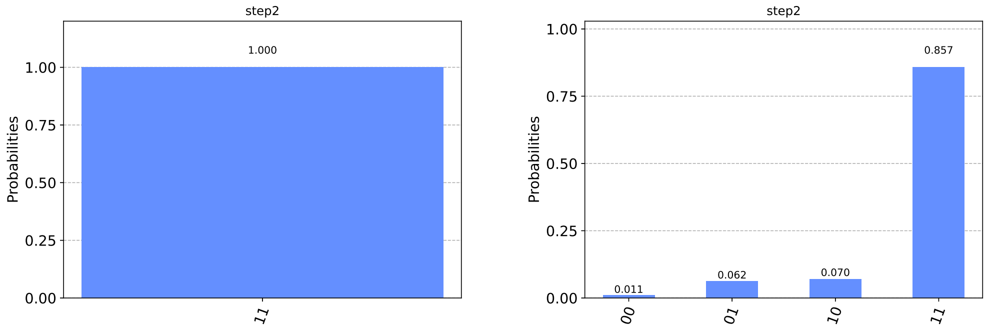
The new shift operator S' fixes $|\alpha_{k-1}\rangle |v_a\rangle$, $0 \leq a \leq 2^n - 1$, if $k \geq \Delta + 1$, i.e.,

$$S' |\alpha_{k-1}\rangle |v_a\rangle = \begin{cases} |\alpha_{k-1}\rangle |v_a \oplus \Omega(k)\rangle \, ; & 1 \leq k \leq \Delta, \\ |\alpha_{k-1}\rangle |v_a\rangle \, ; & k \geq \Delta + 1. \end{cases}$$

The Fig. 4.6 represents the quantum circuit for $\mathcal{Q}_3$. Notice that the circuit does not shift a position state along the direction $|\alpha_3\rangle \equiv |11\rangle$. Finally, the original coin state is retrieved at the end of the shift operation because each quantum wire contains even number of X gates.

4.3.2 Application to hitting times

We use Qiskit [2] to implement DTQW on IBM's Quantum simulators and quantum computers. The Qiskit simulator *qasm_simulator* runs locally and other IBM's simulators such as *simulator_mps*, *simulator_extended_stabilizer*, etc, are accessed via IBM provider *ibm-q*.



(a) *qasm_simulator* (b) *ibmq_manila*

Figure 4.7: Probability distribution of DTQW on $\mathcal{Q}_2$ after two steps when run on Qiskit simulator (a) *qasm_simulator* and on IBM's quantum computer (b) *ibmq_manila*.

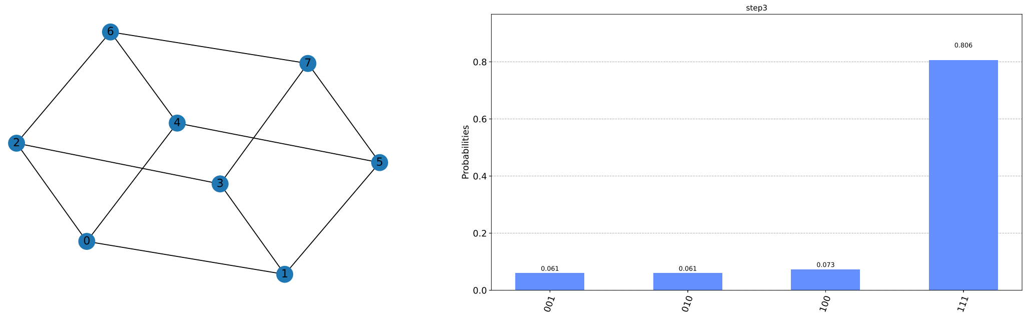


Figure 4.8: DTQW on $\mathcal{Q}_3$ with steps $T = 3$, target vertex 111, and target probability $p = 0.804$.

Discrete-time coined quantum walks on Quantum computers

The quantum circuit for DTWQ on $\mathcal{Q}_2$, see Fig. B.1, was run on IBM's quantum computer *ibmq_manila v1.0.3*, the result of which is compared with the result on *qasm_simulator*, as shown in Fig. 4.7. The quantum circuits corresponding to higher dimensional cubelike graphs could not be run successfully on real quantum computers due to unavoidable gate errors and noise from the environment around the quantum computer. In case of $\mathcal{Q}_3$, the errors and noise were low and therefore could be run successfully.

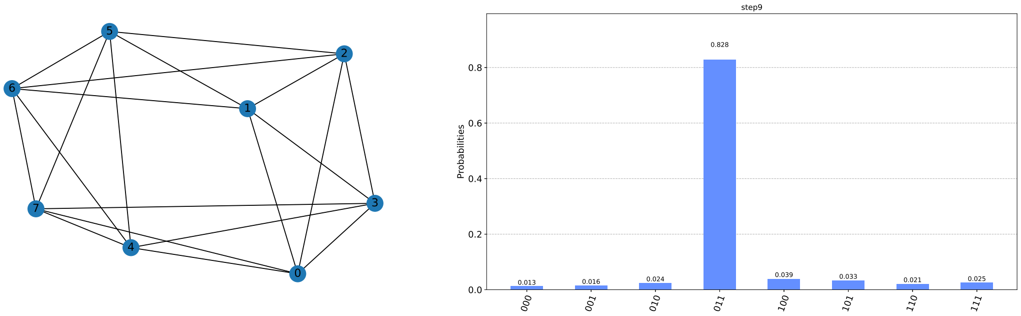


Figure 4.9: DTQW on $\mathcal{AQ}_3$ with $T = 9$, target 011, and $p = 0.812$.

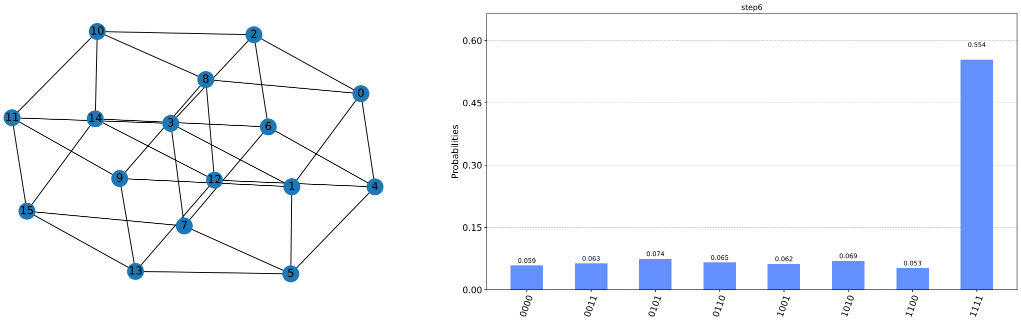


Figure 4.10: DTQW on $\mathcal{Q}_4$ with $T = 6$, target 1111, and $p = 0.562$.

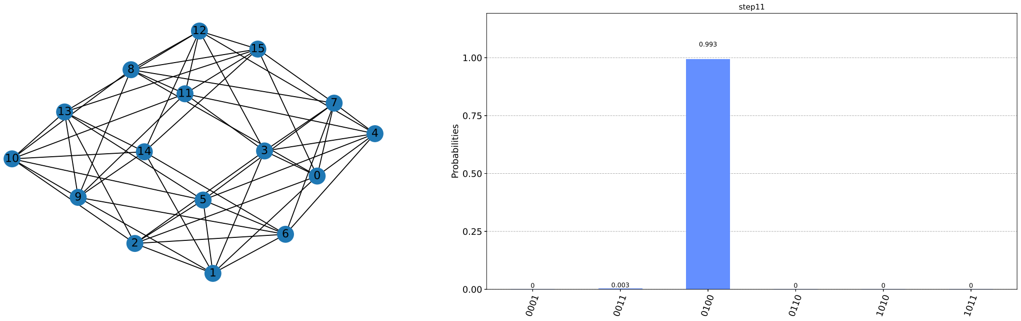


Figure 4.11: DTQW on $\mathcal{AQ}_4$ with $T = 11$, target 0100, and $p = 0.993$.

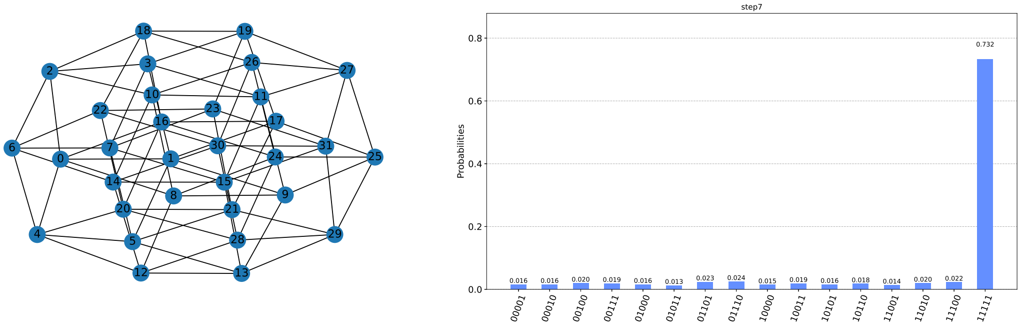


Figure 4.12: DTQW on $\mathcal{Q}_5$ with $T = 7$, target 11111, and $p = 0.722$.

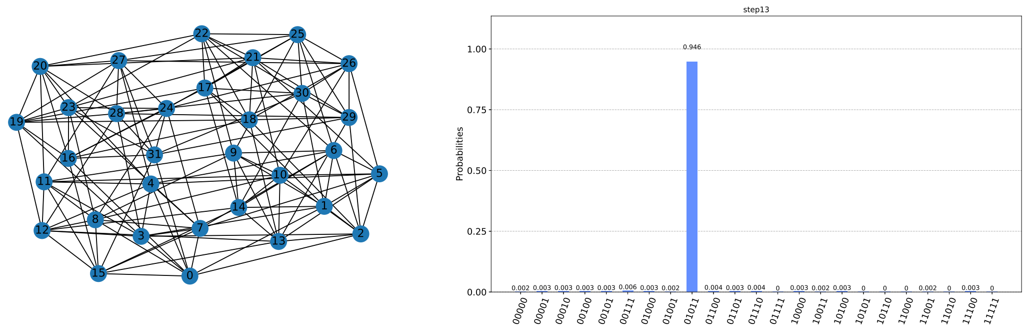
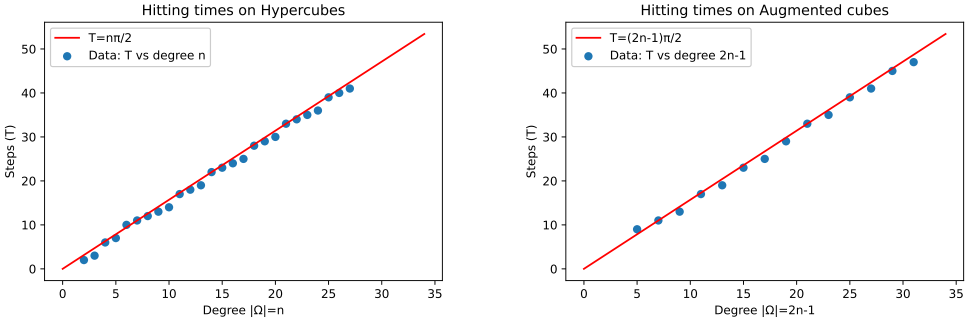


Figure 4.13: DTQW on $\mathcal{AQ}_5$ with $T = 13$, target 01011, and $p = 0.953$.



(a)

(b)

Figure 4.14: Plot of hitting time T vs the degree (a) $\mathcal{Q}_n$ and (b) $\mathcal{AQ}_n$.

Hitting times of Hypercubes and Augmented cubes

Upon implementing quantum circuits for DTQW on cubelike graphs, we have observed that the specific vertex, called the target vertex, to which the walker reach with highest probability, is unique and equal to the binary XOR operations of elements of the generating set Ω, i.e.,

$$\text{target vertex } = \bigoplus_{x \in \Omega} x. \tag{4.29}$$

Therefore, the target vertex in the Hypercube of dimension n is 1^n and in Augmented cube of dimension n the target vertex is $(01)^x 1$, where $x = \frac{n-1}{2}$, if n is odd and $(01)^y 00$, where $y = \frac{n-2}{2}$, if n is even. In Fig. 4.8, 4.9, 4.10, 4.11, 4.12 and 4.13 the target probabilities for Augmented cubes are higher than that of Hypercubes of same dimensions. In Table 4.2, we have displayed the hitting time T along

Table 4.2: Hitting time T with target vertex and target probability p corresponding to dimensions of Hypercubes $\mathcal{Q}_n$ and Augmented cubes $\mathcal{AQ}_n$.

Hypercubes $\mathcal{Q}_n$			
$n = \Delta$	T	Target	p
3	3	111	0.804
4	6	1111	0.562
5	7	11111	0.722
6	10	1^6	0.816
7	11	1^7	0.912
8	12	1^8	0.954
9	13	1^9	0.950
10	14	1^{10}	0.901
11	17	1^{11}	0.927
12	18	1^{12}	0.956
13	19	1^{13}	0.947
14	22	1^{14}	0.929
15	23	1^{15}	0.961
16	24	1^{16}	0.960

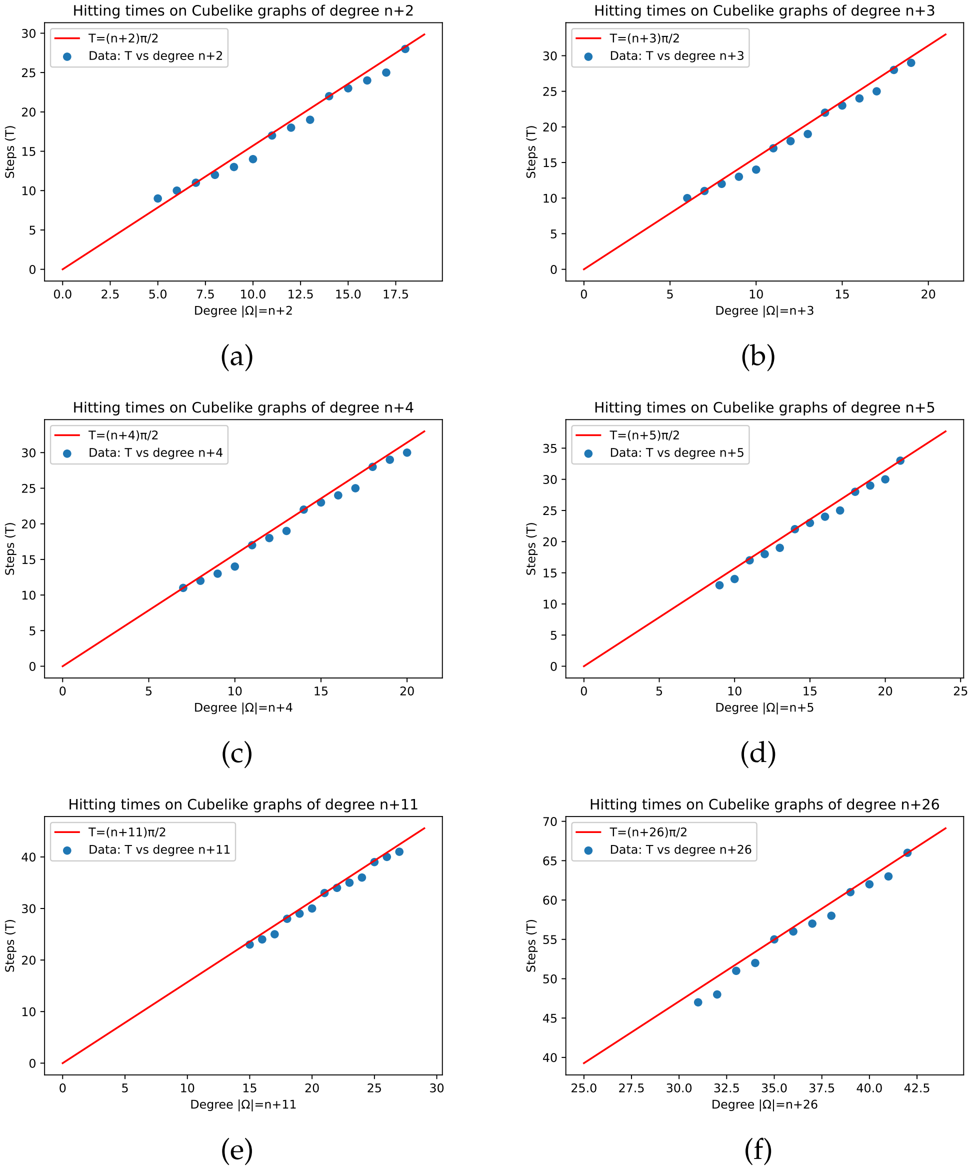
Augmented cubes $\mathcal{AQ}_n$				
n	Δ	T	Target	p
3	5	9	011	0.812
4	7	11	0100	0.993
5	9	13	01011	0.953
6	11	17	$(01)^2 00$	0.928
7	13	19	$(01)^3 1$	0.919
8	15	23	$(01)^3 00$	0.969
9	17	25	$(01)^4 1$	0.926
10	19	29	$(01)^4 00$	0.955
11	21	31	$(01)^5 1$	0.919
12	23	35	$(01)^5 00$	0.954
13	25	39	$(01)^6 1$	0.958
14	27	41	$(01)^6 00$	0.962
15	29	45	$(01)^7 1$	0.977
16	31	47	$(01)^7 00$	0.961

with target and target probability corresponding to dimensions of $\mathcal{Q}_n$ and $\mathcal{AQ}_n$. In either case, the target probability gets closer to 1 as the dimension increases. In case of Hypercubes, T is linear with n that tallies with the theoretical result stated by Kempe in [49], while in Augmented cubes it is linear with the degree of the graph. See Fig. 4.14b, where we have shown that T is linear with the degree $2n - 1$ of $\mathcal{AQ}_n$.

4.3.3 A conjecture on hitting times and target vertex

The result mentioned by J. Kempe in [49] can be generalized to other cubelike graphs. In Fig. 4.15 and 4.16 we have shown plots of steps T verses degrees $n + k$ of some cubelike graphs, where T is the number of iterations required to hit the target vertex, and n is the dimension of cubelike graph of degree $n + k$. In the first figure, we fix the value of k and increase n, and in the second figure, we fix the dimension n and increase the value of k. In either case, we find that T verses $n + k$ plot is linear and follows the relation;

$$T \approx (n + k) \times \pi/2. \tag{4.30}$$

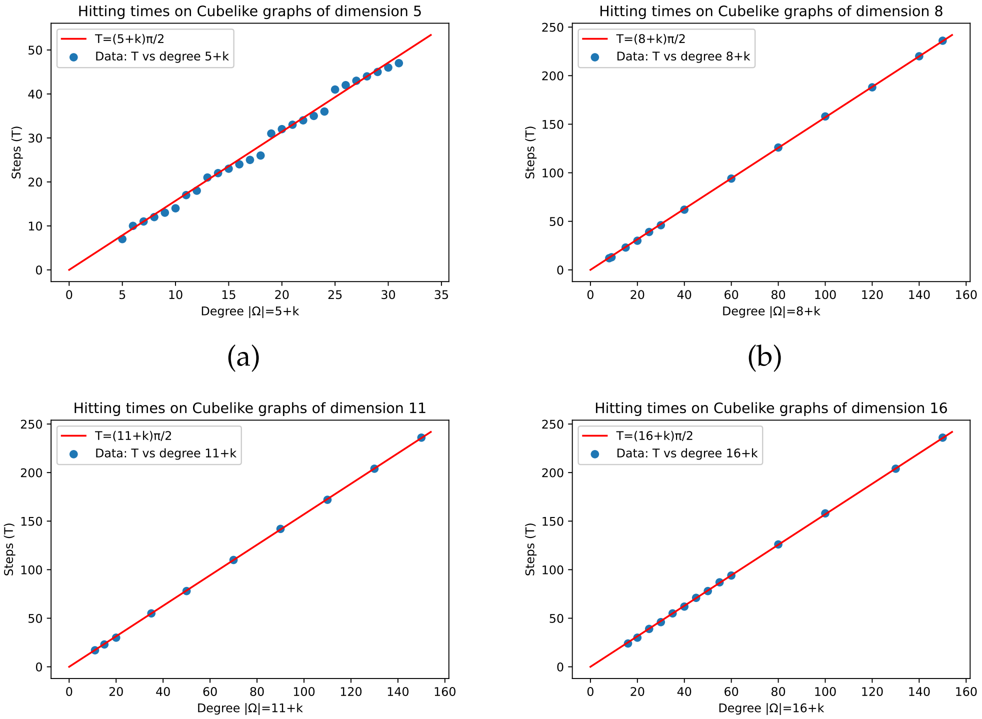


(a) (b)

(c) (d)

(e) (f)

Figure 4.15: Plots of steps T required to attain the target probability verses the degree $| \Omega | = n + k$ of cubelike graphs, where k is fixed and n is the dimension.



(a)

(b)

(c)

(d)

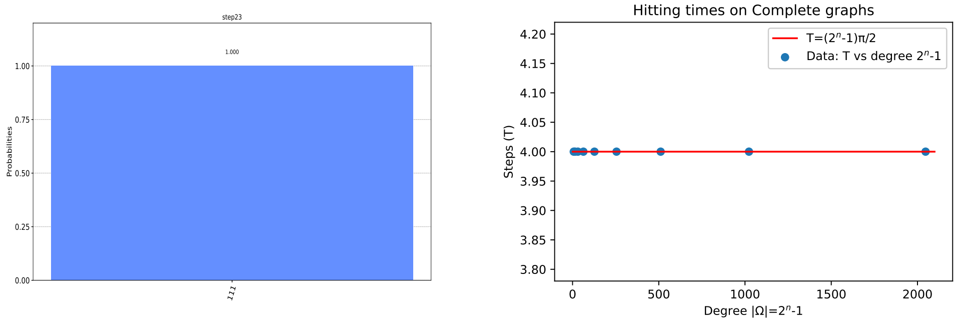
Figure 4.16: Plots of steps T requried to attain the target probability verses the degree $\mid \Omega \mid = n + k$ of cubelike graphs of fixed dimension n.

It is to be noted that it is just a coincidence that T vs dimension plot is same as T vs degree plot for DTQW on Hypercubes. The observations and remarks made above lead to the following conjecture.

Conjecture 1. *Let $\Gamma = Cay(\mathbb{Z}_2^n, \Omega)$ be an n-dimensional Cubelike graph with degree $\Delta = \mid \Omega \mid$. Define the target vertex by;*

$$v_{targ} = \bigoplus_{x \in \Omega} x. \tag{4.31}$$

Then, there exists a hitting time $T \approx \frac{\pi\Delta}{2}$ such that the target probability p_{targ}, the probability by which DTWQ reaches v_{targ}, is asymptotically equal to 1. Moreover, the parity of T is same as that of Δ, i.e., T is even only if Δ is even.



(a) (b)

Figure 4.17: (a) Probability distribution of $\mathcal{Q}_3$ after step $T = 23$, and (b) Plot for hitting time T verses degree $2^n - 1$ of complete graph on 2^n vertices.

Remarks

We implemented efficient quantum circuits for discrete quantum random walks on families of cubelike graphs such as hypercubes and augmented cubes. Our implementations show that the hitting times of all cubelike graphs are asymptotically linear in the degree Δ of the graphs. That is, for the hitting time $\approx \frac{\pi\Delta}{2}$, the probability that the walker is at the target vertex approaches 1 as Δ approaches infinity. Our circuits run on IBM's quantum computing platform Qiskit, both on real quantum computers and simulators. We note that $T \approx \frac{\Delta\pi}{2}$ is not necessarily the minimum hitting time or its multiple for a Δ-regular cubelike graph. For example, in DTQW on $\mathcal{Q}_3$, see Fig. 4.17a, the walker hits the target vertex $|111\rangle$ after steps $T = 23$ with probability 1. Another example is that of complete graphs $Cay(\mathbb{Z}_2^n, \Omega)$, with $\Omega = \mathbb{Z}_2^n \backslash \{0^n\}$. In Fig. 4.17b, the hitting time for the complete graphs on 2^n vertices has been shown constantly with $T = 4$ and target probability equal to 1, which we have proved analytically. It will be interesting to find other hitting times for a cubelike graph, particularly the minimum one for which the walker hits the target with high probability. Further, studying this problem on other regular graphs would be interesting, particularly Cayley graphs.

4.4 Summary

In section 4.1, we computed a closed form for the state $\mathcal{U}^T |D\rangle |v_0\rangle$ (see Eq. 4.17) of the quantum system associated with a cubelike graph. Through Conjecture 1, we are aware of the unique target vertex v_{targ} in a cubelike graph. Therefore, we only need to extract the coefficient of $|D\rangle |v_{targ}\rangle$ from the expressions for $\mathcal{X}_k$ and $\mathcal{Y}_k$ given in Eq. 4.9. Thus, we can obtain the expression for the amplitude of the target vertex and subsequently verify our experimental results. We did try this approach for hypercubes but could not derive the result for hitting times.

CHAPTER 5

Continuous-time quantum walks on Cayley graphs

Continuous-time quantum walks (CTQW) are generalization of continuous-time classical random walks into the quantum world. CTQW is universal for quantum computation [23, 26, 36] and is hence predominantly used in designing quantum algorithms. Some of the earlier work on CTQW can be found in [4, 24, 25, 35, 57]. An important feature of a quantum walk is the quantum state transfer from one vertex to another. It has been found that many families of graphs allow transmission of quantum states with fidelity equal to unity, i.e., the state transfer is perfect (see [14, 18, 20, 22, 29, 39, 40, 48]). Among these graph families, cubelike graphs are the most versatile ones and their properties have been characterized for determining the existence and identifying the pair of vertices admitting perfect state transfer in constant time. It can be easily shown that not all cubelike graphs admit perfect state transfer (PST). For instance, K_{2^n} is a cubelike graph that does not admit PST. However, all cubelike graphs are periodic with period dividing π [40]; in fact, all integral graphs are periodic and cubelike graphs are integral graphs.

In this chapter, we investigate weighted Cayley graphs over $\mathbb{Z}_r^n$ in order to classify them into three categories of graphs - those that admit PST, those that do not admit PST but are periodic, and those that are not periodic. We have also constructed efficient quantum circuits for the CTQW on weighted cubelike graphs and have run these circuits on IBM's quantum computing platform and verified the results on PST [58].

5.1 Perfect state transfer

Let Γ be an undirected and weighted graph with loops and A be the adjacency matrix. A quantum walk on Γ is described by an evolution of the quantum system associated with the graph. Suppose the graph has N vertices, then it is associated with a Hilbert space $\mathcal{H}_P \cong \mathbb{C}^N$, called the position space, and the computational basis for $\mathcal{H}_P$ is represented by;

$$\{|v\rangle : v \text{ is a vertex in } \Gamma\}.$$

The continuous-time quantum walk on Γ is described by the transition matrix $\mathcal{U}(t) = e^{\imath tA}$, where $\imath = \sqrt{-1}$, i.e., if $|\psi(0)\rangle$ is the initial state of the system then, at time t, the state of the system is given by

$$|\psi(t)\rangle = e^{\imath tA} |\psi(0)\rangle .$$

The matrix exponential of $\imath tA$ can be seen as

$$\mathcal{U}(t) = \sum_{k=0}^{\infty} \imath^k \frac{t^k A^k}{k!}.$$

Since A is symmetric, $\mathcal{U}(t)$ is symmetric, and $\overline{e^{\imath tA}} = e^{-\imath tA}$ implies $\mathcal{U}(t)$ is unitary. Moreover,

$$\mathcal{U}(t_1 + t_2) = e^{\imath(t_1+t_2)A} = e^{\imath t_1 A} e^{\imath t_2 A} = \mathcal{U}(t_1)\mathcal{U}(t_2).$$

As a special case of the spectral theorem for normal matrices, see Theorem 33 in Appendix A, we express the transition matrix $\mathcal{U}(t)$ in a more useful way.

Remark 3. *Let Γ be a graph on N vertices with the adjacency matrix A. Suppose P is an orthogonal matrix such that $D = P^T AP$ is real diagonal, and let P_{*k} denote the k-th* *column of P, then the transition matrix $\mathcal{U}(t) = e^{\iota t A}$ is expressed as*

$$\begin{aligned}\mathcal{U}(t) &= Pe^{\iota t D}P^T \\ &= \sum_{k=1}^{N} e^{\iota t D_{k,k}} \left|P_{*k}\right\rangle \left\langle P_{*k}\right| \\ \implies \mathcal{U}(t)_{u,v} &= \sum_{k=1}^{N} e^{\iota t D_{k,k}} P_{u,k}P_{v,k}.\end{aligned} \tag{5.1}$$

Definition 3. *A graph is said to admit perfect state transfer, if the quantum walker beginning at some vertex u reaches a distinct vertex v with probability 1, i.e., $\mathcal{U}(\tau)\left|u\right\rangle = \lambda\left|v\right\rangle$, for some $\lambda \in \mathbb{C}$, satisfies*

$$\langle v|e^{\iota\tau A}|u\rangle = |\lambda|^2 = 1, \qquad \textit{for some real time } \tau.$$

Alternatively, we say perfect state transfer occurs from the vertex u to the vertex v.

Example 4. *Consider the graph on $\mathcal{Q}_2$, see Fig. 4.2. The adjacency matrix A of the graph is given by*

$$A = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix},$$

with spectral decomposition

$$A = PDP^T, \textit{ where } P = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}, \textit{ and } D = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2 \end{bmatrix}.$$

Therefore, the spectral decomposition for the transition matrix with time $t = \pi/2$ is

$$\mathcal{U}(t = \pi/2) = \sum_{k=1}^{4} e^{\imath \frac{\pi}{2} D_{k,k}} |P_{*k}\rangle \langle P_{*k}| = \begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}.$$

Thus, perfect state transfer occur between the pairs $(1, 4)$ and $(2, 3)$, both in time $\frac{\pi}{2}$.

Lemma 6. *If perfect state transfer occurs from a vertex u to a vertex v in a graph, then perfect state transfer occurs from v to u.*

Proof. Suppose for some real τ and scalar λ, $\mathcal{U}(\tau) |u\rangle = \lambda |v\rangle$, with $|\lambda| = 1$. Since $\mathcal{U}(\tau)$ is symmetric,

$$\mathcal{U}(\tau) |v\rangle = \lambda |u\rangle .$$

□

Notice that $\mathcal{U}(\tau)^2 |u\rangle = \lambda^2 |u\rangle$. But, $\mathcal{U}(\tau)^2 = \mathcal{U}(2\tau)$. Therefore, the quantum walker returns back to the starting vertex u after time 2τ. Such property is termed as periodicity.

Definition 4. *If a quantum walker beginning at a vertex u in a graph and returns back after time τ with one probability, i.e.,*

$$\mathcal{U}(\tau) |u\rangle = \lambda |u\rangle , \qquad |\lambda| = 1,$$

we say the graph is periodic at u or it is periodic relative to u, with period τ. If the graph is periodic relative to every vertex with the same period τ, we say the graph is periodic.

Lemma 7. *If a graph admits perfect state transfer from vertex u to v, then the graph is periodic at u.*

Lemma 8. *If perfect state transfer occurs between u and v, and between u and w, then $v = w$.*

There are graphs which are periodic at a vertex, or periodic at each vertex but do not admit perfect state transfer. For example, star $K_{1,n}$ is periodic at one vertex and complete graph K_n is periodic, but none of them has PST pairs. In general, simple graphs do not admit perfect state transfer. However, by assigning weights to edges some of them may show the occurrence of perfect state transfer. In [9], it is shown that the join of a weighted two-vertex graph with any regular graph has perfect state transfer. For more reference on weighted graphs allowing PST, see [8, 37, 66].

5.2 Cayley graphs over $\mathbb{Z}_r^n$ that have the same eigenvectors

In this section, we study complex-weighted undirected graphs whose adjacency matrix is a normal matrix. We use this to identify Cayley graphs which have the same set of eigenvectors. The motive is to study continuous-time qauntum walk on these graphs.

5.2.1 Construction of graphs that have the same eigenvectors

Given a unitary matrix P, we want to find all normal matrices whose eigenvectors form columns of P. The simplest way is to choose any complex diagonal matrix D that gives a normal matrix $A = PDP^{\dagger}$. The following lemma does the same thing but in a different manner.

Theorem 15. *Let P be an $N \times N$ unitary matrix. Then,*

1. *there exists an $N \times N$ invertible matrix Q, whose each row is the Hadamard product (entrywise product) of a row of P with the complex-conjugate of another row of P,*

2. *for every column matrix $Z \in \mathbb{C}^{N\times 1}$, there is a normal matrix A whose eigenvectors form columns of P and its eigenvalues are given by $X = Q^{-1}Z$.*

Proof. Define the matrix S by

$$S = \begin{bmatrix} P_{*1} \otimes \bar{P}_{*1} & \cdots & P_{*N} \otimes \bar{P}_{*N} \end{bmatrix}, \tag{5.2}$$

where the j-th column of S is the tensor product of the j-th column of P with its complex-conjugate. Clearly, S has rank N, and each row of S is the Hadamard product of two rows of P. Suppose rows corresponding to N paired indices $\mathcal{I} = \{(r_j, c_j) : 1 \leq j, r_j, c_j\}$ are linearly independent, then the submatrix Q, given by

$$Q = \begin{bmatrix} P_{r_1,1}\bar{P}_{c_1,1} & \cdots & P_{r_1,N}\bar{P}_{c_1,N} \\ \vdots & \ddots & \vdots \\ P_{r_N,1}\bar{P}_{c_N,1} & \cdots & P_{r_N,N}\bar{P}_{c_N,N} \end{bmatrix}, \tag{5.3}$$

is invertible. Let $Z \in \mathbb{C}^{N\times 1}$ and put $X = Q^{-1}Z$. We now construct the normal matrix A uniquely associated with Z with eigenvalues X. Assign values to N entries in A corresponding to the index set $\mathcal{I}$ as;

$$\begin{bmatrix} A_{r_1,c_1} \\ \vdots \\ A_{r_N,c_N} \end{bmatrix} = Z. \tag{5.4}$$

Then, the two system of linear equations

$$QX = Z \text{ and } SX = Y, \text{ where } Y = \begin{bmatrix} A_{*1} \\ \vdots \\ A_{*N} \end{bmatrix}, \tag{5.5}$$

have same solutions for X. Alternatively, $SX = Y$ can be written as

$$\begin{gathered} A_{i,j} = \sum_{k=1}^{N} X_{k,1} P_{i,k} \bar{P}_{j,k}, \qquad 1 \leq i,j \leq N \\ \implies A = PDP^{\dagger}, \text{ where } D \text{ is diagonal with } D_{k,k} = X_{k,1} \end{gathered} \tag{5.6}$$

Thus, for every Z we get a normal matrix A whose eigenvectors form columns of P and has eigenvalues $Q^{-1}Z$. □

Theorem 15 gives a method to construct a family of graphs that have the same set of eigenvectors. For each column matrix Z in the theorem, there is a unique graph Γ associated with Z, with the fixed set of eigenvectors forming the columns of P. Thus, the adjacency matrix A is uniquely determined by P and Z, with the eigenvalues $X = Q^{-1}Z$. Notice that, the invertible matrix Q need not be unique. In fact, if Q_1 and Q_2 are two distinct invertible matrices in Theorem 15, then they are row-equivalent matrices, i.e., Q_2 can be obtained from Q_1 by performing elementary row operations on Q_1.

Example 5. *Consider the* 4×4 *normalized Hadamard matrix*

$$P = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}.$$

Then, the first four rows of S, see Eq. 5.2, are linearly independent. Hence, the matrix $Q = \frac{1}{4}P$, see Eq. 5.3, is invertible. Suppose $Z = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$, then the associated adjacency matrix A has eigenvalues $X = Q^{-1}Z = \begin{bmatrix} 2 \\ 0 \\ -2 \\ 0 \end{bmatrix}$. The corresponding graph is the 2*-dimensional hypercube Q_2. If $Z = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}$, then the associated graph is a complete graph on* 4 *vertices with eigenvalues $X = \begin{bmatrix} 3 \\ -1 \\ -1 \\ -1 \end{bmatrix}$. For $Z = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$, we get a disconnected graph*

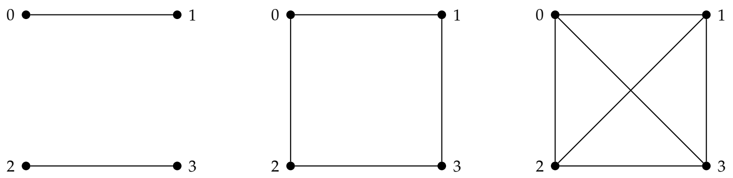


Figure 5.1: All the three graphs, on four vertices each, have the same set of eigenvectors forming the columns of P in Example 5, with the associated $Z = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$, $\begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$, and $\begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}$, respectively.

$K_2 \cup K_2$, the union of two complete graphs on two vertices, with eigenvalues $X = \begin{bmatrix} 1 \\ -1 \\ 1 \\ -1 \end{bmatrix}$.

Corollary 2. *Suppose P is orthogonal in Theorem 15, then for every $Z \in \mathbb{R}^{N \times 1}$, the associated matrix A is real and symmetric, with real eigenvalues.*

Lemma 9. *If a row of P in Theorem 15 has all entries non-zero, say the k-th row, then the matrix Q is given by*

$$Q = \begin{bmatrix} \bar{P}_{k,1} P_{*1} & \cdots & \bar{P}_{k,N} P_{*N} \end{bmatrix}. \tag{5.7}$$

Proof. Suppose the k-th row of P has all entries non-zero, and for some scalars $\lambda_1, \ldots, \lambda_N$,

$$\sum_{j=1}^{N} \lambda_j \bar{P}_{k,j} P_{*j} = 0.$$

Since, the columns of P form a linearly independent set, $\lambda_j \bar{P}_{k,j} = 0$ for all $1 \leq j \leq N$. But, $\bar{P}_{k,j} \neq 0$ implies $\lambda_j = 0$, for all values of j. Hence, the columns of Q form a linearly independent set and thus Q is invertible. □

Corollary 3. *If the k-th row of P has constant entries $\begin{bmatrix} \mu & \cdots & \mu \end{bmatrix}$, for some scalar μ, then the invertible matrix Q is given by $Q = \bar{\mu} P$ and its inverse is $Q^{-1} = \frac{1}{\bar{\mu}} P^{\dagger}$.*

5.2.2 Eigenvalues and eigenvectors of Cayley graphs over $\mathbb{Z}_r^n$

Consider the Cayley graphs over $\mathbb{Z}_r^n$, where r and n are positive integers. The following result shows that this graph family has the same set of eigenvectors.

Theorem 16. *Let r and n be positive integers, and put $N = r^n$. Define an $N \times N$ matrix P by*

$$P_{i,j} = \frac{1}{\sqrt{N}} \omega^{\langle \rho(i) | \rho(j) \rangle}, \qquad 1 \leq i, j \leq N. \tag{5.8}$$

where, $\omega = e^{\frac{2\pi\iota}{r}}$ is the r-th root of unity, and $\rho(i)$ is the i-th element in the lexicographically ordered $\mathbb{Z}_r^n$. Let $Z \in \mathbb{R}^{N\times 1}$ such that $Z_{h,1} = Z_{l,1}$, $1 \leq h, l \leq N$, whenever $\rho(h) = -\rho(l)$. Then, the associated matrix A, appearing in Theorem 15, is real and symmetric with real eigenvalues $X = \sqrt{N} P^{\dagger} Z$.

Proof. The matrix P is unitary, viz.,

$$\begin{aligned} (PP^{\dagger})_{i,j} &= \sum_{k=1}^{N} P_{i,k} P^{\dagger}_{k,j} = \sum_{k=1}^{N} P_{i,k} \bar{P}_{j,k} \\ &= \frac{1}{N} \sum_{k=1}^{N} \omega^{\langle \rho(i) | \rho(k) \rangle} \omega^{-\langle \rho(j) | \rho(k) \rangle} \\ &= \frac{1}{\sqrt{N}} \sum_{k=1}^{N} \omega^{\langle \rho(i) - \rho(j) | \rho(k) \rangle} \end{aligned} \tag{5.9}$$

Recall that the r-th root of unity ω satisfies

$$1 + \omega^s + \omega^2 s + \cdots + \omega^{(r-1)s} = 0, \tag{5.10}$$

where s is a fixed non-zero integer. Thus, for a fixed n-tuple $(x_1, x_2, \ldots, x_n) \in \mathbb{Z}_r^n$, we get

$$\frac{1}{N} \sum_{y \in \mathbb{Z}_r^n} \omega^{\langle x | y \rangle} = \begin{cases} 0 & x = \mathbf{0} \\ 1 & x \neq \mathbf{0}. \end{cases} \tag{5.11}$$

Using this fact, we get

$$(PP^{\dagger})_{i,j} = \begin{cases} 1 & \rho(i) = \rho(j) \\ 0 & \rho(i) \neq \rho(j) \end{cases} \tag{5.12}$$

Thus, $PP^\dagger = I$ is the identity matrix. Since, the value of $\langle \rho(1)|\rho(j)\rangle$ is zero, the first row of P equals $\frac{1}{\sqrt{N}}\begin{bmatrix}1 & 1 & \cdots & 1\end{bmatrix}$. Thus, by Corollary 3, the invertible matrix Q is given by $\frac{1}{\sqrt{N}}P$, which implies $Q^{-1} = \sqrt{N}P^\dagger$. Given a column matrix $Z \in \mathbb{R}^{N\times 1}$, the associated matrix A, see Theorem 15, has eigenvalues $X = Q^{-1}Z = \sqrt{N}P^\dagger Z$. Thus, the $(k,1)$-th entry of X is given by

$$X_{k,1} = \sum_{l=1}^{N} Z_{l,1}\omega^{-\langle \rho(l)|\rho(k)\rangle}. \tag{5.13}$$

Substituting the value of $X_{k,1}$ and P in Eq. 5.6, the (i,j)-entry of A is computed as

$$\begin{aligned} A_{i,j} &= \sum_{k=1}^{N} X_{k,1}P_{i,k}\bar{P}_{j,k} \\ &= \frac{1}{N}\sum_{k=1}^{N}\Big(\sum_{l=1}^{N} Z_{l,1}\omega^{-\langle \rho(l)|\rho(k)\rangle}\Big)\omega^{\langle \rho(i)|\rho(k)\rangle}\omega^{-\langle \rho(j)|\rho(k)\rangle} \\ &= \frac{1}{N}\sum_{l=1}^{N} Z_{l,1}\sum_{k=1}^{N}\omega^{\langle \rho(i)-\rho(j)-\rho(l)|\rho(k)\rangle} \\ \implies A_{i,j} &= Z_{l,1}, \qquad \text{where } \rho(l) = \rho(i)-\rho(j). \end{aligned} \tag{5.14}$$

Since, $\rho(h) = -\rho(l)$ implies $Z_{h,1} = Z_{l,1}$, we get $A_{i,j} = A_{j,i}$. Thus, A is real and symmetric. Therefore, the eigenvalues of A must be real, viz., for $1 \le k \le N$,

$$\begin{aligned} X_{k,1} &= \sum_{l=1}^{N} Z_{l,1}\omega^{-\langle \rho(l)|\rho(k)\rangle} \\ &= \sum_{\substack{1\le l\le N \\ \rho(l)\neq-\rho(l)}} Z_{l,1}\Big(\omega^{-\langle \rho(l)|\rho(k)\rangle} + \omega^{\langle \rho(l)|\rho(k)\rangle}\Big) \\ &\quad + \sum_{\substack{1\le l\le N \\ \rho(l)=-\rho(l)}} Z_{l,1}\omega^{-\langle \rho(l)|\rho(k)\rangle}. \end{aligned} \tag{5.15}$$

We know that sum of a complex number with its complex-conjugate is real, i.e., $\omega^{-\langle \rho(l)|\rho(k)\rangle} + \omega^{\langle \rho(l)|\rho(k)\rangle} = 2\cos(\frac{2\pi}{r}\langle \rho(l)|\rho(k)\rangle)$ is real. Secondly, if a complex number is equal to its complex-conjugate then it is real, i.e., $\rho(l) = -\rho(l)$ implies $\omega^{-\langle \rho(l)|\rho(k)\rangle}$ is real. Notice that, if $x = -x$, $x \in \mathbb{Z}_r^n$, then $x_i = 0$ or $\frac{r}{2}$, for all

$1 \leq i \leq n$. Let $\widetilde{x}$ denote the n-tuple in $\mathbb{Z}_2^n$ with $\widetilde{x}_i = x_i \bmod 2$. Then, we get

$$\omega^{-\langle\rho(l)|\rho(k)\rangle} = e^{-\frac{2\pi}{r}\frac{r}{2}\langle\widetilde{\rho(l)}|\rho(k)\rangle} = \cos\Big(\pi\,\langle\widetilde{\rho(l)}|\rho(k)\rangle\Big). \tag{5.16}$$

Thus, the k-th eigenvalue is real, given by

$$\begin{aligned} X_{k,1} = & \sum_{\substack{1\leq l\leq N\\ \rho(l)\neq-\rho(l)}} Z_{l,1}\times 2\cos\Big(\frac{2\pi}{r}\,\langle\rho(l)|\rho(k)\rangle\Big) \\ & + \sum_{\substack{1\leq l\leq N\\ \rho(l)=-\rho(l)}} Z_{h,1}\times\cos\Big(\pi\,\langle\widetilde{\rho(l)}|\rho(k)\rangle\Big). \end{aligned} \tag{5.17}$$

□

Remark 4. *The real-symmetric matrix A obtained in Theorem 16 is the adjacency matrix of a weighted Cayley graph Cay$(\mathbb{Z}_r^n, \Omega)$, where*

$$\Omega = \{\rho(l) : 1 \leq l \leq N,\ Z_{l,1} \neq 0\},$$

and weight of an edge (u, v) is given by $Z_{h,1}$, where $\rho(h) = u - v$. The underlying unweighted Cayley graph is present as a subgraph containing all vertices and edges such that edges have weights 1. If entries in Z belong to $\{0, 1\}$, then the associated Cayley graph is unweighted. If $r = 2$, then the associated matrix A is the adjacency matrix of a weighted cubelike graph.

Remark 5. *Notice that the inner product $\langle x|y\rangle$ is an element of $\mathbb{Z}$, so one need to be careful while using modulo r with the inner product. Suppose $\langle x|y\rangle = pr + q$, with $0 \leq q < r$. Then,*

$$\begin{aligned} \omega^{\langle x|y\rangle} &= e^{\frac{2\pi\imath}{r}(pr+q)} = e^{2\pi\imath(1+q/r)} \\ &= e^{2\pi}e^{\frac{2\pi\imath}{r}} = e^{\frac{2\pi\imath q}{r}} \\ &= e^{\frac{2\pi\imath}{r}\langle x|y\rangle \ mod\ r} = \omega^{\langle x|y\rangle \ mod\ r}. \end{aligned} \tag{5.18}$$

Remark 6. *In Theorem 16;*

1. *Since, the entries of A depends solely on Z, if $Z \in \mathbb{Z}_2^{N\times 1}$ then A is a $0 - 1$ matrix and thus corresponds to an unweighted Cayley graphs over $\mathbb{Z}_r^n$.*

2. *Suppose r is not even, then no element of $\mathbb{Z}_r^n$, except for $\mathbf{0}$, is its own inverse. In this case, the k-th eigenvalue is given by*

$$X_{k,1} = \sum_{\substack{1 \leq l \leq N \\ \rho(l) \neq -\rho(l)}} Z_{l,1} \times 2\cos\left(\frac{2\pi}{r} \langle \rho(l) | \rho(k) \rangle \right). \tag{5.19}$$

   Moreover, each eigenvalue, except for $X_{1,1}$, occurs in pair because if $\rho(k) = -\rho(s)$ for $k \neq s$, then $\cos(\frac{2\pi}{r} \langle \rho(l) | \rho(k) \rangle) = \cos(\frac{2\pi}{r} \langle \rho(l) | \rho(s) \rangle)$ for all l. This implies $X_{k,1} = X_{s,1}$.

3. *If $r = 2$, then each n-tuple $x \in \mathbb{Z}_2^n$ is its own inverse. In this case, the k-th eigenvalue is given by*

$$X_{k,1} = \sum_{l=1}^{N} Z_{l,1} \times \cos\left(\pi \langle \rho(l) | \rho(k) \rangle \right). \tag{5.20}$$

   Since the cos function has values 0 or 1 or -1, if Z has integer entries then all eigenvalues are integers.

Suppose r is prime, then with further restrictions over Z, in Theorem 16, we obtain integral eigenvalues for the associated adjacency matrix A. Before our next result, we study an equivalence relation over $\mathbb{Z}_r^n$. Define a relation $\sim$ on $\mathbb{Z}_r^n$ by;

$$u \sim v \text{ if } v = iu, \text{ for some } 1 \leq i \leq r-1.$$

The relation $\sim$ is reflexive and transitive, but may not be symmetric. For example, if $r = 4$ and $n = 2$ then $(0,2) = 2(0,1)$ but for no power i we get $(0,1) = i(0,2)$.

Lemma 10. *The relation $\sim$ defined on $\mathbb{Z}_r^n$ is an equivalence relation only if r is prime.*

Proof. The additive identity $\mathbf{0}$ is related to itself alone. Let $u, v, w \in \mathbb{Z}_r^n$ be three non-zero n-tuples. Then, $u = 1u$ implies $\sim$ is reflexive. If $v = iu$ and $w = jv$ for some $1 \leq i, j \leq r-1$, then $w = (ij \bmod r)u$ with $1 \leq ij \bmod r \leq r-1$, which implies $\sim$ is transitive. Since r is prime, there is unique integer k, $1 \leq k \leq r-1$, such that $ik \bmod r = 1$. Thus, $kv = (ik \bmod r)u = u$, and the relation $\sim$ is symmetric. Hence, $\sim$ is an equivalence relation on $\mathbb{Z}_r^n$. □

Lemma 11. *Let r be a prime, and $\mathcal{L}$ be the equivalence classes of the equivalence relation $\sim$ on $\mathbb{Z}_r^n$. Let $L_u \in \mathcal{L}$ be an equivalence class with a non-zero representative u, then*

1. *L has $r-1$ elements,*
2. *the set $\{\langle x|v\rangle \mod r : v \in L_u\}$, $x \neq \mathbf{0}$, is either equal to $\{1, 2, \ldots, r-1\}$ or $\{0\}$.*

Proof. Let $u = (u_1, \ldots, u_n)$, then $L_u = \{(iu_1, \ldots, iu_n) : 1 \leq i \leq r-1\}$. Since the group operation occurs component-wise and at least one coordinate of u is non-zero, u generates exactly $r-1$ distinct element because $u^r = \mathbf{0}$. Next, for any non-zero element $x = (x_1, \ldots, x_n) \in \mathbb{Z}_r^n$,

$$\langle x|iu\rangle = \sum_{j=1}^{n} x_j(iu_j) = i\left(\sum_{j=1}^{n} x_j u_j\right) = i\,\langle x|u\rangle\,.$$

Thus, $\langle x|iu\rangle \mod r = 0$ iff $\langle x|u\rangle \mod r = 0$, so if $\langle x|u\rangle \mod\ \neq 0$ then it generates the group $\mathbb{Z}_r$. □

Example 6. *Let $r = 3$ and $n = 2$. Define the equivalence relation $\sim$ on $\mathbb{Z}_3^2$ by*

$$u \sim v \text{ if } v = iu, \text{ for some } i, \text{ with } 1 \leq i \leq 2.$$

Then, the equivalence classes on $\mathbb{Z}_3^2$ relative to $\sim$ are $\{(0,0)\}$, $\{(0,1),(0,2)\}$, $\{(1,0),(2,0)\}$, $\{(1,1),(2,2)\}$ and $\{(1,2),(2,1)\}$.

Theorem 17. *In Theorem 16, assume r is prime. Let $\sim$ be the relation on $\mathbb{Z}_r^n$ defined by; $u \sim v$ iff $v = iu$ for some $1 \leq i \leq r-1$. Let $Z \in \mathbb{R}^{N\times 1}$ such that $Z_{g,1} = Z_{h,1}$ whenever $\rho(g) \sim \rho(h)$. Then,*

1. *the matrix A associated with Z is real and symmetric with real eigenvalues $X = \sqrt{N}P^\dagger Z$,*
2. *if Z has integer entries then entries in A are integers and corresponding eigenvalues are integers.*

Proof. Let R be the set of representatives of equivalence classes of the relation $\sim$

such that if $x, y \in R$ then $x \not\sim y$. Suppose A is the matrix associated with Z, then

$$\begin{aligned}
X_{k,1} &= \sqrt{N} \sum_{l=1}^{N} P^{\dagger}_{k,l} Z_{l,1} \\
&= \sum_{l=1}^{N} Z_{l,1} \omega^{\langle \rho(l) | \rho(k) \rangle} \\
&= \sum_{\rho(h) \in R} Z_{h,1} \left(\sum_{\rho(h) \sim \rho(l)} \omega^{\langle \rho(l) | \rho(k) \rangle} \right) \\
&= \sum_{\substack{\rho(h) \in R \\ \langle \rho(h) | \rho(k) \rangle = 0}} (r-1) Z_{h,1} + \sum_{\substack{\rho(h) \in R \\ \langle \rho(h) | \rho(k) \rangle \neq 0}} -Z_{h,1}.
\end{aligned} \tag{5.21}$$

Thus, X has real (or integer) entries if Z has real (or integer) entries. Since $\rho(h) = -\rho(l)$ implies $\rho(h) \sim \rho(l)$, the (i, j)-th entry of A is computed same as in Theorem 16. Thus, A is real and symmetric, and has integer entries if Z has integer entries.

□

Example 7. *Let $r = 3$, $n = 2$, and let $\sim$ be the equivalence relation defined in Example 6. From Theorem 16, the 9×9 unitary matrix P is given by*

$$P = \frac{1}{3} \times \begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & \omega & \omega^2 & 1 & \omega & \omega^2 & 1 & \omega & \omega^2 \\
1 & \omega^2 & \omega & 1 & \omega^2 & \omega & 1 & \omega^2 & \omega \\
1 & 1 & 1 & \omega & \omega & \omega & \omega^2 & \omega^2 & \omega^2 \\
1 & \omega & \omega^2 & \omega & \omega^2 & 1 & \omega^2 & 1 & \omega \\
1 & \omega^2 & \omega & \omega & 1 & \omega^2 & \omega^2 & \omega & 1 \\
1 & 1 & 1 & \omega^2 & \omega^2 & \omega^2 & \omega & \omega & \omega \\
1 & \omega & \omega^2 & \omega^2 & 1 & \omega & \omega & \omega^2 & 1 \\
1 & \omega^2 & \omega & \omega^2 & \omega & 1 & \omega & 1 & \omega^2
\end{bmatrix}.$$

We have assumed the lexicographic order of elements in $\mathbb{Z}_3^2$, i.e.,

$$\mathbb{Z}_3^2 = \{(0,0), (0,1), (0,2), (1,0), (1,1), (1,2), (2,0), (2,1), (2,2)\}.$$

Suppose $Z \in \mathbb{R}^{9\times 1}$ is a column matrix satisfying

$$Z_{1,1} = a,\ Z_{2,1} = Z_{3,1} = b,\ Z_{4,1} = Z_{7,1} = c,\ Z_{5,1} = Z_{9,1} = d,\ Z_{6,1} = Z_{8,1} = e,$$

for some real a, b, c and d. Then, the associated matrix A has eigenvalues $X = 3P^{\dagger}Z$, i.e.,

$$X = \begin{bmatrix} a + 2b + 2c + 2d + 2e \\ a - b + 2c - d - e \\ a - b + 2c - d - e \\ a + 2b - c - d - e \\ a - b - c - d + 2e \\ a - b - c + 2d - e \\ a + 2b - c - d - e \\ a - b - c + 2d - e \\ a - b - c - d + 2e \end{bmatrix}. \tag{5.22}$$

We have used the equation $1 + \omega + \omega^2 = 0$ to evaluate the value of X. Notice that, if the values of a, b, c, d, and e are integers, then all eigenvalues are integers. So, we have computed the eigenvalues for the adjacency matrix A. In other words, we have constructed the graph whose eigenvectors form columns of P by assigning weights to edges adjacent to the first vertex.

Corollary 4. *In Theorem 17,*

1. *If $\rho(k) \sim \rho(s)$ then $X_{k,1} = X_{s,1}$.*
2. *If $r > 2$ and entries in Z are integers, then*

   (a) $X_{1,1}$ is even, viz.,

$$X_{1,1} = (r-1) \times \sum_{\rho(h)\in R} Z_{h,1}, \text{ since } \forall h, \langle \rho(h) | \rho(1) \rangle = 0. \tag{5.23}$$

(b) The difference of an eigenvalue with the first eigenvalue is a multiple of r, viz.,

$$X_{1,1} - X_{k,1} = r \times \sum_{\substack{\rho(h) \in R \\ \langle \rho(h) | \rho(k) \rangle \neq 0}} Z_{h,1}. \tag{5.24}$$

3. *If $r = 2$ and entries in Z are integers, then all eigenvalues have same parity.*

Proof. 1. Since $A_{i,j} = Z_{l,1}$, where $\rho(l) = \rho(i) - \rho(j)$, the result follows.

2. If $\rho(k) \sim \rho(s)$, then $\langle \rho(h) | \rho(k) \rangle = 0$ iff $\langle \rho(h) | \rho(s) \rangle = 0$. Thus, $X_{k,1} = X_{s,1}$, whenever $\rho(k) \sim \rho(s)$. In Example 7, we see that $X_{2,1} = X_{3,1}$, $X_{4,1} = X_{7,1}$, $X_{5,1} = X_{9,1}$ and $X_{6,1} = X_{8,1}$.

3. Using Eq. 5.21, we get the required expression.

4. Substituting $r = 2$ in Eq. 5.24, we see that $X_{1,1} - X_{k,1}$ is even for all k. Thus, all eigenvalues have same eigenvalues.

□

Remarks

The study of normal matrices in Theorem 15 has applications to generating graphs having the same eigenvectors. This can be done by assigning values to N selected pairs of vertices and calculating the eigenvalues of the corresponding graph without actually computing its adjacency matrix. We looked for a way to generate undirected connected graphs and studied the family of Cayley graphs over $\mathbb{Z}_r^n$ in Theorem 16. Notice that, if we lessen the conditions on Z, i.e. $Z \in \mathbb{C}^{N \times 1}$, then the corresponding graphs are complex-weighted Cayley graphs over $\mathbb{Z}_r^n$.

5.3 Continuous-time quantum walks on Cayley graphs over $\mathbb{Z}_r^n$

In this section, we study the CTQW on Cayley graphs over $\mathbb{Z}_r^n$. Suppose perfect state transfer occurs between vertices a and b with time τ, then the (a, b)-th entry of the transition matrix $\mathcal{U}(\tau)$ has absolute value 1, i.e.,

$$\begin{aligned} |\mathcal{U}(\tau)_{a,b}| = \left| \sum_{k=1}^{N} e^{\iota\tau X_{k,1}} P_{a,k} \bar{P}_{b,k} \right| &= 1 \\ \implies \left| \frac{1}{N} \sum_{k=1}^{N} e^{\iota\tau X_{k,1}} e^{\frac{\iota 2\pi}{r}\langle a|\rho(k)\rangle} e^{-\frac{\iota 2\pi}{r}\langle b|\rho(k)\rangle} \right| &= 1 \\ \implies \left| \sum_{k=1}^{N} e^{\iota\tau X_{k,1} + \frac{\iota 2\pi}{r}\langle a-b|\rho(k)\rangle} \right| &= N \\ \implies \left| \sum_{k=1}^{N} e^{\iota\tau X_{k,1} + \frac{\iota 2\pi}{r}\langle \sigma|\rho(k)\rangle} \right| &= N, \qquad \text{where } \sigma = a - b. \end{aligned} \tag{5.25}$$

The absolute value of the summation is equal to N only if each term is equal to one another, i.e.,

$$\begin{aligned} e^{\iota\tau X_{k,1} + \frac{\iota 2\pi}{r}\langle \sigma|\rho(k)\rangle} &= e^{\iota\tau X_{1,1}} \\ \implies e^{\iota\left[\tau\left(X_{k,1} - X_{1,1}\right) + \frac{2\pi}{r}\langle\sigma|\rho(k)\rangle\right]} &= 1. \\ \implies \tau\left(X_{k,1} - X_{1,1}\right) + \frac{2\pi}{r}\langle\sigma|\rho(k)\rangle &= 2\pi m_k, \quad m_k \in \mathbb{Z} \\ \implies \frac{\tau}{2\pi} \times r(X_{k,1} - X_{1,1}) + \langle\sigma|\rho(k)\rangle &= r m_k, \quad m_k \in \mathbb{Z}. \end{aligned} \tag{5.26}$$

In general, eigenvalues are not integers, and it is not straightforward to check if PST or periodicity occur in these graphs. If all eigenvalues are integers, then with time $\tau = 2\pi$ and $\sigma = \mathbf{0}$, we get

$$r(X_{k,1} - X_{1,1}) \bmod r = 0. \tag{5.27}$$

This implies that if all eigenvalues are integers then the graph is periodic with period dividing 2π.

5.3.1 Perfect state transfer in weighted cubelike graphs

Suppose $r = 2$, then the associated graphs are weighted cubelike graphs. Recall that, $X_{k,1} - X_{1,1}$ is even for all k, see Corollary 4. Thus, by substituting $\sigma = \mathbf{0}$ and

$\tau = \pi$ in Eq. 5.25, we get

$$(X_{k,1} - X_{1,1}) \bmod 2 = 0. \tag{5.28}$$

This implies weighted cubelikes graphs are periodic with period $\tau = \pi$.

Theorem 18. *Suppose $r = 2$ in Theorem 16 and let $Z \in \mathbb{Z}^{N\times 1}$. Define*

$$\sigma = \sum_{Z_{l,1}\neq 0} Z_{l,1}\rho(l). \tag{5.29}$$

1. *If $\sigma = \mathbf{0}$, then the associated weighted cubelike graph is periodic with period $\tau = \frac{\pi}{2}$.*
2. *If $\sigma \neq \mathbf{0}$, then the associated weighted cubelike graph admits perfect state transfer between vertices a and b, satisfying $\sigma = a - b$, with time $\tau = \frac{\pi}{2}$. The graph is periodic with period dividing π.*

Proof. Using Eq. 5.20, we compute the difference of eigenvalues as

$$X_{k,1} - X_{1,1} = \sum_{l=1}^{N} Z_{l,1}\Big(\cos(\pi \langle \rho(l)|\rho(k)\rangle) - 1\Big). \tag{5.30}$$

Perfect state transfer occurs between a and b with time $\tau = \frac{\pi}{2}$ if the solution for the following equation exist for all k, i.e.,

$$\Big(\frac{X_{k,1} - X_{1,1}}{2} + \langle \sigma|\rho(k)\rangle\Big) \bmod 2 = 0, \qquad \forall k. \tag{5.31}$$

Without loss of generality, see Remark 5, assume that all inner products and additions are taken under modulo 2. Thus, for $1 \leq k \leq N$, we compute σ as

$$\begin{aligned}
\langle \sigma | \rho(k) \rangle &= \sum_{l=1}^{N} Z_{l,1}\Big(1 - \cos(\pi \langle \rho(l) | \rho(k) \rangle)\Big) \bmod 2 \\
&= \sum_{\substack{Z_{l,1} \neq 0 \\ \langle \rho(l) | \rho(k) \rangle \neq 0}} Z_{l,1}\Big(1 - \cos(\pi \langle \rho(l) | \rho(k) \rangle)\Big) \bmod 2 \\
&= \sum_{\substack{Z_{l,1} \neq 0 \\ \langle \rho(l) | \rho(k) \rangle \bmod 2 = 1}} Z_{l,1} \bmod 2 \\
&= \sum_{Z_{l,1} \neq 0} Z_{l,1} \langle \rho(l) | \rho(k) \rangle \bmod 2 \\
\implies \langle \sigma | \rho(k) \rangle &= \langle \sum_{Z_{l,1} \neq 0} Z_{l,1} \rho(l) | \rho(k) \rangle \bmod 2.
\end{aligned} \tag{5.32}$$

Thus, $\sigma = \sum_{Z_{l,1} \neq 0} Z_{l,1}\rho(l)$. If $\sigma = \mathbf{0}$, then the graph is periodic with period dividing $\tau = \frac{\pi}{2}$, otherwise the graph admits PST between a and b with time $\tau = \frac{\pi}{2}$. □

Example 8. *In Theorem 18, let $n = 3$ and $Z = \begin{bmatrix} 0 & 1 & 2 & 3 & 4 & 1 & 2 & 3 \end{bmatrix}^T$. Then, σ is computed as*

$$\begin{aligned}
\sigma &= 0 \times (0,0,0) + 1 \times (0,0,1) + 2 \times (0,1,0) + 3 \times (0,1,1) \\
&\quad + 4 \times (1,0,0) + 1 \times (1,0,1) + 2 \times (1,1,0) + 3 \times (1,1,1) \\
&= (0,0,0).
\end{aligned} \tag{5.33}$$

By Theorem 18, the graph associated with Z is periodic with period $\tau = \frac{\pi}{2}$, with time $\frac{\pi}{2}$. Suppose we alter the values Z as $Z_{2,1} = 2$ and $Z_{5,1} = 3$. Then, the newly computed σ is

$$\sigma = (1,0,1).$$

In this case, PST occurs between $\{a, b\}$ if $\sigma = a - b$. Thus, PST pairs are

$$\{\{(0,0,0),(1,0,1)\},\{(0,0,1),(1,0,0)\},\{(0,1,0),(1,1,1)\},\{(0,1,1),(1,1,0)\}\}.$$

Remark 7. *In Theorem 18, the associated graph is weighted cubelike graph $Cay(\mathbb{Z}_2^n, \Omega)$,* *where $\rho(l) \in \Omega$ iff $Z_{l,1} \neq 0$, $1 \leq l \leq N$. An alternate definition of σ is given by*

$$\sigma = \sum_{x \in \Omega} x f(x), \tag{5.34}$$

where, f is the weight function defined by $f(\rho(l)) = Z_{l,1}$. Since, σ belongs to $\mathbb{Z}_2^n$, i.e., each entry σ_i is taken under modulo 2, we can re-write the expression for σ as

$$\sigma = \sum_{Z_{l,1} \ mod \ 2 \neq 0} Z_{l,1} \rho(l). \tag{5.35}$$

5.3.2 Periodicity in Cayley graphs over $\mathbb{Z}_r^n$

Suppose r is prime, then the Cayley graphs over $\mathbb{Z}_r^n$ have integral eigenvalues, by Theorem 17, and thus are periodic graphs [40]. We discuss the period of CTQW on this graph in the following result.

Theorem 19. *With the assumptions made in Theorem 17; let Z contains integer values. Then, the associated weighted Cayley graphs over $\mathbb{Z}_r^n$, where r is prime, is periodic with period dividing $\frac{2\pi}{r}$. However, these graphs do not admit PST for $r > 2$.*

Proof. Since $(X_{k,1} - X_{1,1})$ is a multiple of r (by Corollary 4), substituting $\sigma = \mathbf{0}$ and $\tau = \frac{2\pi}{r}$ in Eq. 5.26, we get

$$(X_{k,1} - X_{1,1}) \bmod r = 0. \tag{5.36}$$

Thus, Cayley graphs over $\mathbb{Z}_r^n$, where r is prime, are periodic with period dividing $\tau = \frac{2\pi}{r}$. Next, assume PST occurs between distinct vertices a and b. Let $\rho(k) \sim \rho(s)$ for $k \neq s$ such that $\langle \sigma | \rho(k) \rangle \neq 0$, where $\sigma = a - b$. By Lemma 11, if $\sigma \neq \mathbf{0}$ then $\langle \sigma | \rho(k) \rangle = i \langle \sigma | \rho(s) \rangle$ for some $1 < i < r$. Substituting this expression in Eq. 5.26, we get

$$\begin{aligned} \left(\frac{\tau}{2\pi} \times r(X_{k,1} - X_{1,1}) + \langle \sigma | \rho(k) \rangle \right) \bmod r = 0 \\ \left(\frac{\tau}{2\pi} \times r(X_{s,1} - X_{1,1}) + i \langle \sigma | \rho(k) \rangle \right) \bmod r = 0. \end{aligned} \tag{5.37}$$

Since $X_{k,1} = X_{s,1}$, by Corollary 4, all terms are same except for the factor i. Thus, two equations are not consistent and hence PST can not occur in such graphs for any value of τ and σ. □

Example 9. *In Example 7, suppose* $Z = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}^T$*, i.e.,* $a = 0$*,* $b = 1$ $c = 0$*,* $d = 1$ *and* $e = 0$*. Thus, set of eigenvalues, obtained using Eq. 5.22, is* $X = \begin{bmatrix} 4 & -2 & -2 & 1 & -2 & 1 & 1 & 1 & -2 \end{bmatrix}$*. We see that* $X_{2,1} - X_{1,1} = -6$*,* $X_{4,1} - X_{1,1} = -3$*,* $X_{5,1} - X_{1,1} = -3$ *and* $X_{6,1} - X_{1,1} = -3$*. Thus, the Cayley graph associated with Z is periodic with period* $\tau = \frac{2\pi}{3}$*.*

Remarks

We studied the case where r is prime in Theorem 16 and 17. For other values of r, i.e., r is not prime, eigenvalues may not be integral. If we show that eigenvalues are not a multiple of the square root of a square-free integer, then the corresponding graph is not periodic [40]. For the existence of PST, notice that in Eq. 5.25 even if for some value of τ the first term $\frac{\tau r}{2\pi}(X_{k,1} - X_{1,1})$ is integer for all k then the existence of σ satisfying

$$\langle \sigma | \rho(k) \rangle = \frac{\tau r}{2\pi}(X_{k,1} - X_{1,1}) \bmod r, \qquad \forall k, \tag{5.38}$$

may not hold true, in general. If such σ exists for some $Z \in \mathbb{R}^{N \times 1}$ then the corresponding graph admits PST and the vertex-set is partitioned into PST pairs, i.e., $\{\{a, b\} : \sigma = a - b\}$. Clearly, if r is odd then the size of the vertex-set is odd and they can not be partitioned into PST pairs.

5.4 Identifying pair of vertices in multipartite graphs that admit perfect state transfer

The n-dimensional hypercube is a bipartite graph and a Cayley graph is a multipartite graph. The minimum partition of a Cayley has each part of same size. We discuss another subfamily of multipartite graphs that admit perfect state transfer or periodicity and study properties of graphs that may be helpful in determining a pair of vertices between which PST occurs.

Theorem 20. *The complete bipartite graphs that admit perfect state transfer are $K_{1,1}$ and $K_{2,n}$ for $n \geq 1$. The star $K_{1,n}$ is periodic for $n \geq 3$.*

In [41], Godsil states that if a graph G admits perfect state transfer from a vertex u to a vertex v, then any automorphism that fixes u must fix v. With this fact, one can derive the results of Theorem 20. We, on the other hand, present an alternate proof to Theorem 20 by the use of spectral decomposition of the corresponding adjacency matrix.

Proof of Theorem 20. Assume that if $\{X, Y\}$ is the partition of a complete bipartite graph $K_{m,n}$, then the vertices in X are labeled by $x_1, \ldots, x_m$ and the vertices in Y are labeled by $y_1, \ldots, y_n$. The adjacency matrix A of $K_{m,n}$ is of the form

$$\begin{bmatrix} O_1 & I_1 \\ I_2 & O_2 \end{bmatrix}$$

where O_1 is $m \times m$ zero-matrix, O_2 is $n \times n$ zero-matrix, I_1 is $m \times n$ matrix of ones and I_2 is $n \times m$ matrix of ones. The eigenvalues of A are $\{\theta_1 = -\sqrt{mn}, \theta_2 = 0, \theta_3 = \sqrt{mn}\}$, where θ_2 has multiplicity $m + n - 2$. The eigenvector corresponding to the eigenvalue θ_1 is

$$v_1 = \frac{1}{n\sqrt{2m}} \begin{bmatrix} n \\ \vdots \\ n \\ -\sqrt{mn} \\ \vdots \\ -\sqrt{mn} \end{bmatrix},$$

the eigenvector corresponding to θ_3 is

$$v_{m+n} = \frac{1}{n\sqrt{2m}} \begin{bmatrix} n \\ \vdots \\ n \\ \sqrt{mn} \\ \vdots \\ \sqrt{mn} \end{bmatrix},$$

and the eigenvectors corresponding to θ_2 are

$$v_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \\ \vdots \\ 0 \\ - \\ 0 \\ \vdots \end{bmatrix}, \; v_3 = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 \\ 1 \\ -1 \\ \vdots \\ 0 \\ - \\ 0 \\ \vdots \end{bmatrix}, \cdots, v_m = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 \\ \vdots \\ 1 \\ -1 \\ - \\ 0 \\ \vdots \end{bmatrix},$$

$$v_{m+1} = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ - \\ 1 \\ -1 \\ \vdots \end{bmatrix}, \cdots, v_{m+n-1} = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ - \\ 0 \\ \vdots \\ 1 \\ -1 \end{bmatrix}.$$

Note that the the m^{th} row of v_m is -1. The orthogonal projection on the eigenspace belonging to θ_1 and θ_3 are,

$$E_1 = \frac{1}{2mn^2}\begin{bmatrix} n^2 & \cdots & n^2 & -n\sqrt{mn} & \cdots & -n\sqrt{mn} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ n^2 & \cdots & n^2 & -n\sqrt{mn} & \cdots & -n\sqrt{mn} \\ -n\sqrt{mn} & \cdots & -n\sqrt{mn} & mn & \cdots & mn \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ -n\sqrt{mn} & \cdots & -n\sqrt{mn} & mn & \cdots & mn \end{bmatrix},$$

and

$$E_3 = \frac{1}{2mn^2}\begin{bmatrix} n^2 & \cdots & n^2 & n\sqrt{mn} & \cdots & n\sqrt{mn} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ n^2 & \cdots & n^2 & n\sqrt{mn} & \cdots & n\sqrt{mn} \\ n\sqrt{mn} & \cdots & n\sqrt{mn} & mn & \cdots & mn \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ n\sqrt{mn} & \cdots & n\sqrt{mn} & mn & \cdots & mn \end{bmatrix}.$$

The orthogonal projection on the eigenspace belonging to $\theta_2 = 0$ is

$$E_2 = \frac{1}{2}\left[\begin{array}{cccccc|cccccc} 1 & -1 & & & & & & & & & & \\ -1 & 2 & -1 & & & & & & & & & \\ & -1 & 2 & -1 & & & & & \mathbf{0} & & & \\ & & \ddots & & & & & & & & & \\ & & -1 & 2 & -1 & & & & & & & \\ & & & -1 & 1 & & & & & & & \\ \hline & & & & & & 1 & -1 & & & & \\ & & & & & & -1 & 2 & -1 & & & \\ & & \mathbf{0} & & & & & & \ddots & & & \\ & & & & & & & & -1 & 2 & -1 & \\ & & & & & & & & & -1 & 1 & \end{array}\right]$$

Let X, Y, and Z be the $m \times m$, $m \times n$, and $n \times m$ submatrices of E_3 satisyfing

$$E_3 = \frac{1}{2mn^2} \begin{bmatrix} X & Y \\ Y & Z \end{bmatrix}.$$

Then, E_1 can be rewritten as

$$E_1 = \frac{1}{2mn^2} \begin{bmatrix} X & -Y \\ -Y & Z \end{bmatrix}.$$

Let P and Q be the $m \times m$ and $n \times n$ submatrices of E_2 satisfying

$$E_2 = \frac{1}{2} \begin{bmatrix} P & 0 \\ 0 & Q \end{bmatrix}.$$

The matrix $H(t)$ for a time t is

$$\begin{aligned} H(t) =& e^{-\iota\sqrt{mn}t} E_1 + e^{\iota\sqrt{mn}t} E_3 + \frac{1}{2} E_2 \\ =& \frac{\cos\sqrt{mn}t}{2mn^2} \begin{bmatrix} 2X & 0 \\ 0 & 2Z \end{bmatrix} + \frac{\iota \sin\sqrt{mn}t}{2mn^2} \begin{bmatrix} 0 & 2Y \\ 2Y & 0 \end{bmatrix} + \frac{1}{2} \begin{bmatrix} P & 0 \\ 0 & Q \end{bmatrix} \end{aligned} \tag{5.39}$$

As seen in equation 5.39, if $m = n = 1$, then $K_{1,1}$ admit PST between x_1 and y_1. If $m = 1$ and $n \geq 3$ (or $m \geq 3$ and $n = 1$), then the graph is periodic at x_1 with period $\pi/\sqrt{n}$. For $m = 2$ and $n \geq 1$ (or $m \geq 1$ and $n = 2$), the graph admits PST between the pair $\{x_1, x_2\}$ with time $\pi/\sqrt{2n}$. Since there is no other possible PST pair for $m, n \geq 3$, the result is proved. □

Remarks

To study PST in multipartite graph $K_{n_1,n_2,\ldots,n_k}$ we can adopt the similar approach. Without loss of generality assume $n_1 \leq n_2 \leq \cdots \leq n_k$ and $k \geq 3$, then the adjacency matrix A of $K_{n_1,n_2,\dots,n_k}$ is given by

$$A = \begin{bmatrix} O_1 & J_{12} & \cdots & J_{1k} \\ J_{21} & O_2 & \cdots & J_2k \\ \vdots & \vdots & \cdots & \vdots \\ J_{k1} & Jk2 & \cdots & O_k \end{bmatrix}$$

where, for $1 \leq i, j \leq k$, J_{ij} is $n_i \times n_j$ matrix of ones and O_i is $n_i \times n_i$ zero matrix. Let λ be an eigenvalue of A with an eigenvector X of the form

$$[x_{11}, \dots, x_{1n_1}, x_{21}, \dots, x_{2n_2}, \dots, x_{k1}, \dots, x_{kn_k}]^T,$$

then for $1 \leq l \leq k$, $AX = \lambda X$ gives

$$\sum_{i \neq l} x_{ij} = \lambda x_{ls}; \qquad 1 \leq i \leq k, 1 \leq j \leq n_i, 1 \leq s \leq n_l. \tag{5.40}$$

Hence, we get $x_{l1} = x_{l2} = \cdots = x_{ln_l} = x_l$. Thus, for l with $1 \leq l \leq k$, we get

$$\sum_{i \neq l} x_{ij} = \lambda x_l; \qquad 1 \leq i \leq k, 1 \leq j \leq n_i. \tag{5.41}$$

For $\lambda = 0$, there are $\sum_i (n_i - 1) = n_1 + \dots n_k - k$ eigenvectors which are of the form $V_{ij} = [0, \cdots, 0, 1, -1, 0, \cdots, 0]^T$, with $1 \leq i \leq k,\ 1 \leq j < n_i$, where the value 1 occurs at the row $n_1 + \cdots + n_{i-1} + i$. Thus, the orthogonal projection on the eigenspace belonging to $\lambda = 0$ is of the form

$$E_0 = \begin{bmatrix} P_1 & 0 & \cdots & 0 \\ 0 & P_2 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & P_k \end{bmatrix}$$

where, P_k is $n_k \times n_k$ matrix with entries

$$P_k = \begin{bmatrix} 1 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & & & \\ & & -1 & 2 & -1 & \\ & & & -1 & 1 & \end{bmatrix}$$

It may not be possible to give a general statement about PST in the graph as it depends on particular values of $n_1, n_2, \ldots, n_k$.

5.5 Simulation of continuous-time quantum walks on cubelike graphs

We simulate continuous-time quantum walk on weighted cubelike graphs and verify the existence of perfect state transfer or periodicity as mentioned in Theorem 18. Suppose $f : \mathbb{Z}_2^n \to \mathbb{Z}$ is an integer-valued function defined by; $f(\rho(h)) = Z_{h,1}$, then f is the weight function for the cubelike graph associated with Z such that weight on an edge (x, y) is given by $f(x \oplus y)$. Using these notations, rewrite the Theorem 18 as follows;

Theorem 21. *With the assumptions made in Theorem 18 for $r = 2$, let $f : \mathbb{Z}_2^n \to \mathbb{Z}$ be an integer-valued function such that $f(\rho(h)) = Z_{h,1}$. For $x \in \mathbb{Z}_2^n$, define a subset $O_x = \{y \in \mathbb{Z}_2^n : \langle x|y \rangle \ mod\ 2 = 1\}$. Let e_i, $1 \leq i \leq n$, denote the n-tuple with entry 1 at position i and zero everywhere else. Let $\sigma \in \mathbb{Z}_2^n$ such that*

$$\sigma_i = \sum_{y \in O_{e_i}} f(y) \ mod\ 2. \tag{5.42}$$

Then, the associated weighted cubelike graph satisfies;

1. *if $\sigma = \mathbf{0}$, then the graph is periodic with period $\frac{\pi}{2}$,*

2. *if $\sigma \neq \mathbf{0}$, then PST occurs between every pair $\{u, v\}$ satisfying $u \oplus v = \sigma$, with time $\tau = \frac{\pi}{2}$.*

Proof. Notice that, for $k = 2^{n-i} + 1$, $\rho(k) = e_i$. This implies, $\langle \sigma | \rho(2^{n-i} + 1 \rangle = \sigma_i$. Thus, using Eq. 5.32, we get

$$\begin{aligned} \sigma_i &= \sum_{Z_{l,1} \neq 0} Z_{l,1} \left\langle \rho(l) | \rho(2^{n-i} + 1) \right\rangle \mod 2 \\ &= \sum_{Z_{l,1} \neq 0} Z_{l,1} \rho(l)_i \bmod 2 \\ &= \sum_{\rho(l) \in O_{e_i}} f(\rho(l)) \bmod 2. \end{aligned} \tag{5.43}$$

Notice that $Z_{l,1} \neq 0$ and $\rho(l)_i \neq 0$ implies $\rho(l) \in O_{e_i}$. Thus, Eq. 5.32 and 5.42 are equivalent, and the results follow. □

5.5.1 Decomposition of the evolution operator

Group representations

An m-degree representation of a finite group G is a homomorphism ρ from G into the general linear group $GL(V)$ of an m-dimensional vector space V over the field $\mathbb{F}$, where $\mathbb{F}$ is a complex or real field. Since $GL(V)$ is isomorphic to $GL_m(\mathbb{F})$, the general linear group of degree m that consists of $m \times m$ invertible matrices, an equivalent definition for the group representation is the group homomorphism

$$\rho : G \to GL_m(\mathbb{F}).$$

The group algebra $\mathbb{C}[G]$ is an inner product space whose vectors are formal linear combinations of the group elements, i.e.,

$$\mathbb{C}[G] = \left\{ \sum_{g \in G} \lambda_g g : \lambda_g \in \mathbb{C} \right\},$$

with the vector addition, the scalar multiplication, and the inner product defined by;

$$\sum_{g\in G}\lambda_g g+\sum_{g\in G}\mu_g g=\sum_{g\in G}(\lambda_g+\mu_g)g, \qquad \text{(addition)},$$
$$\lambda\sum_{g\in G}\lambda_g g=\sum_{g\in G}(\lambda\lambda_g)g \qquad \text{(scalar multiplication)},$$
$$\left\langle\sum_{g\in G}\lambda_g g,\sum_{g\in G}\mu_g g\right\rangle=\sum_{g\in G}\lambda_g\bar{\mu}_g, \qquad \text{(inner product)}.$$

The regular representation on G, $\rho_{reg}: G\to GL(\mathbb{C}[G])$, is defined by;

$$\rho_{reg}(x)\left(\sum_{g\in G}\lambda_g g\right)=\sum_{g\in G}\lambda_g(xg)=\sum_{y\in G}\lambda_{x^{-1}y}y.$$

The decomposition

If $G=\mathbb{Z}_2^n$, then for $x\in\mathbb{Z}_2^n$ the regular representation acts on $\mathbb{Z}_2^n$ as

$$\rho_{reg}(x)y=x\oplus y=(x_1\oplus y_1,\ldots,x_n\oplus y_n), \qquad y\in\mathbb{Z}_2^n.$$

Let X, Y and Z denote the three Pauli matrices that acts on the computational basis $\{|0\rangle,|1\rangle\}$ of the two dimensional Hilbert space $\mathbb{C}^2$ as

$$X|a\rangle=|a\oplus 1\rangle,\ Y|a\rangle=(-1)^a\imath|a\oplus 1\rangle,\ Z|a\rangle=(-1)^a|a\rangle,\ a\in\{0,1\}.$$

The group element y is also a vector in $\mathbb{C}[\mathbb{Z}_2^n]$ whose matrix representation is $|y\rangle=|y_1\rangle\otimes\cdots\otimes|y_n\rangle$. Hence, the action of $\rho_{reg}(x)$ over y can be rewritten as

$$\begin{aligned}\rho_{reg}(x)y&=(X^{x_1}|y_1\rangle)\otimes\cdots\otimes(X^{x_n}y_n),\text{ where } X^{x_i}|y_i\rangle=|x_i\oplus y_i\rangle,\\&=(X^{x_1}\otimes\cdots\otimes X^{x_n})(|y_1\rangle\otimes\cdots\otimes|y_n\rangle).\end{aligned}$$

The adjacency matrix A of $Cay(\mathbb{Z}_2^n,f)$ is decomposed by using the regular representation on $\mathbb{Z}_2^n$, viz., given $x,y\in\mathbb{Z}_2^n$, the value $\rho_{reg}(x)y=x\oplus y$ corresponds to

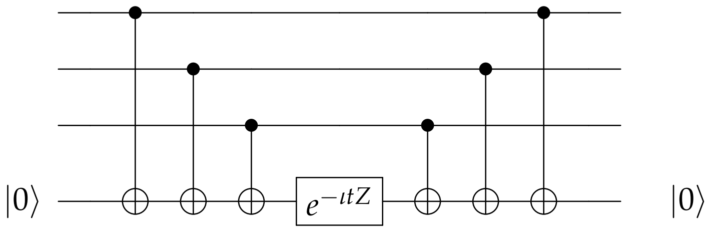


Figure 5.2: A quantum circuit to implement $e^{-\imath tA}$, where $A = Z \otimes Z \otimes Z$.

the (x, y)-entry of A, so A can be expressed as;

$$A = \sum_{x \in \mathbb{Z}_2^n} f(x)\rho_{reg}(x). \tag{5.44}$$

Since $\rho_{reg}(x)$ commutes with $\rho_{reg}(y)$ for all $x, y \in \mathbb{Z}_2^n$, the evolution operator $\mathcal{U}(t) = e^{-\imath tA}$ is decomposed into;

$$\mathcal{U}(t) = \prod_{x \in \mathbb{Z}_2^n} U(x,t), \qquad U(x,t) = e^{-\imath t f(x)\rho_{reg}(x)}. \tag{5.45}$$

5.5.2 Quantum circuits

The idea to design a quantum circuit for CTQW on a cubelike graph has been taken from [59]; if the Hamiltonian is given by $A = Z_1 \otimes \cdots \otimes Z_n$, where $Z_i = Z$, then the phase shift applied to the system is $e^{-\imath t}$ if the parity of the n qubits in the computational basis is even, otherwise, the phase shift applied is $e^{\imath t}$. Fig. 5.2 illustrates the quantum circuit for $e^{-\imath tA}$, where $A = Z \otimes Z \otimes Z$.

Let $x \in \mathbb{Z}_2^n$, then the regular representation $\rho_{reg}(x)$ is given by

$$\rho_{reg}(x) = \otimes_{i=1}^{n} X^{x_i} = H^{\otimes n} \left(\otimes_{i=1}^{n} Z^{x_i}\right) H^{\otimes n}, \text{ since } X = HZH.$$

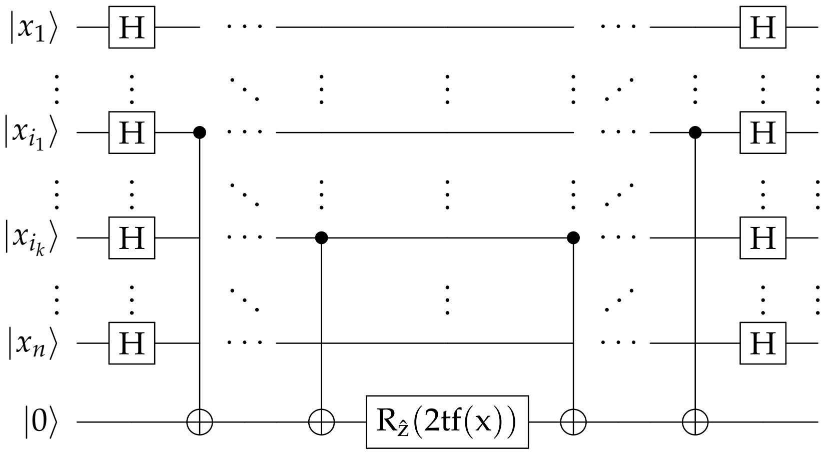


Figure 5.3: A quantum circuit for $U(x,t) = e^{-\imath t f(x)\rho_{reg}(x)}$.

Applying the changes to the operator $U(x,t)$ in Eq. 5.45, we get

$$\begin{aligned} U(x,t) &= e^{-\imath t f(x)\rho_{reg}(x)} = e^{-\imath t f(x)\left[\otimes_{i=1}^{n} X^{x_i}\right]} \\ &= \sum_{l=0}^{\infty} \frac{(-\imath t f(x))^l}{l!} \left[\otimes_{i=1}^{n} X^{x_i}\right]^l \\ &= \sum_{l=0}^{\infty} \frac{(-\imath t f(x)))^{2l}}{(2l)!} I^{\otimes n} + \sum_{l=0}^{\infty} \frac{(-\imath t f(x))^{2l+1}}{(2l+1)!} \left[\otimes_{i=1}^{n} X^{x_i}\right] \\ &= H^{\otimes n} V(x,t) H^{\otimes n}, \qquad V(x,t) = e^{-\imath t f(x)\left[\otimes_{i=1}^{n} Z^{x_i}\right]}. \end{aligned}$$

We see that,

$$\begin{aligned} \left(Z_1^{x_1} \otimes \cdots \otimes Z_n^{x_n}\right) |y\rangle &= (-1)^{x_1 y_1} |y_1\rangle \otimes \cdots \otimes (-1)^{x_n y_n} |y_n\rangle \\ &= (-1)^{\sum_{i=1}^{n} x_i y_i} |y_1\rangle \otimes \cdots \otimes |y_n\rangle \\ &= \begin{cases} |y\rangle, & \text{if } \langle x|y\rangle \mod 2 = 0 \\ -|y\rangle, & \text{if } \langle x|y\rangle \mod 2 = 1. \end{cases} \end{aligned}$$

This implies,

$$V(x,t)|y\rangle = \begin{cases} e^{-\imath t f(x) Z} |y\rangle & \text{if } \langle x|y\rangle \mod 2 = 0 \\ e^{\imath t f(x) Z} |y\rangle & \text{if } \langle x|y\rangle \mod 2 = 1. \end{cases}$$

Thus, the action of the operator $V(x,t)$ is equivalent to the application of the rota-

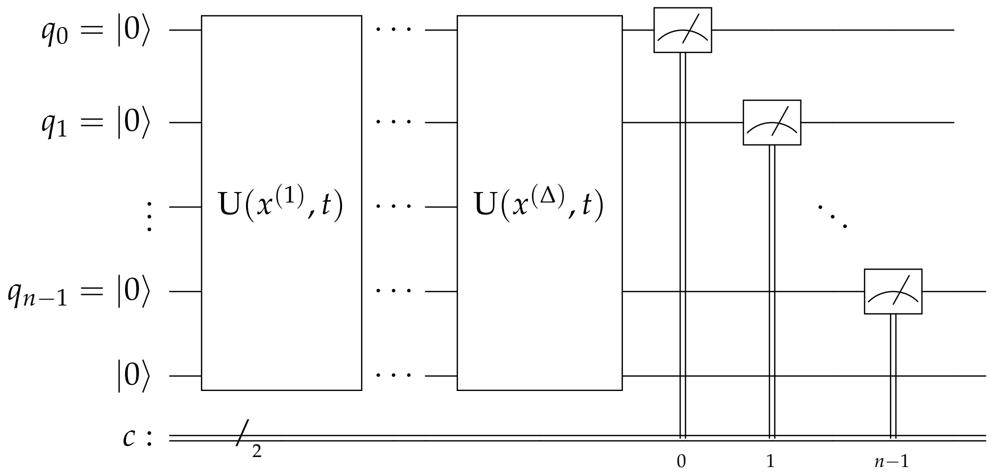


Figure 5.4: An illustration of quantum circuit for CTQW on weighted cubelike graph

tion operator $R_{\hat{z}}(2tf(x))$ about the $\hat{z}$-axis if $\langle x|y\rangle$ is even, and $R_{\hat{z}}(-2tf(x))$ if $\langle x|y\rangle$ is odd. Hence, if x has non-zero entries at positions $i_1, \ldots, i_k$, then the quantum circuit for the operator $e^{-\imath tf(x)\rho_{reg}(x)}$ is depicted by Fig. 5.3. Suppose elements in $\Omega_f = \{y : f(y) \neq 0\}$ are represented by $\Omega_f = \{x^{(1)}, \ldots, x^{(\Delta)}\}$, where Δ is the cardinality of Ω_f, then the quantum circuit for the continuous-time quantum walk is as shown in Fig. 5.4, where the initialized state, in general, is $|0\rangle^{\otimes n}$ along with an ancilla qubit with state $|0\rangle$.

Remark 8. *As seen in Fig. 5.4, the Hadamard gates H applied at the end of $U(x^{(i)}, t)$ and the beginning of $U(x^{(i+1)}, t)$, $1 \leq i < \Delta$, are not required, because $H^2 = I$, thus the actual number of H gates required are $2n$. Secondly, the number of rotation operators used are Δ. Lastly, for each $x \in \Omega_f$, the number of CNOT gates applied are equal to the Hamming weight $wt(x)$ of x. Thus, the total number of CNOT gates used are $\sum_{x \in \Omega_f} wt(x)$.*

5.5.3 The Quantum Simulation

Recall that, if $u \oplus v = \sigma$, where σ is given by Eq. 5.42 in Theorem 21, then $\{u, v\}$ is the PST pair. This partitions the vertex set into PST pairs. The graph shown in Fig. 4.1a admits PST between pairs $\{000, 111\}$, $\{001, 110\}$, $\{010, 101\}$, $\{011, 100\}$, and the other graph in Fig. 4.1b has PST pairs $\{000, 011\}$, $\{001, 010\}$, $\{100, 111\}$,

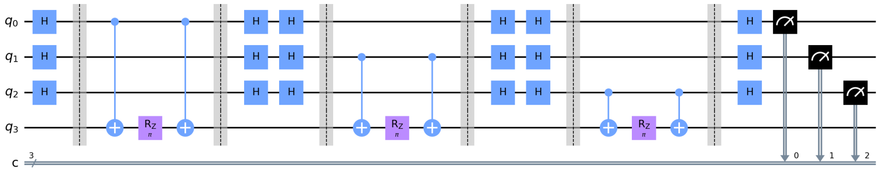


Figure 5.5: A quantum circuit for CTQW on $Cay(\mathbb{Z}_2^3, \{001, 010, 100\})$.

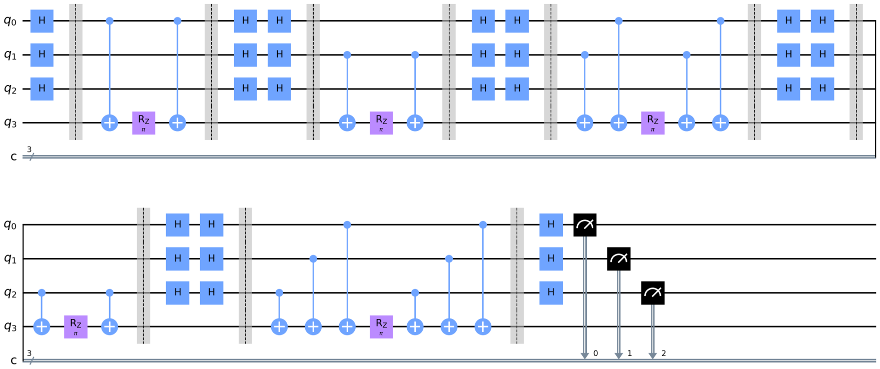


Figure 5.6: A quantum circuit for CTQW on $Cay(\mathbb{Z}_2^3, \{001, 010, 011, 100, 111\})$.

$\{101, 110\}$. Since weighted cubelike graphs, as described in Theorem 21, are vertex-transitive, the study of PST between the pair $\{\mathbf{0}, \sigma\}$ is equivalent to any other pair. Therefore, every quantum circuit is initialized to state $|0\rangle^{\otimes n}$, see Fig. 5.5 and Fig. 5.6 which illustrate quantum circuits for the above graphs mentioned. Suppose the weight function f is defined by; $f(001) = 4$, $f(011) = 8$, $f(101) = 3$, and zero on other elements, then the 3-tuple σ is computed as (using Theorem 21);

$$\begin{aligned}
O_{001} = \{001, 011, 101\} &\implies f(001) + f(011) + f(101) \bmod 2 = 1 \\
&\implies \sigma_1 = 1 \\
O_{010} = \{011\} &\implies f(011) \bmod 2 = 0 \\
&\implies \sigma_2 = 0 \\
O_{100} = \{101\} &\implies f(101) \bmod 2 = 1 \\
&\implies \sigma_3 = 1
\end{aligned}$$

Thus, $\sigma = 101$ and $\{000, 101\}$ is a PST pair. The same is obtained by simulating the quantum circuit shown in Fig. 5.7.

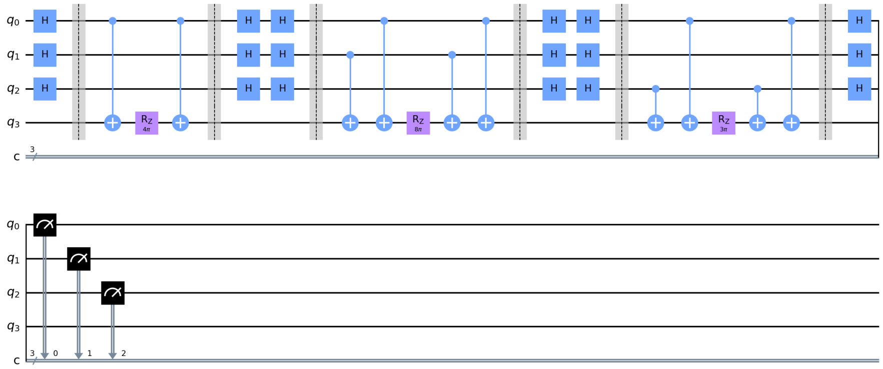


Figure 5.7: A quantum circuit for CTQW on $Cay(\mathbb{Z}_2^3, \{f(001) = 4, f(011) = 8, f(101) = 3\})$.

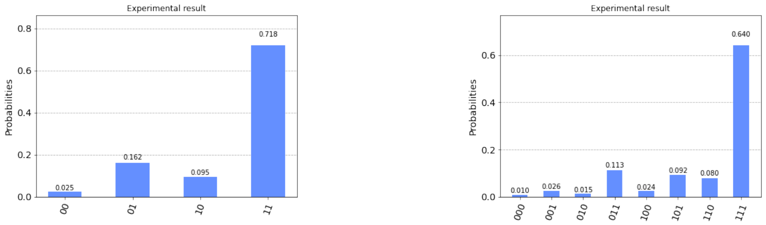


Figure 5.8: Probability distribution of CTQW on the hypercubes $Cay(\mathbb{Z}_2^3, \{01, 10\})$ (left) and $Cay(\mathbb{Z}_2^3, \{001, 010, 100\})$ (right) after time $\frac{\pi}{2}$.

On the other hand, if f is defined by

$$f(010) = 4,\ f(011) = 7,\ f(100) = 8,\ f(101) = 2,\ f(110) = 5, \tag{5.46}$$

then $\sigma = 101$, and $\{000, 101\}$ is a PST pair.

Remark 9. *Given a pair in a cubelike graph, we can assign weights to edges such that PST occurs between the given pair.*

Note 1. *Quantum circuits displayed in Fig. 5.4 can not be run on real quantum computers due to some techincal issues such as quantum decoherence and state fidelity. We have, however, tested small graphs on the computer ibmq_manila as shown in Fig. 5.8.*

Remarks

The unitary matrix P, in Eq. 5.8, is the Fourier transformation that diagonalizes the adjacency matrix of the Cayley graph over $\mathbb{Z}_r^n$. For $r = 2$, the transformation is the

Hadamard transform which has been decomposed and simulated in IBM's quantum simulators and computers. The decomposition of quantum Fourier transform (QFT) is discussed in [59]; however, it can not be used for all Cayley graph over $\mathbb{Z}_r^n$ because the number of qubits must be of the form 2^k, for some k.

5.6 Summary

In section 5.2, we studied properties of normal matrices and constructed weighted Cayley graphs over $\mathbb{Z}_r^n$ having same eigenvectors. We studied the CTQW on these graphs in section 5.3 and extended the work by Cheung et al. [22]. We further discuss the decomposition of the CTQW on weighted cubelike graphs in section 5.5 and verfied the results on IBM's computing platform.

Chapter 6

Conclusion

In this thesis, we have shown the utility of Cayley graphs for computations. The Cayley graphs considered are constructed over $\mathbb{Z}_r^n$.

The first part of the thesis focuses on their utility for multiprocessor computing. The interconnection network of a multiprocessor computing system can be modeled as a graph. Cubelike graphs being Cayley graphs, make a good choice as the graph for these interconnection networks. Further, the parallel algorithms for multiprocessor/parallel computation too can be modeled as a graph. Parallel algorithms that use the divide-and-conquer technique are usually represented as a binary tree. Efficient implementation of a parallel algorithm on a parallel computer is dependent on the presence of the graph of the parallel algorithm as a subgraph of the graph of the interconnection network. Embedding one graph into another helps in showing the extent to which the parallel algorithm can be implemented on an interconnection network. In this thesis, we have studied the embedding of a binary tree as a subgraph in cubelike graphs - particularly hypercubes and augmented cubes. In section 3.3 (Theorem 10), we developed an embedding technique using the recursive structure of hypercubes, which carries out divide-and-conquer algorithms efficiently. This settles a long-standing conjecture by Havel [43], albeit for some special families of binary trees. A similar technique described in section 3.2, if shown correct, can be used to emulate all binary trees on augmented cubes and improve the efficiency of parallel machines which have an augmented cube as their interconnection network.

The second part of this thesis is the study of quantum walks on Cayley graphs, with emphasis on cubelike graphs. Both discrete-time and continuous-time, quantum walks are universal for quantum computation [23, 54], and are useful in designing quantum algorithms. In [50], V. Kendon discusses efficient ways of quantum computing via quantum walks on graphs.

We have implemented quantum circuits for discrete-time coined quantum walks and continuous-time quantum walks on cubelike graphs to study hitting times and perfect state transfer, respectively. We decomposed the evolution operators for DTQW and CTQW on cubelike graphs and constructed corresponding quantum circuits in section 4.3 and 5.5, respectively; thus, the quantum walks on cubelike graphs can be encoded into qubits efficiently, and by running the quantum circuits, we can solve many practical problems in quantum information theory. In the case of CTQW, we have studied PST and periodicity on the weighted Cayley graph over $\mathbb{Z}_r^n$. In the case of DTQW on cubelike graphs, we can compute hitting times using Eq. 4.17, section 4.2, and the conjecture 1, section 4.3. Suppose Γ is a cubelike graph with regularity Δ, then the conjecture gives the target vertex v_{targ}, the approximate value for the evolution step, i.e., $T = \frac{\pi\Delta}{2}$, and states that the parity of T and Δ is same. This fact is used to fix the possible values for T in Eq 4.17, Lemma 4, and so, we only need to compute the coefficient of $|D\rangle\,|v_{targ}\rangle$.

We can study the implementation of DTQW on regular and non-regular graphs. The quantum circuits for the corresponding shift operators can be constructed using the similar technique discussed in section 4.3. Suppose we label each edge using binary strings of fixed length, then the shift can be applied using an expression similar to Eq. 4.28. For the continuous case, we can construct quantum circuits for CTQW on abelian and non-abelian Cayley graphs. The quantum circuit for the quantum Fourier transform is discussed in [59]. Recall that the unitary matrix P that diagonalizes the adjacency matrix A of the Cayley graph is related to Fourier matrices; it is, therefore, reasonable to think of constructing the quantum circuit of the evolution operator for CTQW on Cayley graphs.

References

[1] J. Abhijith et al. Quantum Algorithm Implementations for Beginners. *arXiv:cs.ET/1804.03719*, 2020.

[2] H. Abraham et al. *Qiskit: An Open-source Framework for Quantum Computing*, 2019, qiskit.org.

[3] F. Acasiete, F. Agostini, J. Khatibi Moqadam, and R. Portugal. Implementation of quantum walks on IBM quantum computers. *Quantum Information Processing*, 19(426), 2020.

[4] D. Aharonov, A. Ambainis, J. Kempe, and U. Vazirani. Quantum Walks on Graphs. In *STOC '01: Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing*, pages 50–59, 2001.

[5] Y. Aharonov, L. Davidovich, and N. Zagury. Quantum random walks. *Physical Review A*, 48(2):1687–1690, 1993.

[6] A. Ambainis. Quantum walks and their algorithmic applications. *International Journal of Quantum Information*, 1(4):507–518, 2003.

[7] O. Amini, F. V. Fomin, and S. Saurabh. Counting Subgraphs via Homomorphisms. *SIAM Journal on Discrete Mathematics*, 26(2):695–717, 2012.

[8] R. Angeles-Canul, R. Norton, M. Opperman, C. Paribello, M. Russell, and C. Tamon. Perfect state transfer, integral circulants, and join of graphs. *Quantum Information & Computation*, 10(3-4):325–342, 2010.

[9] R. J. Angeles-canul, R. M. Norton, M. C. Opperman, C. C. Paribello, M. C.

Russell, and C. Tamon. Quantum Perfect State Transfer on Weighted Join Graphs. *International Journal of Quantum Information*, 7(8):1429–1445, 2009.

[10] L. Babai, W. M. Kantor, and E. M. Luks. Computational Complexity and the Classification of Finite Simple Groups. In *24th Annual Symposium on Foundations of Computer Science*, pages 162–171, 1983.

[11] A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. A. Smolin, and H. Weinfurter. Elementary gates for quantum computation. *Physical Review A*, 52(5):3457–3467, 1995.

[12] M. Batty, S. Braunstein, A. Duncan, and S. Rees. Quantum algorithms in group theory. *arXiv:quant-ph/0310133v2*, 2003.

[13] S. W. Bent, D. D. Sleator, and R. E. Tarjan. Biased Search Trees. *SIAM Journal on Computing*, 14(3):545–568, 1985.

[14] A. Bernasconi, C. Godsil, and S. Severini. Quantum networks on cubelike graphs. *Physical Review A*, 78(5):052320, 2008.

[15] T. Beyer and S. M. Hedetniemi. Constant Time Generation of Rooted Trees. *SIAM Journal on Computing*, 9(4):706–712, 1980.

[16] S. Bezrukov, B. Monien, W. Unger, and G. Wechsung. Embedding ladders and caterpillars into the hypercube. *Discrete Applied Mathematics*, 83(1):21–29, 1998.

[17] N. Biggs. *Algebraic Graph Theory*. Cambridge University Press, London, 2nd edition, 1974.

[18] S. Bose. Quantum Communication through an Unmodulated Spin Chain. *Physical Review Letters*, 91(20):207901, 2003.

[19] E. Campos, S. E. Venegas-Andraca, and M. Lanzagorta. Quantum tunneling and quantum walks as algorithmic resources to solve hard K-SAT instances. *Scientific Reports*, 11:16845, 2021.

[20] W.-F. Cao, Y.-G. Yang, D. Li, J.-R. Dong, Y.-H. Zhou, and W.-M. Shi. Quantum state transfer on unsymmetrical graphs via discrete-time quantum walk. *Modern Physics Letters A*, 34(38):1950317, 2019.

[21] R. Chang, W. Gasarch, and J. Toran. On finding the number of graph automorphisms. In *Proceedings of Structure in Complexity Theory. Tenth Annual IEEE Conference*, pages 288–298, 1995.

[22] W. Cheung and C. Godsil. Perfect state transfer in cubelike graphs. *Linear Algebra and its Applications*, 435(10):2468–2474, 2011.

[23] A. M. Childs. Universal Computation by Quantum Walk. *Physical Review Letters*, 102(18):180501, 2009.

[24] A. M. Childs, R. Cleve, E. Deotto, E. Farhi, S. Gutmann, and D. A. Spielman. Exponential Algorithmic Speedup by a Quantum Walk. In *STOC '03: Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing*, pages 59–68, 2003.

[25] A. M. Childs, E. Farhi, and S. Gutmann. An Example of the Difference Between Quantum and Classical Random Walks. *Quantum Information Processing*, 1:35–43, 2002.

[26] A. M. Childs, D. Gosset, and Z. Webb. Universal Computation by Multiparticle Quantum Walk. *Science*, 339(6121):791–794, 2013.

[27] S. Choudum, L. Sivakumar, and V. Sunitha. Graph Embedding and Interconnection Networks. In *K. Thulasiraman, S. Arumugam, A. Brandstädt, and T. Nishizeki (ed.), Handbook of Graph Theory, Combinatorial Optimization, and Algorithms*, pages 653–688. Chapman and Hall/CRC, 2015.

[28] S. A. Choudum and V. Sunitha. Augmented cubes. *Networks*, 40(2):71–84, 2002.

[29] M. Christandl, N. Datta, T. C. Dorlas, A. Ekert, A. Kay, and A. J. Landahl. Perfect transfer of arbitrary states in quantum spin networks. *Physical Review A*, 71(3):032312, 2005.

[30] D. E. Deutsch and R. Penrose. Quantum computational networks. *Proceedings of the Royal Society of London A*, 425(1868):73–90, 1989.

[31] D. P. DiVincenzo. The Physical Implementation of Quantum Computation. *Fortschritte der Physik*, 48(9-11):771–783, 2000.

[32] B. Douglas and J. Wang. Complexity Analysis of Quantum Walk Based Search Algorithms. *Journal of Computational and Theoretical Nanoscience*, 10(7):1601–1605, 2013.

[33] M. Drezgich, A. P. Hines, M. Sarovar, and S. Sastry. Complete Characterization of Mixing Time for the Continuous Quantum Walk on the Hypercube with Markovian Decoherence Model. *Quantum Information and Computation*, 9(9):856–878, 2009.

[34] J. Díaz, M. Serna, and D. M. Thilikos. Counting H-colorings of partial k-trees. *Theoretical Computer Science*, 281(1):291–309, 2002.

[35] E. Farhi and S. Gutmann. Quantum computation and decision trees. *Physical Review A*, 58(2):915–928, 1998.

[36] R. P. Feynman. Quantum Mechanical Computers. *Foundations of Physics*, 16:507–531, 1986.

[37] Y. Ge, B. Greenberg, O. Perez, and C. Tamon. Perfect state transfer, graph products and equitable partitions. *International Journal of Quantum Information*, 9(3):823–842, 2011.

[38] K. Georgopoulos, C. Emary, and P. Zuliani. Comparison of quantum-walk implementations on noisy intermediate-scale quantum computers. *Physical Review A*, 103(2):022408, 2021.

[39] C. Godsil. Periodic Graphs. *The Electronic Journal of Combinatorics*, 18(1), 2011.

[40] C. Godsil. State Transfer on Graphs. *Discrete Mathematics*, 312(1):129–147, 2012.

[41] C. Godsil. When can perfect state transfer occur? *Electronic Journal of Linear Algebra*, 23:877–890, 2012.

[42] F. Harary, J. P. Hayes, and H. Wu. A survey of the theory of hypercube graphs. *Computers & Mathematics with Applications*, 15(4):277–289, 1988.

[43] I. Havel. On Hamiltonian circuits and spanning trees of hypercubes. *Časopis pro pěstování matematiky*, 109(2):135–152, 1984.

[44] I. M. Havel and P. Liebl. One-legged caterpillars span hypercubes. *Journal of Graph Theory*, 10(1):69–77, 1986.

[45] M. R. Henzinger and V. King. Randomized Fully Dynamic Graph Algorithms with Polylogarithmic Time per Operation. *Journal of the ACM*, 46(4):502–516, 1999.

[46] K. Hoffman and R. A. Kunze. *Linear Algebra*. Prentice-Hall, New Jersey, 2nd edition, 2004.

[47] J. Holm, K. de Lichtenberg, and M. Thorup. Poly-Logarithmic Deterministic Fully-Dynamic Algorithms for Connectivity, Minimum Spanning Tree, 2-Edge, and Biconnectivity. *Journal of the ACM*, 48(4):723–760, jul 2001.

[48] M. Štefaňák and S. Skoupý. Perfect state transfer by means of discrete-time quantum walk search algorithms on highly symmetric graphs. *Physical Review A*, 94(2):022301, 2016.

[49] J. Kempe. Discrete Quantum Walks Hit Exponentially Faster. *Probability Theory and Related Fields*, 133(2):215–235, 2005.

[50] V. Kendon. How to Compute Using Quantum Walks. *Electronic Proceedings in Theoretical Computer Science*, 315:1–17, 2020.

[51] D. Kielpinski, C. Monroe, and D. Wineland. Architecture for a large-scale ion-trap quantum computer. *Nature*, 417:709–11, 2002.

[52] P. Knight, E. Roldán, and J. Sipe. Optical cavity implementations of the quantum walk. *Optics Communications*, 227(1-3):147–157, 2003.

[53] C. Lavor, L. Manssur, and R. Portugal. Grover's Algorithm: Quantum Database Search. *arXiv:quant-ph/0301079v1*, 2003.

[54] N. B. Lovett, S. Cooper, M. Everitt, M. Trevers, and V. Kendon. Universal quantum computation using the discrete-time quantum walk. *Physical Review A*, 81(4):042330, 2010.

[55] L. Lovász. Random walks on graphs: A survey. In *D. Miklós, V. T. Sós, and T. Szőnyi (ed.), Combinatorics, Paul Erdős is Eighty*, volume 2, pages 353–398. Bolyai Society Mathematical Studies, Budapest, 1993.

[56] B. Monien and G. Wechsung. Balanced caterpillars of maximum degree 3 and with hairs of arbitrary length are subgraphs of their optimal hypercube. *Journal of Graph Theory*, 87(4):561–580, 2018.

[57] C. Moore and A. Russell. Quantum Walks on the Hypercube. In *Randomization and Approximation Techniques in Computer Science*, pages 164–178, 2002.

[58] J. Mulherkar, R. Rajdeepak, and V. Sunitha. Perfect State Transfer in Weighted Cubelike Graphs. *arXiv:quant-ph/2109.12607*, 2021.

[59] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, New York, 10th anniversary edition, 2011.

[60] R. Otter. The Number of Trees. *Annals of Mathematics*, 49(3):583–599, 1948.

[61] E. Roldán and J. Soriano. Optical implementability of the two-dimensional quantum walk. *Journal of Modern Optics*, 52(18):2649–2657, 2005.

[62] N. Shenvi, J. Kempe, and K. B. Whaley. Quantum random-walk search algorithm. *Physical Review A*, 67(5):052307, 2003.

[63] A. Shioura, A. Tamura, and T. Uno. An Optimal Algorithm for Scanning All Spanning Trees of Undirected Graphs. *SIAM Journal on Computing*, 26(3):678–692, 1997.

[64] S. Singh, B. Adhikari, S. Dutta, and D. Zueco. Perfect state transfer on hypercubes and its implementation using superconducting qubits. *Physical Review A*, 102(6):062609, 2020.

[65] A. J. Skinner, M. E. Davenport, and B. E. Kane. Hydrogenic Spin Quantum Computing in Silicon: A Digital Approach. *Physical Review Letters*, 90(8):087901, 2003.

[66] L. Vinet and H. Zhan. Perfect state transfer on weighted graphs of the Johnson scheme. *Letters in Mathematical Physics*, 110(9):2491–2504, 2020.

[67] E. Wanzambi and S. Andersson. Quantum Computing: Implementing Hitting Time for Coined Quantum Walks on Regular Graphs. *arXiv:quant-ph/2108.02723*, 2021.

[68] F. Xia, J. Liu, H. Nie, Y. Fu, L. Wan, and X. Kong. Random Walks: A Review of Algorithms and Applications. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 4(2):95–107, 2020.

[69] P. Xue, B. C. Sanders, and D. Leibfried. Quantum Walk on a Line for a Trapped Ion. *Physical Review Letters*, 103(18):183602, 2009.

[70] X. Zhan, H. Qin, Z.-h. Bian, J. Li, and P. Xue. Perfect state transfer and efficient quantum routing: A discrete-time quantum-walk approach. *Physical Review A*, 90(1):012331, 2014.

CHAPTER A

Abstract Algebra

A.1 Group Theory

Definition 5. *A group is a non-empty set G equipped with a binary operation $\star$ that satisfy; (a) $\forall a, b \in G$, $a \star b \in G$ (closure), (b) $\forall a, b, c \in G$, $(a \star b) \star c = a \star (b \star c)$ (associativity), (c) $\exists e \in G$, such that $a \star e = e \star a = a$ $\forall a \in G$ (identity), and (d) for each $a \in G$, $\exists b \in G$ such that $a \star b = b \star a = e$ (inverses).*

The element e is called identity and it is unique. The inverse of an element, say $a \in G$, is unique and denoted by a^{-1}. The group G is called abelian or commutative if $a \star b = b \star a$ for all $a, b \in G$. The i-th power of a, where $i \in \mathbb{Z}$, represents the product of a with itself, repeated i times, i.e., $a^i = \underbrace{a \star a \cdots \star a}_{i}$.

Note 2. *We usually write $a \star b$ as ab, unless the group operation is to be specified.*

Example 10. *Let n be a positive integer. The set of integers modulo n, denoted by $\mathbb{Z}_n$, is a group under operation of addition modulo n. The element 0 is the additive identity.*

Example 11. *Let A be a set on $n > 0$ elements. The set of all permutations (bijective maps on A) of A, denoted by $Sym(A)$, is a group under the composition of functions.*

Definition 6. *Let G be a group and H be a subset of G. We say H is a subgroup of G if H is a group under the product of G.*

Example 12. *The set of all $n \times n$ invertible matrices over $\mathbb{F} = \mathbb{R}$ (the real numbers) or $\mathbb{C}$ (the complex numbers), denoted by $GL_n(\mathbb{F})$, is a group under the matrix multiplication, called the general linear group. The subset $SL_n(\mathbb{F})$ consisting of matrices with determinant one is a subgroup, called the special linear group of degree n.*

Theorem 22. *Let G be a group and S be subset of G. Let $\langle S \rangle$ be the smallest subgroup of G containing S. Then the followings are equivalent.*

(i) $\langle S \rangle$ is the set of all elements of G representable as a product of elements of S raised to positive, zero, or negative integer exponents.

(ii) $\langle S \rangle$ is the intersection of all subgroups of G containing S.

We say S generates the subgroup $\langle S \rangle$. If S generates the group G, then S is a generating subset of G. The order of a group G, denoted by $\mid G \mid$, is its cardinality. The order of a group element g, denoted by $\mid g \mid$, is the order of the subgroup generated by g. The order of the identity element is one. If the order of g is finite, then $\langle g \rangle = \{e, g, g^2, \ldots, g^{r-1}\}$.

Proposition 6. *If G is a group and $g \in G$. Then the order of g is the minimum positive integer r such that g^r is the identity.*

Theorem 23 (Lagrange's theorem). *Let G be a finite group and H be a subgroup. Then, $\mid H \mid$ divides $\mid G \mid$.*

A.1.1 Cyclic group

Definition 7. *A group G is said to be cyclic if it is generated by a single element. Suppose g is a generator then elements in G is represented by g^i for some $i \in \mathbb{Z}$.*

The set of integers $\mathbb{Z}$, under the addition operation, is a cyclic group generated by 1 or -1. The additive identity is 0. The order of every non-zero element is infinite. The subgroup generated by an element $m \notin \{-1, 0, 1\}$, has infinite order, i.e., $\langle m \rangle = \{mi : i \in \mathbb{Z}\}$.

Theorem 24. *Let G be a group. Let $g \in G$ with finite order $r =\mid g \mid$, and $H = \langle g \rangle$. Then the following holds.*

1. *If $g^n = 1$ and $g^m = 1$, for some $n, m > 0$, then $g^d = 1$, where $d = gcd(n, m)$ is the greatest common divisor of n and m.*

2. $\mid g^n \mid = \frac{r}{gcd(r,n)}$.

3. *$H = \langle g^n \rangle$ if and only if $gcd(n, r) = 1$*

A.1.2 Group homomorphism

Definition 8. *A (group) homomorphism ϕ from a group G_1 to another group G_2 is a map satisfying $\phi(gh) = \phi(g)\phi(h)$, $\forall g, h \in G_1$. An (group) isomorphism is a bijective homomorphism, in which case, G_1 and G_2 are called isomorphic, denoted by $G_1 \cong G_2$. If $G_1 = G_2$, then the isormorphism is known as automorphism.*

Proposition 7. *The kernel* ker ϕ *is a subgroup of G and the image of ϕ, denoted by $Im\ \phi$, is a subgroup of G_2.*

Proposition 8. *Let $\phi : G_1 \to G_2$ be a group homomorphism. Let e_1 and e_2 be identities of G_1 and G_2, respectively. Then the following holds.*

- $\phi(e_1) = e_2$.
- $\phi(g^{-1}) = \phi(g)^{-1}$.
- $\phi(g^i) = \phi(g)^i$, $i \in \mathbb{Z}$.
- *The kernel of ϕ, $ker\ \phi = \{g \in G : \phi(g) = 1\}$, is a subgroup of G_1.*
- *The image of ϕ, $im\ \phi = \{\phi(g) : g \in G\}$, is a subgroup of G_2.*

Example 13. *Let G be a group. Let $g \in G$.*

- *If $\mid g \mid = \infty$, then $\langle g \rangle \cong \mathbb{Z}$, where the group homomorphism is given by $\phi(g^n) = n$.*
- *If $r = \mid g \mid < \infty$, then $\langle g \rangle \cong \mathbb{Z}_r$, where the group homomorphism is given by $\phi(g^i) = ni \bmod r$ with $gcd(n, r) = 1$.*

Theorem 25 (Cayley's theorem)**.** *Every group is isomorphic to a subgroup of $Sym(A)$ for some appropriate A.*

Since symmetric groups on two distinct sets of same cardinality are isomomorphic, we use $Sym(n)$ to denote the symmetric group of degree n, where n is the cardinality of a set.

A.1.3 Direct products

Definition 9. *Let $G_1, G_2, \dots, G_m$ be m groups. The direct product $G_1 \times G_2 \times \cdots \times G_m$ is a group consisting of n-tupes $(g_1, g_2, \dots, g_m)$, where $g_i \in G_i$, with operation defined componentwise, viz.,*

$$(g_1, g_2, \dots, g_m)(h_1, h_2, \dots, h_m) := (g_1 h_1, g_2 h_2, \dots, g_m h_m).$$

The direct product of m copies of a group G is denoted by $G^m = G \times G \times \cdots \times G$.

Example 14. *Let $G = \mathbb{Z}_2$. Then, the direct product of m copies of $\mathbb{Z}_2$, denoted by $\mathbb{Z}_2^m$, is a group with operation*

$$(x_1, \dots, x_m) \oplus (y_1, \dots, y_m) = (x_1 \oplus y_1, \dots, x_m \oplus y_m),$$

where $x_i \oplus y_i$ is the addition modulo 2.

Theorem 26 (Fundamental theorem of finitely generated abelian groups)**.** *Let G be a finitely generated group, i.e., $G = \langle S \rangle$ for some finite subset $S \subset G$. If G is abelian then*

$$G \cong \mathbb{Z}^r \times \mathbb{Z}_{n_1} \times \mathbb{Z}_{n_2} \times \cdots \times \mathbb{Z}_{n_s}, \tag{A.1}$$

for some integers $r, n_1, n_2, \dots, n_s$ satisfying;

- *$r \geq 0$ and $n_j \geq 2$, $1 \leq j \leq s$,*
- *n_{j+1} divides n_j.*

The expression in Eq. A.1 is unique upto rearrangement.

Example 15. *A group of order four is either isomorphic to $\mathbb{Z}_4$ or $\mathbb{Z}_2 \times \mathbb{Z}_2$.*

A.1.4 Group action

Definition 10. *A group action of a group G on a set A is a function $\sigma : G \to S$ satisfying; $\forall a \in A$, (i) $\sigma(e)a = a$, and (ii) $\sigma(gh)a = \sigma(g)(\sigma(h)a)$ (compatibility).*

Theorem 27. *For a fixed group element g, $\sigma(g) : A \to A$ is a permutation of A and the map $\psi : G \to Sym(A)$, defined by $g \mapsto \sigma(g)$, is a group homomorphism.*

Note 3. *We usually write $\sigma(g)a$ as ga.*

A.2 Linear Algebra

A field $\mathbb{F}$ is a non-empty set equipped with two operations, namely, addition $+$ and multiplication $\cdot$, such that $\mathbb{F}$ is abelian group under both operations, and multiplication is distributive over addition. A vector space V is a non-empty set, whose elements are called vectors, equipped with two operations, namely, vector addition $+$ and scalar multiplication $\cdot$ satisfying (a) V is abelian group under the addition operation, (b) F acts on V, and (c) scalar multiplication is distributive over vector addition and vice-versa.

Note 4. *We use the convention, mentioned in many books, of using the symbol $+$ for addition and $\cdot$ for multiplication between two entities in any field or a vector space. We do not mention the field if it is clear from the context which field is considered.*

Example 16. *The real numbers $\mathbb{R}$ and the complex numbers $\mathbb{C}$ are infinite fields. The integers modulo p, $\mathbb{Z}_p$, where p is prime, is a finite field.*

Example 17. *The cartesian product of fields $\mathbb{F}$, denoted by $\mathbb{F}^n$, is a vector space where vector addition and scalar multiplication are induced by field operations and they are defined componentwise, viz., $\lambda(x_1, \dots, x_n) + (y_1, \dots, y_n) = (\lambda x_1 + y_1, \dots, \lambda x_n + y_n)$.*

Definition 11. *A subspace W of a vector space V, over the field $\mathbb{F}$, is a subset which is itself a vector space, over the same field, equipped with the operations on V.*

Example 18. *The set of all $n \times n$ matrices over the field $\mathbb{F}$, denoted by $M_n(\mathbb{F})$, is a vector space. The set of symmetric matrices is a subspace of $M_n(\mathbb{F})$. If $\mathbb{F} = \mathbb{C}$, the set of Hermitian (self-adjoint) matrices is not a subspace.*

Definition 12. *A vector v in a vector space V is a linear combination of the vectors v_1, v_2, $\dots$, v_m if there exist scalars $\lambda_1, \lambda_2, \dots, \lambda_m$ such that $v = \lambda_1 v_1 + \lambda_2 v_2 + \cdots + \lambda_m v_m$.*

Definition 13. *Let S be a subset of a vector space V. The subspace spanned by S is the set of all vectors which are linear combinations of finitely many vectors in S.*

Note 5. *A linear combination is a sum of finitely many non-zero vectors.*

Definition 14. *A subset S of a vector space V is said to be linearly dependent if the zero vector 0 (the additive identity) is a linear combination of some distinct vectors in S. If S is not linearly dependent then it is called linearly independent.*

Definition 15. *A basis B for a vector space V is a linearly independent set of vectors that spans the space.*

Example 19. *A basis for $\mathbb{F}^n$, is the set of vectors $\epsilon_1, \epsilon_2, \ldots, \epsilon_n$, where ϵ_j is the n-tuple with value 1 at j-th position and zero elsewhere. This basis is called the standard basis of $\mathbb{F}^n$.*

Theorem 28. *If a basis of a vector space has finitely many vectors then any basis is finite and has the same number elements.*

Definition 16. *The dimension of a vector space is the cardinality of its basis.*

Definition 17. *Let B be an ordered basis of a vector space V of dimension n, then the coordinate of a vector v relative to B is represented by the n-tuple $(\lambda_1, \lambda_2, \ldots, \lambda_n)$, where $\lambda_1, \lambda_2, \ldots, \lambda_n$ are coefficients in the linear combination of vectors in B.*

Note 6. *Whenever we use the term basis, we imply an ordered basis.*

A.2.1 Linear Transformation

Definition 18. *A linear transformation is a function T from a vector space V_1 to another vector space V_2, over the field $\mathbb{F}$, satisfying; $T(\lambda u + v) = \lambda Tu + Tv$. A bijective linear transformation is called (linear) isomorphism.*

Note 7. *In quantum physics, a linear transformation from a vector space to itself is called an (linear) operator.*

Theorem 29. *Every n-dimensional vector space over the field $\mathbb{F}$ is isomorphic to $\mathbb{F}^n$.*

Theorem 30. *Let V be an n-dimensional vector space and W be an m-dimensional vector space, over the field $\mathbb{F}$, with respective (ordered) basis B and B'. There is one-one correspondence between the set of all linear transformation from V into W and the set of all $m \times n$ matrices over $\mathbb{F}$.*

A.2.2 Hilbert spaces

Definition 19. *An inner product on a vector space V, over the field $\mathbb{C}$ or $\mathbb{R}$, is a function $\langle \cdot, \cdot \rangle \times V \to \mathbb{F}$ satisfying; $\forall u, v, w \in V$ (a) $\langle u, u \rangle > 0$ only if $u \neq 0$ and $\langle u, u \rangle = 0$ only if $u = 0$, (b) $\langle u, \lambda v + w \rangle = \lambda \langle u, v \rangle + \langle u, w \rangle, \forall \lambda \in \mathbb{F}$, (c) $\langle u, v \rangle = \overline{\langle v, u \rangle}$. A vector space equipped with an inner product is called an inner product space.*

Example 20. *The complex vector space $\mathbb{C}^n$ is an inner product space, wherein inner product of $u = (\lambda_1, \ldots, \lambda_n)$ and $v = (\mu_1, \ldots, \mu_n)$ is defined by $\langle u, v \rangle := \sum_{j=1}^{n} \bar{\lambda}_j \mu_j$.*

Definition 20. *A norm induced by an inner product is defined by $\|x\| := \sqrt{(x, x)}$. The distance between two vectors u and v is given by $d(u, v) := \|u + (-v)\|$.*

Definition 21. *A sequence of vectors $v_1, v_2, v_3, \ldots,$ in a vector space V is called Cauchy if for every positive real number ϵ there is a positive integer N such that for all integers $m, n > N$, the distance satisfies $d(v_m, v_m) < \epsilon$. The vector v is said to be a limit of the sequence if for every real number $\epsilon > 0$, there exists a positive integer N such that for all $n \geq N$, $d(v_n, v) < \epsilon$. Alternatively, we say the sequence converges to v.*

Definition 22. *A Hilbert space is a complete inner product space, i.e., every Cauchy sequence in the space has a limit in it.*

Definition 23. *A vector space V is separable if it contains a countable dense subset S, i.e., there exists a sequence of vectors such that every non-empty open subset of the space contains atleast one vector of the sequence.*

A.2.3 Orthonormal basis and Eigenvectors

Definition 24. *In an inner product space V, two vectors u and v are said to be orthogonal if their inner product is zero, i.e., $\langle u, v \rangle = 0$. A vector w is normal if its norm induced by* *the inner product is 1, i.e., $\|w\| = 1$. A set of vectors is called orthonormal set if they are mutually orthogonal and each vector is normal.*

Proposition 9 (Fourier expansion)**.** *Let $B = \{v_1, v_2, \dots, v_n\}$ be an orthonormal basis for an inner product space V, then each vector $v \in V$ can be expressed as*

$$v = \langle v_1, v \rangle v_1 + \langle v_2, v \rangle v_2 + \cdots + \langle v_n, v \rangle v_n.$$

Theorem 31 (Gram-Schmidt Orthonormalization)**.** *Every finite dimensional vector space has orthonormal basis.*

Definition 25. *Let A be $n \times n$ matrix over the field $\mathbb{F}$. A scalar λ is an eigenvalue of A if there exists a non-zero vector u satisfying $Au = \lambda u$. The eigenspace relative to λ is the subspace spanned by corresponding eigenvectors of A.*

Theorem 32. *If A has an eigenvalue λ, then $A - \lambda I$ is singular (not invertible), i.e., $det(A - \lambda I) = 0$, where I is the (multiplicative) identity of $GL_n(\mathbb{F})$.*

Definition 26. *Let V be a finite inner product space.. Let W be a subspace of V with orthonormal basis $\{v_1, v_2, \dots, v_m\}$. The orthogonal projection of a vector $v \in V$ on W is defined by $\sum_{j=1}^{n} \langle v_j, v \rangle v_j$. The function that maps each vector in V to its orthogonal projection on W is called the orthogonal projection of V on W.*

A.2.4 Spectral theorem

An $n \times n$ normal matrix $N \in \mathbb{C}^n \times \mathbb{C}^n$ is defined by

$$NN^\dagger = N^\dagger N,$$

where $N^\dagger$ is the adjoint (complex-conjugate transpose) of N. If $NN^\dagger = I$, then N is called unitary matrix. Suppose $N = N^\dagger$ then N is Hermitian (also known as self-adjoint). If entries in N are real and N is Hermitian then N is a real-symmetric matrix, i.e., $N = N^T$. The spectral theory diagonalizes a normal matrix as stated below.

Theorem 33 (Spectral theorem). *[46] Let N be an $n \times n$ normal matrix. Then, the following statements hold true.*

1. *There is an $n \times n$ unitary matrix P such that $D = P^{-1}NP$ is diagonal.*
2. *Suppose $\lambda_1, \ldots, \lambda_m$ are distinct eigenvalues of N, and E_j is the orthogonal projection on the eigenspace associated with λ_j, then the spectral decomposition (also called spectral resolution) of N is given by*

$$N = \lambda_1 E_1 + \cdots + \lambda_m E_m. \tag{A.2}$$

The set $S = \{\lambda_1, \ldots, \lambda_m\}$ of eigenvalues is called the spectrum of N. In Theorem 33, the j-th column of P, denoted by P_{*j}, is an eigenvector of N with eigenvalue $d_j = D_{j,j}$, i.e., $NP_{*j} = d_j P_{*j}$. The spectral theory for unitary, Hermitian and real-symmetric matrices can be formulated as;

Theorem 34. *[46] If N is a normal matrix with unitary matrix P satisfying $D = P^{-1}DP$, such that D is diagonal, then,*

1. *N is self-adjoint iff its eigenvalues are real,*
2. *N is unitary iff its eigenvalues are of absolute value* 1.
3. *P is orthogonal and D is real iff N is real-symmetric matrix.*

Another useful result from linear algebra associates spectra of normal matrix N with its matrix exponential e^N.

Theorem 35. *[46] Let N be a normal matrix with unitary matrix P satisfying $D = P^{-1}NP$, such that D is diagonal. Let $N = \sum_{j=1}^{m} \lambda_j E_j$ be the spectral decomposition of N. Suppose f is a complex-valued function defined over S, then the following statements hold true.*

1. *The linear operator $f(N)$ defined by*

$$f(N) = \sum_{j=1}^{m} f(\lambda_j) E_j$$

is a diagonalizable normal operator with spectrum $f(S)$. In other words,

$$f(D) = P^{-1} f(N) P$$

is diagonal with the same unitary matrix P, i.e., the j-th column P_{*j} of P is eigenvector of $f(N)$ with eigenvalue $f(d_j)$, $1 \leq j \leq n$.

A.2.5 Discrete Fourier Transform

Definition 27. *Let ω be an n-th root of unity, i.e., $\omega = e^{2\pi/n}$. The Fourier matrix of order n is the $n \times n$ matrix, denoted by F_n, with (i,j)-th entry $\bar{\omega}$. The discrete Fourier transform of a vector $v \in \mathbb{C}^n$ is the product $F_n v$, and the inverse Fourier transform of v is $F_n^{-1} v$.*

The Fourier matrix is symmetric and $F_n^\dagger = \bar{F}_n$. The matrix $\frac{1}{\sqrt{n}} F_n$ is unitary with the inverse $\frac{1}{\sqrt{n} \bar{F}_n}$. The k-th entry of $F_n v$ is $\sum_{j=1}^{n} \lambda_j \bar{\omega}^{jk}$ and that of $F_n^{-1} v$ is $\sum_{j=1}^{n} \lambda_j \omega^{jk}$, where λ_j is the j-th entry of v.

A.2.6 Dirac notation

This notation is usually used in quantum mechanics where a vector space is a separable Hilbert space. A vector v is denoted by $|v\rangle$ and pronounced as ket-v. A linear transformation $f : V \to \mathbb{F}$ is called a linear functional and the space of all linear functionals is called the dual space of V, denoted by V^*. A linear functional f is denoted by $\langle f|$ and pronounced as bra-f.

Theorem 36. *Let V be a finite-dimensional vector space with a basis $B = \{v_1, \ldots, v_n\}$, then there is a unique dual basis $B^* = \{f_1, \ldots, f_n\}$ such that $f_i(v_j) = \delta_{ij}$. Moreover, a linear function f can be expressed as $f = \sum_{j=1}^{n} f(v_j) f_j$ and a vector $v \in V$ can be expressed as $v = \sum_{j=1}^{n} f_j(v) v_j$.*

Theorem 37. *Let V be a finite-dimensional inner product space, and f be a linear functional on V. There there is a unique vector $u \in V$ such that $f(v) = \langle u, v\rangle$ for all $v \in V$.*

We use Theorem 37, to define bra-u by $\langle u| := f$ and let $\langle u|v\rangle$ denote the inner product $\langle |u\rangle, |v\rangle\rangle$, i.e., $\langle u|v\rangle := \langle u, v\rangle$.

Example 21. *In the vector space $\mathbb{C}^{n\times 1}$, a vector $|v\rangle$ is a column matrix and $\langle v|$ is a row vector in $\mathbb{C}^{1\times n}$. Notice that $\langle v|^{\dagger} = |v\rangle$ and $|v\rangle^{\dagger} = \langle v|$.*

We use the convention that $|u\rangle$ is always a column vector and $\langle u|$ is complex-conjugate of $|u\rangle$.

Definition 28. *The outer product of vectors $|u\rangle$ and $|v\rangle$ in $\mathbb{C}^n$ is an $n \times n$ matrix $|u\rangle\langle v|$ whose (i,j)-th entry is $\lambda_i \bar{\mu}_j$, where λ_i is the i-th coordinate of $|u\rangle$ and μ_j is the j-th coordinate of $|v\rangle$.*

A.2.7 Tensor products

Definition 29. *Let V and W be Hilbert spaces over the field $\mathbb{C}$, with basis $\{|v_1\rangle, \ldots, |v_n\rangle\}$ and $\{|w_1\rangle, \ldots, |w_m\rangle\}$, respectively. The tensor product of V and W, denoted by $V \otimes W$, is an nm-dimensional Hilbert space with the basis $\{|v_1\rangle \otimes |w_1\rangle, |v_1\rangle \otimes |w_2\rangle, \ldots, |v_n\rangle \otimes |w_m\rangle\}$.*

A generic vector $|\psi\rangle$ in $V \otimes W$ is a linear combination of the basis vectors,

$$|\psi\rangle = \sum_{i=1}^{n}\sum_{j=1}^{m} a_{ij}|v_i\rangle \otimes |w_j\rangle.$$

The tensor product is bilinear, i.e., for $v, v_1, v_2 \in V$ and $w, w_1, w_2 \in W$ and $a \in \mathbb{C}$,

$$|v\rangle \otimes (a|w_1\rangle + b|w_2\rangle) = a|v\rangle \otimes |w_1\rangle + b|v\rangle \otimes |w_2\rangle,$$

$$(a|v_1\rangle + b|v_2\rangle) \otimes |w\rangle = a|v_1\rangle \otimes |w\rangle + b|v_2\rangle \otimes |w\rangle.$$

The scalar multiplication is given by,

$$a(|v\rangle \otimes |w\rangle) = (a|v\rangle) \otimes |w\rangle = |v\rangle \otimes (a|w\rangle) \qquad a \in \mathbb{C},\ v \in V,\ w \in W.$$

The tensor product of a linear operator A on V and B on W, denoted by $A \otimes B$, is a linear operator defined by

$$(A \otimes B)(|v\rangle \otimes |w\rangle) = (A|v\rangle) \otimes (B|w\rangle).$$

Remark 10. *An outer product is a tensor product.*

Let A_1, A_2 be operators in V and B_1, B_2 be operators on W, then

$$(A_1 \otimes B_1) \circ (A_2 \otimes B_2) = (A_1 \circ A_2) \otimes (B_1 \circ B_2).$$

The inner product of vectors is given by

$$\Big\langle |v_1\rangle \otimes |w_1\rangle, |v_2\rangle \otimes |w_2\rangle \Big\rangle = \langle v_1|v_2\rangle\langle w_1|w_2\rangle.$$

Let A be an $m \times n$ matrix and B be a $p \times q$ matrix. Then, $A \otimes B$ is $mp \times nq$ matrix given by,

$$A \otimes B = \begin{bmatrix} a_{11}B \cdots a_{1n}B \\ \ddots \\ a_{m1}B \cdots a_{mn}B \end{bmatrix}.$$

A.3 Representation Theory of finite abelian group

Definition 30 (Group representation)**.** *A representation of a group G is a (group) homomorphism from G into general linear group $GL_n(\mathbb{F})$, for some positive integer n.*

Let G be a finite abelian group, and σ be a representation of G. Fix $g \in G$, then,

$$\begin{aligned} &\sigma(gx)v = \sigma(xg)v \qquad \forall x \in G \\ \implies &\sigma(g)\sigma(x)v = \sigma(x)\sigma(g)v \end{aligned}$$

Thus, σ commutes will all invertible transformations.

CHAPTER B

Quantum Computation

B.1 Postulates of quantum mechanics

A quantum mechanical particle moves according to specific rules called postulates of quantum mechanics, which came by experiments. There are four rules as follows.

B.1.1 The first postulate

A quantum state of an isolated quantum system is a unit vector in a Hilbert space. The associated Hilbert space is known as the state space of the quantum system.

Example 22. *A spin-$\frac{1}{2}$ particle has quantum state a linear combination of a spin up $|\uparrow\rangle$ state and a spin down $|\downarrow\rangle$ state. The quantum state of the spin particle can be realised as a unit vector in $\mathbb{C}^2$ with the computational basis $\{|0\rangle, |1\rangle\}$, where $|0\rangle$ and $|1\rangle$ denote $|\uparrow\rangle$ and $|\downarrow\rangle$, respectively. Thus, if $|\psi\rangle$ is a quantum state of the particle then there exists $a, b \in \mathbb{C}$ satisfying $|a|^2 + |b|^2 = 1$ such that $|\psi\rangle = a|0\rangle + b|1\rangle$.*

In general, the computational basis of an N-dimensional Hilbert space is denoted by $\{|0\rangle,\ |1\rangle, \dots,\ |N-1\rangle\}$, and a quantum state of the associated system is written as $|\psi\rangle = \sum_{i=0}^{N-1} a_i |i\rangle$, for some scalars satisying $\sum_{i=1}^{N-1} |a_i|^2 = 1$.

B.1.2 The second postulate

A closed quantum system evolves according to the Schrödinger equation

$$\iota \frac{d}{dt} |\psi(t)\rangle = \hat{H} |\psi(t)\rangle, \qquad \iota = \sqrt{-1},$$

where $\psi(t)$ is the state of the system at time t, and $\hat{H}$ is the Hamiltonian of the system. If $\hat{H}$ is time-independent then the solution of the Schrödinger equation is given by

$$|\psi(t)\rangle = e^{-\imath t\hat{H}} |\psi(0)\rangle .$$

In general, if $|\psi(0)\rangle$ is the initial state of the system, then the state of the system after time t is obtained by a unitary operator $\mathcal{U}$, i.e. $|\psi(t)\rangle = \mathcal{U} |\psi(0)\rangle$.

B.1.3 The third postulate

Projective quantum measurement of a quantum state of the system is described by applying an observable M, which is a Hermitian operator, on the state space of the quantum system. The result obtained is one of the real eigenvalues of the observable. The measurement destroys the state, and the system acquires a new state. Suppose $\sum_\lambda \lambda E_\lambda$ is the spectral decomposition of M, then the probability that the measurement gives a value λ is $\langle\psi|E_\lambda|\psi\rangle$. If λ occurs then the state of the system immediately after the measurement is $\frac{1}{\sqrt{\langle\psi|E_\lambda|\psi\rangle}} E_\lambda |\psi\rangle$.

Example 23. *Consider the system of the spin-$\frac{1}{2}$ particle. Define two operators $E_\lambda = |0\rangle\langle 0|$ and $E_\mu = |1\rangle\langle 1|$. Clearly, they are projective operators. Suppose the state of the qubit is $|\psi\rangle = a|0\rangle + b|1\rangle$, then upon measuring the state the value λ occurs with probability $\langle\psi| |0\rangle\langle 0| |\psi\rangle = |a|^2$. Similarly, the value μ occurs with probability $\langle\psi| |1\rangle\langle 1| |\psi\rangle = |b|^2$. The state after the measurement is either $\frac{a}{|a|}|0\rangle$ or $\frac{b}{|b|}|1\rangle$.*

B.1.4 The fourth postulate

A composite quantum system consists of two or more quantum systems. The associated state space is the tensor product of Hilbert spaces associated with each quantum system. Suppose a system of m quantum states are respectively associated with Hilbert spaces $\mathcal{H}_1, \ldots, \mathcal{H}_m$, then the composite system has the state space $\mathcal{H} = \mathcal{H}_1 \otimes \cdots \otimes \mathcal{H}_m$. The computational basis for $\mathcal{H}$ is given by $\{|i\rangle : 0 \leq i \leq N-1\}$, where N is the dimension of $\mathcal{H}$, and $|i\rangle = |i_1\rangle \otimes \cdots \otimes |i_m\rangle$, where $|i_j\rangle$ is a computational basis vector for the j-th component state space $\mathcal{H}_j$.

Example 24. *In quantum computing and quantum information theory, quantum bit (popularly known as qubit) is the functional unit, which is a unit vector in $\mathbb{C}^2$. A spin-$\frac{1}{2}$ particle is an example of a qubit. A system of n qubits evolves in the Hilbert space $\left(\mathbb{C}^2\right)^{\otimes n}$, where each qubit has the state space $\mathbb{C}^2$. The corresponding computational basis is denoted by*

$$\{|k\rangle \equiv |x_{n-1} \cdots x_1 x_0\rangle : \, k = \sum_{i=0}^{n-1} x_i 2^i, \, 0 \leq k \leq 2^n - 1\},$$

where the vector $|x_{n-1} \cdots x_1 x_0\rangle$ is the short form for $|x_{n-1}\rangle \otimes \cdots \otimes |x_1\rangle \otimes |x_0\rangle$ and $|k\rangle$ is the decimal notation for the vector. The projective measurement in this basis is the observable $M = \sum_{k=0}^{2^n-1} \lambda_k |k\rangle \langle k|$, for some real scalars λ_k, $0 \leq k \leq 2^n - 1$. Let $|\psi\rangle = \sum_{k=0}^{2^n-1} a_k |k\rangle$ be a quantum state of the composite qubits, where $\sum_{k=0}^{2^n-1} |a_k|^2 = 1$. Then, the measurement outputs a real value λ_k with the probability $\langle\psi| |k\rangle \langle k| |\psi\rangle = |a_k|^2$, and the state after the measurement is $|k\rangle$, where the global phase $\frac{a_k}{|a_k|}$ has been ignored. We see that, the state of the n-qubit system after the measurement is represented by one of its computational basis vector.

Note 8. *A quantum state of a composite system may not be separable, i.e., some quantum states can not be written as tensor product of individual particle's state in respective Hilbert spaces. For example, it is impossible to write $\frac{1}{\sqrt{2}}(|0\rangle |0\rangle + |1\rangle |1\rangle$ as*

$$(a |0\rangle + b |1\rangle) \otimes (a' |0\rangle + b' |1\rangle).$$

Such phenomena is called entangled state.

B.2 Quantum Fourier Transform

Given an orthonormal basis $\{|0\rangle, |1\rangle, \ldots, |N-1\rangle\}$, the quantum Fourier transform (QFT) is a linear operator that acts on an arbitrary state as

$$\mathcal{F} : \sum_{j=0}^{N-1} x_j |j\rangle \longmapsto \sum_{k=0}^{N-1} y_k |k\rangle, \tag{B.1}$$

where the amplitudes y_k are the discrete Fourier transform of the amplitudes x_j, i.e.,

$$y_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j \omega^{jk} \left|k\right\rangle , \qquad \omega = e^{\frac{2\pi\imath}{N}} , \tag{B.2}$$

where ω is the N-th root of unity. The quantum Fourier transform F_N is unitary, viz.,

$$\begin{aligned}
(\mathcal{F}\mathcal{F}^{\dagger})_{rs} &= \sum_{k=0}^{N-1} \mathcal{F}_{rk}\bar{\mathcal{F}}_{sk} \\
&= \frac{1}{N} \sum_{k=0}^{N-1} \omega^{rk}\omega^{-sk} \\
&= \frac{1}{N} \sum_{k=0}^{N-1} \omega^{k(r-s)} = \begin{cases} 1 & ; \text{ if } r = s \\ 0 & ; \text{ otherwise.} \end{cases}
\end{aligned} \tag{B.3}$$

B.3 Quantum Circuits

We describe an abstract model, and not the actual quantum computer, for quantum computation that is discussed in [11, 30]; however one can find some earlier works on the physical realization of quantum computers in [18, 31, 51, 65].

A quantum circuit is a model for quantum computation that consists of (1) quantum register that stores quantum information as inputs, (2) quantum gates that operate on input values, (3) meters that yield classical values as outputs, and (4) quantum wires through which quantum information travels. Quantum information is the quantum state of the system under consideration. A quantum register is a system of multiple qubits, where each qubit is a two-level quantum system. So, if the size of a register is n, then it stores information in the form of a unit vector in $\mathbb{C}^{2^{\otimes n}}$. Quantum information, including entangled state, travels through quantum wires at an arbitrary distance. A quantum gate is a unitary matrix; it is a reversible operator that maps the space of a register to itself. A pictorial representation of a quantum circuit for a discrete-time quantum walk on the cycle of size four is shown in Fig. B.1, wherein double horizontal lines at the bottom of the circuit and double vertical lines attached to each meter represent the classical

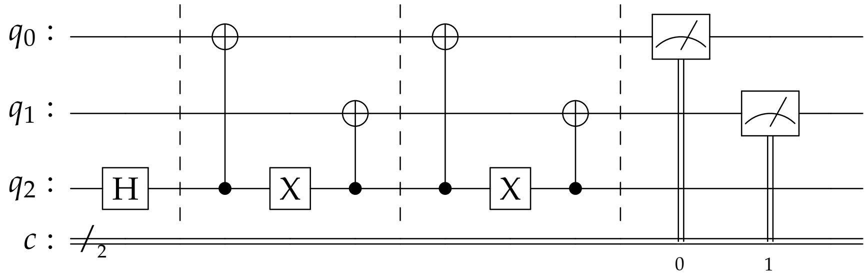


Figure B.1: A quantum circuit for DTWQ on $\mathcal{Q}_2$.

channel through which classical values travel.

B.3.1 Quantum gates

Any unitary matrix of order 2^k, for some k, serves as a quantum gate. However, only the gates of smaller dimensions, i.e. $k = 1$ or $k = 2$, are useful for practical purposes, i.e., larger quantum gates are not efficient. It is, therefore, important to decompose large unitary gates into smaller ones. In [11], it is shown that all unitary gates can be expressed as a composition of one-qubit gates and two-qubit gates. Therefore, it is enough to study at most two-qubit gates. The most commonly used one-qubit gates are;

1. Pauli matrices $\mathrm{X} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$, $\mathrm{Y} = \begin{bmatrix} 0 & -\iota \\ \iota & 0 \end{bmatrix}$ and $\mathrm{Z} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$,

2. Hadamard gate $\mathrm{H} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -2 \end{bmatrix}$,

3. Phase gate $\mathrm{S} = \begin{bmatrix} 1 & 0 \\ 0 & \iota \end{bmatrix}$,

4. $\pi/8$ gate $\mathrm{T} = \begin{bmatrix} 1 & 0 \\ 0 & e^{\iota \frac{\pi}{4}} \end{bmatrix}$,

5. Rotation operators about $\hat{x}, \hat{y}$ and $\hat{z}$ axes, respectively, are

$$
\begin{aligned}
R_{\hat{x}}(\theta) &= e^{-\iota\theta\frac{\mathrm{X}}{2}} = \cos\frac{\theta}{2}\mathrm{I} - \iota\sin\frac{\theta}{2}\mathrm{X},\\
R_{\hat{y}}(\theta) &= e^{-\iota\theta\frac{\mathrm{Y}}{2}} = \cos\frac{\theta}{2}I - \iota\sin\frac{\theta}{2}Y,\\
R_{\hat{z}}(\theta) &= e^{-\iota\theta\frac{\mathrm{Z}}{2}} = \cos\frac{\theta}{2}I - \iota\sin\frac{\theta}{2}Z.
\end{aligned}
$$

The rotation operator about an arbitrary direction $\hat{n} = x\hat{x} + y\hat{y} + z\hat{z}$ is given by

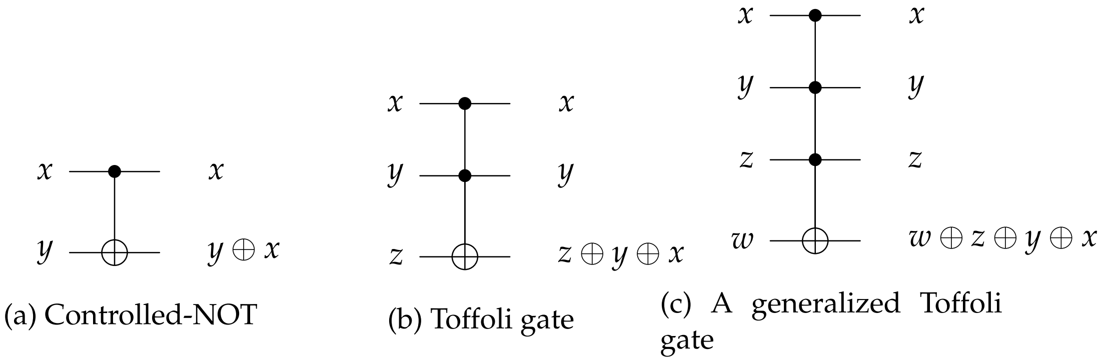
$$
R_{\hat{n}}(\theta) = e^{-\iota\theta\frac{\langle\hat{n}|\bar{\sigma}\rangle}{2}} = \cos\frac{\theta}{2}\mathrm{I} - \iota\sin\frac{\theta}{2}(x\mathrm{X} + y\mathrm{Y} + z\mathbf{Z}),
$$

where $\bar{\sigma} = (\mathrm{X}, \mathrm{Y}, \mathrm{Z})$. An arbitrary one-qubit gate has a phase difference from the general rotation operator, i.e.,

$$
U = e^{\iota\alpha}R_{\hat{n}}(\theta), \text{ for some real } \alpha \text{ and } \theta.
$$

Each quantum gate is represented by a capital letter inside a square box, and the number of quantum wires entering into the gate is equal to the leaving wires; see Fig. B.1 where the symbol $-\boxed{\mathrm{X}}-$ represents the Pauli X gate. The X gate is the quantum analog of classical NOT gate that sends a quantum state $|x\rangle$ to $|x \oplus 1\rangle$, where $x \in \{0, 1\}$.

There are operations that one-qubit gates can not perform, such as controlled-NOT (CNOT) operation, in which $|y\rangle\,|x\rangle$ is sent to $|y \oplus x\rangle\,|x\rangle$. The operator is denoted by $C_X(x, y))$, where x is the control qubit and y is the target qubit (see Fig. B.2). A generalization to this operation is the generalized Toffoli gates, which are denoted by $C_X(c, t)$, where c is the set of control qubits, and t is the target. Another generalization to controlled-NOT gate is controlled-U gate, where U is any unitary gate acting on n qubits. So, there is exactly one control qubit and more than one target qubits in a controlled-U gate. Using CNOT gate we develop an useful operation of swapping the qubits, i.e., $|y\rangle\,|x\rangle$ is sent to $|x\rangle\,|y\rangle$; the operator is denoted by $SWAP(x, y)$. The SWAP gate is constructed using three controlled-NOT gates (see Fig. B.3).

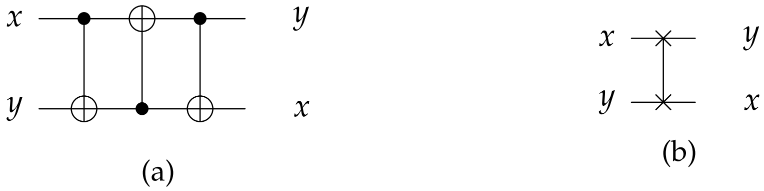


(a) Controlled-NOT

(b) Toffoli gate

(c) A generalized Toffoli gate

Figure B.2: Generalized Toffoli gates.



(a)

(b)

Figure B.3: (a) SWAP gate. (b) Notation for the SWAP gate used in quantum circuit.

List of Publications

- **Journal publication**

  1. J. Mulherkar, R. Rajdeepak, and V. Sunitha. Implementation of quantum hitting times of cubelike graphs on IBM's Qiskit platform. *International Journal of Quantum Information*, Vol. 20, No. 07, 2250020 (2022).

- **Conference publications**

  1. J. Mulherkar, R. Rajdeepak, and V. Sunitha. Quantum simulation of perfect state transfer on weighted cubelike graphs. *International Conference on Mathematics and Computing*. Vellore Institute of Technology, Vellore, India (January 6-8, 2022).
     (This article is to be printed in the book series Springer Proceedings in Mathematics and Statistics.)

  2. J. Mulherkar, R. Rajdeepak, and V. Sunitha. Perfect state transfer in weighted cubelike graphs. *International Conference on Discrete Mathematics*. Manonmaniam Sundaranar University (MSU), Tirunelveli, India (October 11-13, 2021).

  3. R. Rajdeepak, and V. Sunitha. Embedding perfectly balanced 2-caterpillar into its optimal hypercube. *Conference on Graphs, Networks, and their Applications*. Moscow Institute of Physics and Technology (MIPT), Moscow, Russia (May 13-15, 2019).