Quantum Search Algorithms

Birgit Hein, Dipl.-Phys.

Thesis submitted to The University of Nottingham

for the degree of Doctor of Philosophy

Feburary 2010

Abstract

In this thesis two quantum search algorithms on two different graphs, a hypercube and a d-dimensional square lattice, are analysed and some applications of the lattice search are discussed.

The approach in this thesis generalises a picture drawn by Shenvi, Kempe and Whaley, which was later adapted by Ambainis, Kempe and Rivosh. It defines a one parameter family of unitary operators U_λ with parameter λ. It will be shown that two eigenvalues of U_λ form an avoided crossing at the λ-value where U_λ is equal to the old search operator.

This generalised picture opens the way for a construction of two approximate eigenvectors at the crossing and gives rise to a 2×2 model Hamiltonian that is used to approximate the operator U_λ near the crossing. The thus defined Hamiltonian can be used to calculate the leading order of search time and success probability for the search. To the best of my knowledge only the scaling of these quantities has been known.

For the algorithm searching the regular lattice, a generalisation of the model Hamiltonian for m target vertices is constructed. This algorithm can be used to send a signal from one vertex of the graph to a set of vertices. The signal is transmitted between these vertices exclusively and is localised only on the sender and the receiving vertices while the probability to measure the signal at one of the remaining vertices is significantly

smaller.

However, this effect can be used to introduce an additional sender to search settings and send a continuous signal to all target vertices where the signal will localise. This effect is an improvement compared to the original search algorithm as it does not need to know the number of target vertices.

Publications

Parts of this work have or will be published in

[1]: B. Hein and G. Tanner. Quantum search algorithms on the hypercube. *J. Phys. A*, **42**: 085303, 2009,

[2]: B. Hein and G. Tanner. Wave communications across regular lattices. *Phys. Rev. Lett.*, **103**: 260501, 2009,

[3]: B. Hein and G. Tanner. Quantum search algorithms on a regular lattice. Accepted for Publication in Phys. Rev. A, Jun. 2010.

Acknowledgements

First of all, I would like to thank my parents, my brother and my sister who always supported me on my way.

Then, I would like to thank my supervisor Gregor Tanner for many interesting discussions, his support and for always having an open door for me to drop in when I had problems.

I also would like to thank the Complex and Disordered Systems group in Nottingham for providing an inspiring environment and interesting discussions in our seminar. Additionally I want to thank my former and present office mates for a good, friendly and productive atmosphere.

And not to forget the school's administrative staff, especially Andrea Blackbourn and Helen Cunliffe.

Furthermore, I have been able to contribute to several conferences and seminars and have participated in many interesting discussions. This helped me to view my project from different angles and to look at questions I would not have looked at otherwise. Thank you to everyone who made this possible.

Contents

CHAPTER 1

Introduction

1.1 Examples for classical search algorithms

Search algorithms are a tool frequently used in every day life. In general, there are two different types of search algorithms depending on the structure of the problem, that is, if the underlying database is sorted or not. It is easy to find a name in a telephone book because all entries are sorted in an alphabetical order, but if that would not be the case? Imagine the entries are in some random order. What is the most efficient way to find a telephone number? It would be necessary to go through all entries until the demanded person is found. Obviously, $\frac{N}{2}$ entries would have to be checked on average if N is the number of entries in the book.

There are numerous search problems, the randomly ordered telephone book being just one example. The probably most prominent example is the travelling salesman problem which has a long standing history starting in the 19th century and probably first discussed by Kirkman and Hamilton (see [4] on the history): a salesman has to visit a given number of towns exactly once in no particular order and return to his hometown. The search problem is to identify the shortest among all possible routes. In its most

abstract form, the towns can be considered as being the vertices of a fully connected metric graph and the distances between the towns are identified with the lengths of the bonds connecting the vertices. Now the solution of the problem is the shortest cycle on the graph that visits all vertices once.

A third example is the so called class of Boolean satisfiability problems, short k-SAT [5]. The problem asks for a d-digit string of Boolean variables $(x_1,\ x_2,\ x_3, \ldots, x_d)$ that fulfils a set of given clauses. Each clause consists of k Boolean variables that are connected by the logical OR. The set of clauses is connected by the logical AND operation, that is, every clause has to have at least one Boolean variable equal $TRUE$ for a Boolean string $(x_1, \ldots, x_d)$ to be recognised as solution. An example for 3-SAT is

$$E = (x_1\ OR\ x_2\ OR\ \neg x_3)\ \ AND\ \ (x_1\ OR\ \neg x_2\ OR\ x_4)\ AND \ldots, \tag{1.1.1}$$

where $\neg$ is the logical NOT operation.

k-SAT can be solved using a random walk on a hypercube since the set of Boolean strings with d entries can be identified with the set of vertices of a d-dimensional hypercube as both sets consists of d-digit strings containing 0's and 1's. The solution can be found using the following algorithm:

1. start with a random configuration for $\vec{x}$
2. check $E(\vec{x})$:
   - if $E(\vec{x})$ is true: give $\vec{x}$ as solution and stop
   - if $E(\vec{x})$ is false, proceed to step 3
3. choose a random integer i in $\{1, \ldots, d\}$
4. change the ith entry of $\vec{x}$ and return to step 2 .

This is just an example of a search algorithm that finds a solution for the k-SAT Problem, it is clear that the algorithm will be faster, if it keeps track of the vertices that have been

visited previously such that each vertex $\vec{x}$ is not visited more than once. Apart from that the algorithm described above can be improved in several other ways but it served well as example.

In 2002, Braich *et al.* solved a 20 variable 3-SAT Problem on a DNA Computer [6].

1.2 Overview

1.2.1 Quantum random walks

In 1993 Aharonov, Davidovich and Zagury introduced a one dimensional model for a quantum random walk [7]. This model considers a quantum particle, that is, a particle described by a wave function, performing a random walk. It is shifted a distance l either to the left or to the right depending on whether its spin state is up or down. After each shift the spin state can be arranged in a superposition of up and down. If both operations are applied alternately, the particle will spread out on the line. A measurement of the position after each shift operation leads to the quantum particle performing a classical random walk. However, if the position is not measured the particle performs a so called 'quantum random walk' and the average distance from the starting position will increase faster than for classical random walks (see [7, 8, 9] or section 2.2.2). This clear difference from the behaviour of a classical random walk occurs due to destructive interference near the starting position.

In general, there exits two distinct classes of quantum random walks, continuous and discrete time quantum random walks. An introductory overview on quantum random walks, which also highlights the difference between discrete time and continuous time quantum random walks, can be found in [9]. For the discrete time quantum random walk the position space can also be regarded as discretised. This leads to a position space

with a lattice structure which can be considered as graph. These graphs underlying the quantum random walks are highly regular and have the same bond length for all bonds. In particular it is interesting to outline the differences between classical and quantum random walks. As calculated by Aharonov, Davidovich and Zagury, quantum random walks are able to spread faster through a system [7, 9, 10]. Aharonov, et al. [10] also showed, that the quantum speed up of random walks is at most polynomial. For a classical 1-dimensional random walk the average distance from the starting position increases with $\sqrt{T}$, where T is the number of applications of the shift operation. Notably, the propagation of a quantum random walk on a line is essentially ballistic, that is, the average distance is proportional T.

Quantum random walks play also an important role in quantum computation. In 1998, Farhi and Gutmann presented an analysis using continuous time quantum random walks to construct a quantum algorithm for decision trees. They showed that if a classical algorithm needs a time polynomial in the size of the problem, the quantum algorithm presented in their paper is as efficient. On the other hand an example has been introduced for which the classical algorithm needs an exponential time whereas the quantum algorithm succeeds again in polynomial time.

One can also define the notion of a 'hitting time', that is, the time that is needed to propagate from one vertex of a graph to another one. The result in this setting is again that quantum random walks are significantly faster than classical [11, 12].

Disorder in the graph underlying the quantum random walk has been investigated in [13] and a review on quantum walks with decoherence can be found in [14].

Some general reports on the usage of quantum random walks for quantum algorithms have been presented by Kendon [15] or Santha [16]. Quite recently, a few reports on experimental realisations of quantum random walks have been published [17, 18, 19, 20,

21].

In this thesis, only discrete time quantum random walks are discussed. The connection to quantum graphs is highlighted in [22].

1.2.2 Quantum search algorithms

It has been shown by Grover [23] that a quantum search algorithm may find its target among N database entries in a time $\mathcal{O}\left(\sqrt{N}\right)$ whereas a classical algorithm takes a time $\mathcal{O}\left(N\right)$ as in the example of the random telephone book above. It is important to note that Grover's algorithm solves the problem in a time that scales optimal in N as it is known that quantum search algorithms can not scale faster in the number N of database entries than $\sqrt{N}$ [23, 24].

A somewhat related class of quantum search algorithms are search algorithms on a spacial structure, that is on a graph, based on quantum random walks on these graphs. Both classes of quantum random walks, lead to a class of quantum search algorithms.

In 2003 Shenvi, Kempe and Whaley discussed a quantum algorithm searching a d-dimensional hypercube and two years later, Ambainis, Kempe and Rivosh introduced a quantum search algorithm on a d-dimensional square lattice [25]. These algorithms have both some analogies to Grover's search algorithm [23], and are both based on a discrete time quantum random walk. The only difference to a pure quantum random walk is that the target of the search is 'marked' in some way which leads to a local perturbation of the quantum random walk on the underlying graph and have some analogies to Grover's algorithm. They succeed to find the target vertex in a time of order $\mathcal{O}\left(\sqrt{N}\right)$ (apart from the search on the 2 dimensional lattice that scales like $\mathcal{O}\left(\sqrt{N}\ln N\right)$).

On the other hand, a search algorithm on a d-dimensional lattice based on a continuous time quantum random walk, has been investigated by Childs and Goldstone [26]. Using

position space only, the algorithm finds the target in a time in time $\sqrt{N}$ for $d > 4$ and $\mathcal{O}\left(\sqrt{N}\ln N\right)$ in four dimensions. In a second report [27], Childs and Goldstone added a spin degree of freedom and used a lattice version of the Dirac Hamiltonian and found a speed up for lower dimensions. This second search algorithm scales like $\sqrt{N}$ time steps for $d > 2$ and $\mathcal{O}\left(\sqrt{N}\ln N\right)$ in two dimensions.

Just last year, Potoček *et al.* have shown that the search on the hypercube can be improved by a factor $\frac{1}{\sqrt{2}}$ [28] by introducing an auxiliary dimension.

1.2.3 Computer science

As quantum search algorithms are somewhat related to the topics of quantum computation, a short introduction on computer science is added to this introductory chapter:

Modern computer science started in 1936 when Turing developed a model for an abstract computer, now called 'Turing machine' [29]. The Turing machine was constructed to do computations and can be considered the first model of a programmable computer. Turing claimed that this machine is able to calculate any algorithm that can be solved on any hardware, which includes a modern computer or a mathematician.

In computer science it is often interesting to analyse the complexity of a given problem. In the example of the random telephone book mentioned above, the classical algorithm takes $\mathcal{O}(N)$ time steps to find the demanded entry while Grover's algorithm succeeds in only $\mathcal{O}\left(\sqrt{N}\right)$ time steps.

Since the first suggestion of quantum computers by Feynman in 1982 [30], research on quantum computers has risen many interesting questions, see e.g.[5] for an introduction Two years after Feynman's proposal, Deutsch suggested a model for a universal quantum computer, a quantum generalisation of the Turing machine [31]. It has then been shown that quantum computers are at least as powerful as classical computers.

This is no surprise since at their deepest level, classical computers are ruled by the laws of quantum mechanics. The inverse problem is much more interesting: Are quantum computers more powerful than their classical counterparts? In 1992 Deutsch and Jozsa have shown that this is indeed the case [32].

The most prominent examples for efficient quantum algorithms is Shor's algorithm for prime factorisation of an integer N that succeeds in a time polynomial in $\log N$ [33], and Grover's algorithm [23] for the search of an unsorted database. A realisation of Shor's algorithm in an photonic chip has been investigated by Politi, Matthews and O'Brien [34].

1.2.4 Quantum graphs

It might also be interesting to give a short review of the related but not central field of quantum graphs.

Not unlike graphs in graph theory, quantum graphs consist of a set of vertices and a set of one dimensional bonds connecting these. These structures itself are not quantum but one can imagine a quantum particle or wave package propagating through the system. The propagation on the bonds is a solution of the one dimensional Schrödinger equation and the propagating particle or wave is scattered whenever it reaches a vertex. Introductions on quantum graphs can be found in many reports like [22, 35, 36, 37, 38].

The first application of quantum graphs in physics has been done in 1936 by Pauling who constructed a model for the free electrons in hydrocarbons [39].

More than 10 years ago Kottos and Smilansky [37, 40] discovered that quantum graphs can also be used as a model for classically chaotic systems if the bond lengths are chosen rationally independent, see also [41]. This opened a new approach to the study of quantum behaviour of classically chaotic systems.

In recent years, the study of quantum graphs has become a field with many applications. Just recently, Schapotschnikow and Gnutzmann used Pauling's ansatz for the calculations of the spectra of larger molecules [42] but quantum graphs can also be used calculate limiting behaviour for wave guides in the limit where the widths of the channels is small compared to any other lengths scale [43], transport through networks [44], localisation effects [45], quantum decay in open chaotic systems [46, 47] or quantum ergodicity [48].

Quantum graphs have also been simulated experimentally [49].

1.2.5 Notations regarding the growth and decay of a function

Throughout this thesis the computer science notations to characterise the growth or decay of a function are used. $f(x) = \mathcal{O}(g(x))$ denotes that there exist two positive constants $x_0 > 0$ and $a > 0$, such that for all $x > x_0$ the inequality $0 \leq f(x) \leq ag(x)$ is true. Similarly $f(x) = \Omega(g(x))$ will be written if $0 \leq bg(x) \leq f(x)$ for all $x > x_0$ for some constants $x_0,\ b > 0$. Furthermore, $f(x) = \Theta(g(x))$ denotes that $f(x) = \mathcal{O}(g(x))$ as well as $f(x) = \Omega(g(x))$.

1.3 Structure and agenda

Chapter 2 starts with more detailed definitions of the underlying graphs, quantum random walks and search algorithms and collects all fundamental definitions used in this thesis.

In the following chapter, chapter 3, a search algorithm on the hypercube is introduced and the relevant operators are defined in more detail. In the scope of this chapter a new 2×2 model of the search operator is constructed. This model will be explained in detail

and used to calculate the first order of localisation amplitude and of the time necessary for the localisation effects. In chapter 4 the model outlined in the preceding chapter is again applied to analyse a search algorithm on a regular d-dimensional lattice. Again the leading order contributions of localisation time and amplitude at the target vertex are calculated.

Some applications of the search algorithm on the lattice as introduced in chapter 4 will be presented in chapter 5. The algorithm will in particular be used to search for more than one target and it will be discussed how this mechanism can be applied to send a signal though the graph. This setting can also be used to improve both, the lattice search and Grover's search algorithm. Additional to that, some ideas to observe the localisation effect in an experimental situation will be presented. This chapter is followed by some thoughts about why the search algorithm leads to a localisation and which properties might play a role in the localisation process in chapter 6.

In chapter 7 a summarisation of the results is presented and some open questions are discussed.

CHAPTER 2

Preliminaries and Definitions

2.1 The underlying graph

Figure 2.1 shows the sketch of a graph with 12 vertices and 19 bonds. In their most general form they can be considered as networks consisting of vertices v connected by bonds b, where each bond b_i has a length l_{b_i} that is assigned to it.

Let V and B be the sets of vertices and bonds and let $|B|$ and $|V|$ be their corresponding cardinal numbers. For simplicity only connected graphs are considered, that is, graphs which can not be divided into subgraphs without removing one or more bonds. Furthermore it will be assumed that for every pair of vertices i and j, there will be at most one

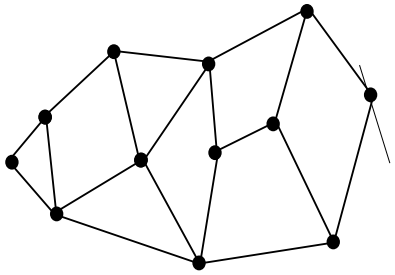

Figure 2.1: Sketch of a graph.

bond connecting these for undirected bonds and at most one bond in each direction for directed bonds. Directed bonds allow transition only in one direction while undirected bonds treat both directions equally, this can be used to model systems with or without time reversal symmetry.

Using this restrictions the ordered pair (i, j) uniquely defines the bond leading from vertex i to vertex j. For undirected bonds the lengths l_{ij} and l_{ji} are equal.

A graph is called connected if it can not be divided into subgraphs without removing some of its bonds. In other words, each vertex of a connected graph can be reached from any other vertex by passing through at most $|B|$ bonds.

The quantum dynamics on such a graphs is defined by the Schrödinger equations for the bonds and the solution are propagating waves. Continuity of the wave function at the vertices demands that the wave function to be continuous: the wave functions on all bonds leaving vertex i have the same limiting behaviour and approach the same value.

Let now k_i be the number of incoming bonds at vertex i and l_i be the number of outgoing bonds. For simplicity $k_i = l_i$ will be assumed at all vertices. Then the mapping of incoming onto outgoing waves at i can be described using a $k_i \times k_i$ matrix σ^i which is called vertex scattering matrix.

Wave propagation on graphs has been discussed in [35, 36, 37, 38].

Following the line of [25, 50], the analysis in this thesis will be restricted to connected k-regular graphs with undirected bonds. A k-regular graph is a graph, where all vertices have the same number of outgoing bonds k. Furthermore all bonds of the graph will have the same bond length which will, for convenience, be normalised to $l_{ij} = 1$ for all (i, j). Additional to that the vertex scattering matrices will be chosen identically and be denoted as $\sigma = \sigma^i$ for all i. Later a special vertex scattering matrix will be introduced at a small number of vertices denoted by σ'.

In particular only two different types of graphs are regarded in this thesis. A d-dimensional hypercube, that is, a cube in d dimensions and a d-dimensional square lattice with n^d vertices and periodic boundary conditions.

2.2 Quantum random walks

Now imagine a classical particle performing a random walk on a graph. When the walker is started on an arbitrary vertex, one of the k outgoing bonds is chosen randomly, say by throwing a dice or flipping a coin. Then the walker proceeds through this bond to the next vertex, where again one out of k bonds is chosen for the next iteration. The probability distribution of the position of the walker spreads out over the graph as time increases.

Kempe presented an introductory overview on quantum walks in [9] and discussed the difference of continuous and discrete time quantum random walks in detail. However, only the discrete time quantum random walk will be discussed here.

Since the propagation of the quantum random walk can be described by unitary operators, the quantum random walk is not random but deterministic, the name 'quantum random walk' is used, to stress the analogy to a classical random walk. However, if the position of the quantum walker is measured after every step, it will perform a classical random walk on the graph, whereas if the position is not measured, the walker will remain in a superposition of many positions and interferes with itself.

2.2.1 The abstract quantum random walk

Quantum random walks on graphs are defined in analogy to classical random walks on classical graphs. The quantum walker is positioned somewhere on the graph and

then a unitary operator σ, called 'coin flip', performs a transformation of an internal degree of freedom. The space corresponding to this internal degree of freedom is called 'coin space' and the state can e.g. be imagined as spin state. Since the superposition of outgoing bonds is chosen according to the walkers coin space state, the coin flip takes the rôle of a vertex scattering matrix as described in 2.1.

In a second manipulation, the walker is then propagated to the next vertex that is chosen according to the walkers state in the coin space by a 'shift operator' S. Since the coin space state is typically a superposition of basis states, the waker is shifted into a superposition of several vertices.

Let $|i\rangle$ be the coin space state of the incoming quantum walker at vertex $|v\rangle$, then the overall state can be denoted as $|i\rangle \otimes |v\rangle$. First the coin flip σ is applied and the state $(\sigma |i\rangle) \otimes |v\rangle$ is obtained before the shift operator S shifts the walker to $S(\sigma |i\rangle) \otimes |v\rangle$.

The main effect of S can be denoted as a permutations in position space as it changes the position of the walker, but it may also perform a permutation in coin space. However, S has to be defined such that it does not violate the underlying graph by shifting the walker to a new vertex that is not connected to the first one by a bond.

The N-dimensional position space and the k-dimensional coin space define a kN dimensional Hilbert space for the quantum random walk.

2.2.2 A 1-dimensional example

To provide some understanding of quantum random walks a quantum random walk on a 1-dimensional lattice with periodic boundary conditions will be introduced. Figure 2.2 shows a sketch of such a graph and provides a numbering for the vertices. It will be assumed that there are n vertices in both directions, with n being large compared to the number of time steps in the quantum random walk. Furthermore the vertices $|n\rangle$ and

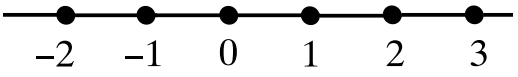

Figure 2.2: Sketch of a 1-dimensional line lattice.

$|-n\rangle$ are identified due to the periodic boundary conditions. Thus the overall number of vertices of the lattice is $N = 2n$.

Let the quantum random walk be started at the vertex $|0\rangle$ and let the two coin space states be denoted as $|\rightarrow\rangle = (1,0)^{\mathsf{T}}$ and $|\leftarrow\rangle = (0,1)^{\mathsf{T}}$. A typical state of the system will be denoted as $|iv\rangle = |i\rangle \otimes |v\rangle$, where $|i\rangle$ is a coin space state and $|v\rangle$ a state in position space.

For this example the coin

$$\sigma = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & \mathrm{i} \\ \mathrm{i} & 1 \end{pmatrix} \tag{2.2.1}$$

and the shift operator

$$S = \sum_{i=-\infty}^{\infty} \left(|\rightarrow i+1\rangle \langle \rightarrow i| + |\leftarrow i-1\rangle \langle \leftarrow i| \right) \tag{2.2.2}$$

will be used. Both operators are taken from [9]. Note that the coin flip does not distinguish between the two different coin state states but treats both directions equally.

Both operators can also be interpreted to define the wave propagation on the graph. σ takes the rôle of the vertex scattering matrix: It defines how much of the incoming amplitude is reflected and how much passes through the vertex and adds a phase factor to both outgoing waves. S is the bond propagation matrix. Note that bond lengths and wave vector have been chosen such that the phase factor due to bond propagation is 1.

Since σ acts only locally on the coin space, the tensor product and the identity operator in position space are used to define a global coin flip

$$C := \sigma \otimes \mathbb{1}_N \tag{2.2.3}$$

$\|v\rangle$	-4	-3	-2	-1	0	1	2	3	4
T=0					1				
T=1				$\frac{1}{2}$		$\frac{1}{2}$			
T=2			$\frac{1}{4}$		$\frac{1}{2}$		$\frac{1}{4}$		
T=3		$\frac{1}{8}$		$\frac{3}{8}$		$\frac{3}{8}$		$\frac{1}{8}$	
T=4	$\frac{1}{16}$		$\frac{3}{8}$		$\frac{1}{8}$		$\frac{3}{8}$		$\frac{1}{16}$

Table 2.1: Probabilities for the quantum random walk.

that flips the coin of the quantum walker at all vertices simultaneously.

The quantum random walk is now defined by an application of C followed by an application of S, that is,

$$U = SC. \tag{2.2.4}$$

Let the walk be started in the state

$$|\text{start}\rangle := \frac{1}{\sqrt{2}} \left(|\rightarrow 0\rangle + |\leftarrow 0\rangle \right). \tag{2.2.5}$$

After one time step, that is one application of U, the state results in

$$U\,|\text{start}\rangle = \frac{1+\mathrm{i}}{2} \left(|\rightarrow 1\rangle + |\leftarrow -1\rangle \right) \tag{2.2.6}$$

and the probability to measure the walk at $|1\rangle$ or $|-1\rangle$ is $\frac{1}{2}$ each.

For the first four steps, the probability of the quantum random walk evolves as listed in table 2.1, whereas a classical random walk with probability $\frac{1}{2}$ of stepping right or left evolves as shown in table 2.2. A clear difference between both random walks can be noted at $T = 4$. Due to interference effects, the quantum walk as a significantly smaller probability of returning to $|0\rangle$ compared to the classical random walk.

Ambainis *et al.* [8] presented a detailed analysis of the difference between classical and quantum random walks on a line. They also introduced absorbing boundaries to the

$\|v\rangle$	-4	-3	-2	-1	0	1	2	3	4
T=0					1				
T=1				$\frac{1}{2}$		$\frac{1}{2}$			
T=2			$\frac{1}{4}$		$\frac{1}{2}$		$\frac{1}{4}$		
T=3		$\frac{1}{8}$		$\frac{3}{8}$		$\frac{3}{8}$		$\frac{1}{8}$	
T=4	$\frac{1}{16}$		$\frac{1}{4}$		$\frac{3}{8}$		$\frac{1}{4}$		$\frac{1}{16}$

Table 2.2: Probabilities for the classical random walk.

process and found that the probability of the walk being absorbed by an absorbing boundary directly left to the starting point is $\frac{2}{\pi}$, where in the classical case it is 1. They carried this investigation a step further, and discovered that the probability increases to $\frac{1}{\sqrt{2}}$ if there is a second absorbing boundary a large distance of m vertices away on the other side. Classically, one would expect the probability to be absorbed on the left side decreases due to the second boundary.

2.2.3 Continuous time quantum random walk

Although this thesis analyses quantum search algorithm constructed from discrete time quantum random walks, a short introduction to continuous time quantum random walks will be provided. Overviews and the relation to discrete time quantum random walks or classical random walks have been presented in [9, 11, 51].

The continuous time quantum random walk takes place in position space only. That is, no additional coin space is needed and no coin is flipped. This class of quantum random walks are a quantum version of a classical continuous time Markov chain [52]. To obtain a continuous time random walk, one classically, defines an infinitesimal generating

matrix H using a jumping rate γ for transitions from one vertex to one of its neighbours:

$$H_{ij} = \begin{cases} -\gamma & \text{if } i \neq j \text{ and } i \text{ and } j \text{ are connected} \\ d_i\gamma & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}, \tag{2.2.7}$$

where d_i is the number of vertices connected to i.

Let now $\vec{p}\,(t)$ be the probability distribution at time t, then the random walk will obey the differential equation

$$\frac{d}{dt}\vec{p}\,(t) = -H\vec{p}\,(t)\,. \tag{2.2.8}$$

This gives rise to the solution $\vec{p}\,(t) = \exp(-Ht)\,\vec{p}\,(0)$.

To obtain a quantum process from this classical random walk, Farhi and Gutmann [52] proposed that the matrix H now becomes a Hamiltonian and the time evolution is defined by the propagator $U(t) = \exp(-iHt/\hbar)$.

Overall, the main difference between the discrete and continuous time quantum random walk is that the continuous time random walk takes place in position space only whereas the discrete time quantum random walk needs an additional coin space. Although there are differences, both models for quantum random walks have some similarities as the fast spreading on a line [52] as discussed above for the discrete time model. Detailed reports on the similarities and differences may be found in [9, 51]. The main advantage of the continuous time quantum walk is the smaller Hilbert space which simplifies the analysis. On the other hand, for the continuous time model it is harder to implement quantum algorithms for graphs if the maximum degree of the graph is not small [51, 53].

Both quantum random walks can be used as a foundation for a quantum search algorithm [25, 26, 27, 50]. However, for low dimensional lattices the continuous time search algorithm with no additional auxiliary space is not as efficient as the discrete time search [26, 27].

2.3 Quantum search algorithms

Quantum search algorithms are able to solve search problems that have one or more solutions. For simplicity only the case for one solution will be discussed here. Given a system of N states that are labelled $v_1, v_2, \ldots, v_N$. Let there be a unique yet unknown target state, say $|v_t\rangle$ that satisfies a given condition C implemented by a function $C(v)$ such that $C(v_t) = 1$ and $C(v_i) = 0$ for all states $|v_i\rangle \neq |v_t\rangle$.

The most prominent quantum search algorithm is Grover's search algorithm. It provides the means to identify the state v_t in a time $\mathcal{O}\left(\sqrt{N}\right)$ [5, 23, 54]. A report on an experimental realisation of Grover's algorithm can be found in [55].

In general, the function implementing condition C is often referred to as 'oracle' and treated as black box since the knowledge of C is not important. The only purpose of C is to recognise the target state $|v_t\rangle$ among all possible input states $|v_j\rangle$ by returning a unique result if questioned for $|v_t\rangle$. An equivalent problem to searching for $|v_t\rangle$, is to invert the unknown function C since the target state is the unique state satisfying $v_t = C^{-1}(1)$.

2.3.1 Grover's search algorithm

The search algorithm for N states introduced by Grover in [23] is implemented by the following rules:

1. initialise the system in the uniform distribution

$$|\psi\rangle := \left(\frac{1}{\sqrt{N}}, \frac{1}{\sqrt{N}}, \frac{1}{\sqrt{N}}, \ldots, \frac{1}{\sqrt{N}}\right)^{\mathsf{T}} \tag{2.3.1}$$

2. repeat the following unitary operations T times, where T is a particular, but yet unknown time $T = \Theta\left(\sqrt{N}\right)$

   (a) rotate the phase of all states $|v_i\rangle$ by $\pi C(v_i)$

   (b) apply the unitary matrix D defined by $D_{ii} = -1 + \frac{2}{N}$ and $D_{ij} = \frac{2}{N}$ for $i \neq j$

3. measure the resulting state $|\psi_{\text{result}}\rangle$.

Grover has shown that the outcome of the search has a notable overlap with the target state $|v_t\rangle$ singled out by $C(v_t) = 1$. The probability to find $|\psi_{\text{result}}\rangle$ in $|v_t\rangle$ is at least $\frac{1}{2}$.

A sketch of the proof is given below. A more detailed discussion will be postponed until section 5.4, an interested reader can also find proofs in [5, 23].

Nielsen and Chuang discussed a geometric approach for Grover's search algorithm in [5]: The steps (2a) and (2b) of the algorithm map the 2-dimensional subspace spanned by $|v_t\rangle$ and $|\psi\rangle$ onto itself. Step (2a) performs a reflection about a vector $|\alpha\rangle$ that lies in this subspace and is orthogonal to $|v_t\rangle$. Similarly $D = 2|\psi\rangle\langle\psi| - \mathbb{1}_N$ can be regarded as reflection about $|\psi\rangle$ and as the starting state lies in this subspace, the state is trapped there.

Both reflections applied in succession perform a rotation about an angle θ that is twice the angle between $|\alpha\rangle$ and $|\psi\rangle$. Therefore $\sin\frac{\theta}{2} = \frac{1}{\sqrt{N}}$ and after $T = \Theta\left(\sqrt{N}\right)$ cycles over the loop of steps (2a) and (2b) the distance of $|\psi_{\text{result}}\rangle$ to $|v_t\rangle$ is minimal. If the rotation is continued once the time T has passed, the distance will increase again as the state is rotated away from $|v_t\rangle$.

The quantum search algorithms discussed in this thesis are related to Grover's algorithm in a sense that the algorithms can be interpreted as rotations in a two dimensional subspace spanned by the uniform distribution which will be used as starting state and a localised state that is close to the target state. Further details will be discussed in section 3.4.3.

2.3.2 The abstract search algorithm

To employ a quantum random walk as search algorithm, the underlying database of the search has to be mapped on the vertices of the graph and an oracle is used to identify the target vertex, that is, the vertex to which the target entry of the database has been mapped. For simplicity, only locally indistinguishable graphs will be discussed, that is, graphs where all vertices have the same number of bonds k and the same coin flip operation is performed at all vertices. Furthermore, all bonds are assumed to be of unit lengths, such that the same phase factor is added due to the propagation along the bonds which will be chosen to be 1. In addition to that, the coin flip will treat all bonds equally and only backscattering may appear with a different probability than propagation in any other direction.

Let S be the moving shift and σ the local coin flip matrix that is applied at all vertices. A global coin flip that flips the coin on all vertices simultaneously is defined by $C = \sigma \otimes \mathbb{1}_N$, where $\otimes$ denotes the tensor product and $\mathbb{1}_N$ the identity operator in position space. Since there are k outgoing bonds σ is a $k \times k$-matrix and the Hilbert space of the problem is kN-dimensional.

The quantum random walk U is defined by first applying the global coin flip and then the moving shift

$$U = SC. \tag{2.3.2}$$

The oracle o is defined such that it first scans the database entry mapped to vertex $|v_i\rangle$ and then returns 0 whenever the entry is not the target entry of the search whereas $o|v_t\rangle = 1$ if ENTRY (v_t) is the target. Now, the return value of the oracle can be used to mark $|v_t\rangle$ by choosing local coin matrices that depend on the oracle output such that σ is applied on all unmarked vertices whereas a different coin flip is applied at the target vertex $|v_t\rangle$.

Following this line of thought, the abstract search algorithm is defined as

$$U' = SC - (\sigma - \sigma') \otimes |v_t\rangle \langle v_t| , \tag{2.3.3}$$

where σ' is the marking coin applied at the target vertex only.

As shown in (2.3.3), the term that is added to U has the form $-(\sigma - \sigma') \otimes |v_t\rangle \langle v_t|$. Since $|v_t\rangle \langle v_t|$ is a projection on the target vertex, the additional term changes at most k^2 matrix elements. As U is a $kN \times kN$-matrix, U' can be regarded as, up to a small perturbation, identical to U. Thus, the abstract quantum search algorithm as described in this section can be regarded as perturbed quantum random walks.

CHAPTER 3

Search algorithm on the hypercube

This chapter starts with an introduction of a search algorithm as introduced by Shenvi, Kempe and Whaley in [50]. The algorithm is constructed to search the graph of a hypercube for a given but unknown target vertex. In this search model, the target vertex acts as a local perturbation in an otherwise highly symmetric quantum random walk. In the previous analysis [50], Shenvi *et al.* gave an analytical estimate the search time and the localisation probability.

The approach used by Shenvi *et al.* gives an estimate of the eigenvalues and eigenvectors of the search operator. For the calculation presented here, only the eigenvalues of the quantum random walk are needed which have been calculated in [56]. The model introduced in this chapter defines a one parameter family of unitary operators U_λ (3.2.1). With increasing λ, this family changes continuously from the unperturbed quantum random walk to the search operator and back. This new picture reveals avoided crossings in the spectrum as will be seen in figure 3.4.

The thus introduced family of unitary operators is then analysed in terms of eigenvectors and eigenvalues. It is argued that only a few entries of the search operator change as a function of the parameter λ and the eigenvectors of the unperturbed system form a convenient basis for the calculation of eigenvectors. However, it is argued that the search algorithm is singled out in the family of unitary operators by an avoided crossing visible when the spectrum is plotted as a function of λ. The search algorithm is then characterised by the eigenvectors and eigenvalues near the crossing at $\lambda = 1$.

The two eigenvectors at the crossing are calculated approximately and the approximate eigenvalue equations are given in (3.3.2) and (3.3.10). The subspace spanned by these two approximate eigenvectors is almost invariant under the action of the search operator and it is shown that, while one approximate eigenvector acts as starting state, the second is localised at the target vertex.

Therefore it is argued, that the search operator can be analysed in the two dimensional subspace spanned by these vectors using a simple model Hamiltonian for the avoided crossing in equation (3.4.1). This leads finally to an expression for the localisation probability (3.6.3) and the search time (3.6.2). The result for the latter is found to agree with the result obtained from numerical simulations of the search as presented in figure 3.5.

3.1 Definitions and model

A d-dimensional hypercube is a graph whose vertices can be encoded using binary strings with d digits containing either 0 or 1 as entry. Vertices, whose strings are equal for all but one digit are connected as shown in figure 3.1. Thus each vertex is connected to d neighbouring vertices via outgoing bonds and the dimension of the Hilbert space $\mathcal{H}$ is $\dim\mathcal{H} = dN$, where $N = 2^d$ is the number of vertices of the graph.

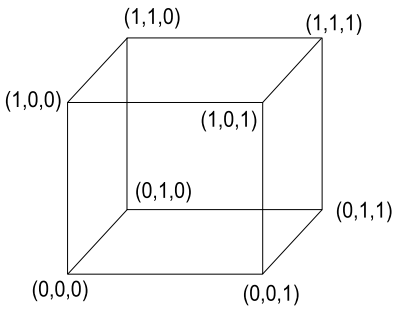

Figure 3.1: The hypercube in $d = 3$ dimensions.

The quantum search algorithm introduced by Shenvi, Kempe and Whaley in [50] will be discussed in this chapter. As outlined in section 2.3.2, quantum search algorithms can be regarded as perturbed quantum random walks. The underlying quantum random walk has been analysed by Moore and Russell [56] who calculated the full set of eigenvalues.

3.1.1 The underlying quantum random walk

In analogy to the discussion in section 2.2 the quantum random walk is ruled by a unitary operator $U = S\,(\sigma \otimes \mathbb{1}_{2^d})$, where σ is the local coin flip on all vertices, S is a moving shift and $\mathbb{1}_{2^d}$ the identity operator in position space.

The uniform distribution in coin space $|s\rangle = \frac{1}{\sqrt{d}} \sum_{i=1}^{d} |i\rangle$ is used to define the local coin flip on each vertex as

$$\sigma = 2\,|s\rangle\,\langle s| - \mathbb{1}_d. \tag{3.1.1}$$

This defines the global coin that flips the coins on all vertices simultaneously as

$$C = \sigma \otimes \mathbb{1}_{2^d}. \tag{3.1.2}$$

The dN states of the system will be denoted as $|ix\rangle = |i\rangle \otimes |x\rangle$, where $|i\rangle$ is a vector in the d-dimensional coin space and $|x\rangle$ a vector in the N-dimensional position space. Using these notations the shift operator S can be denoted conveniently. Since S transfers

the state $|ix\rangle$ into $|ix \oplus e_i\rangle$, where $|e_i\rangle$ is the unit vector in direction i and $\oplus$ is the sum modulo 2, it is defined as

$$S = \sum_{i=1}^{d} \sum_{\vec{x}} |ix \oplus e_i\rangle \langle ix| . \tag{3.1.3}$$

The eigenvalues $v_k^{\pm}$ and eigenvectors $\left|v_{\vec{k}}^{\pm}\right\rangle$ of the quantum random walk

$$U = SC \tag{3.1.4}$$

are discussed in [50, 56] and a detailed calculation can also be found in appendix A. Some for the scope of this thesis unimportant ± 1-eigenvectors will be neglected as the analysis will later be reduced to a subspace orthogonal to them. The remaining two eigenvalues and orthonormalised eigenvectors that can be found for every d-dimensional binary vector $\vec{k}$ are

$$v_k^{\pm} = \mathrm{e}^{\pm \mathrm{i}\omega_k} = 1 - \frac{2k}{d} \pm \frac{2\mathrm{i}}{d}\sqrt{k\,(d-k)} \tag{3.1.5}$$

$$|v_{\vec{k}}^{\pm}\rangle = \sum_{i=1}^{d} \sum_{\vec{x}} (-1)^{\vec{k}\cdot\vec{x}} \frac{2^{-d/2}}{\sqrt{2}} \alpha_{k_i}^{\pm} \beta_k |ix\rangle \tag{3.1.6}$$

[50, 56], appendix A.2, where $k = |\vec{k}| = \sum_{i=1}^{d} k_i$ is the Hamming weight of the vector $\vec{k}$ and

$$\alpha_{k_i}^{\pm} = \begin{cases} 1/\sqrt{k} & \text{if } k_i = 1 \\ \mp \mathrm{i}/\sqrt{d-k} & \text{if } k_i = 0 \end{cases} . \tag{3.1.7}$$

Compared to the previous calculations, a normalisation constant

$$\beta_k = \begin{cases} \sqrt{2} & \text{if } k = 0 \text{ or } k = d \\ 1 & \text{otherwise} \end{cases} \tag{3.1.8}$$

has been added.

Since the eigenvalues are a function of k, they are $\binom{d}{k}$ times degenerate. Furthermore note that for $k = 0$ and d the $\pm$ cases are equivalent. For simplicity the notation will not be altered but the $+$ cases will be regarded only.

3.1.2 The search algorithm

The quantum random walk can now be altered to create a search algorithm. A target vertex $|v\rangle$ is introduced and marked by a different choice for the coin flip that will be denoted as σ'. The perturber coin for the hypercube search is $\sigma' = -\mathbb{1}_d$ which leads to both, a simple analysis and an efficient search algorithm.

To obtain the search algorithm from the unperturbed quantum walk, the old coin σ is subtracted and the perturber coin σ' is added locally at the target vertex. This results in a perturbed global coin flip

$$C' = C - \left(\sigma - \sigma'\right) \otimes |v\rangle \langle v| = C - 2 |s\rangle \langle s| \otimes |v\rangle \langle v| . \tag{3.1.9}$$

Since $|s\rangle$ is a 1-eigenvector of σ the search algorithm $U' = SC'$ can be simplified. Defining the state $|sv\rangle := |s\rangle \otimes |v\rangle$ that is localised at the marked vertex and uniformly distributed in coin space,

$$\begin{aligned} U' &= SC' && (3.1.10)\\ &= S\left(C - 2C |s\rangle \langle s| \otimes |v\rangle \langle v|\right) && (3.1.11)\\ &= U - 2U |sv\rangle \langle sv| . && (3.1.12) \end{aligned}$$

Up to a few entries the operators U and U' are identical, in fact, an orthonormal basis such that equality holds for all but one entry can be constructed using the vectors $|sv\rangle$, $U |sv\rangle$ and $dN - 2$ other basis vectors.

Numerical simulations of the search algorithm U' verify the analytical observation by Shenvi, Kempe and Whaley [50]. The algorithm indeed acts as search algorithm and, starting from the uniform distribution, localises the quantum state at the marked vertex after some time. Figure 3.2 shows the performance on a 12-dimensional hypercube searching for the marked vertex $|v\rangle = \vec{0}$ starting in the uniformly distributed state $|\text{start}\rangle = \frac{-\mathrm{i}}{\sqrt{N}} |s\rangle \otimes \left(\sum_{\vec{x}} |x\rangle\right) = \left|v_{\vec{0}}^{+}\right\rangle$, where the sum is performed over all vertices of the

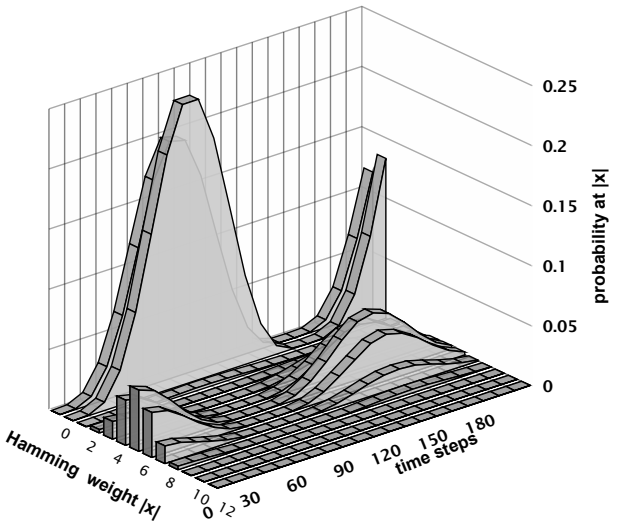

Figure 3.2: Performance of the search algorithm: The search algorithm on the $d = 12$ dimensional hypercube is projected on a line such that all vertices with the same Hamming weight merge into one point. At $t = 0$ the walk starts in the uniform distribution and localises at the marked vertex $v = \vec{0}$ at $t = 74$.

hypercube. The probabilities at the vertices have been projected on a line such that all vertices with the same distance from the marked vertex are identified, see figure 3.3.

The simulation shows that the search localises at the marked vertex after U' has been applied $t = 74$ times. The even higher amplitude at Hamming weight $|\vec{x}| = \sum_{i=1}^{d} x_i = 1$ is the sum of the probabilities for the 12 neighbours of $|v\rangle$, where each of the neighbouring vertices has a probability significantly smaller than the marked vertex.

3.2 Eigenvectors and eigenvalues of the search

For the analysis of the search algorithm, the eigenvectors and eigenvalues of U' are discussed. Since the additional term $U |sv\rangle \langle sv|$ in equation (3.1.12) changes only a few matrix elements of the quantum random walk, it can be treated as a small perturbation.

The analysis follows the line described below. First an operator U_λ is introduced that

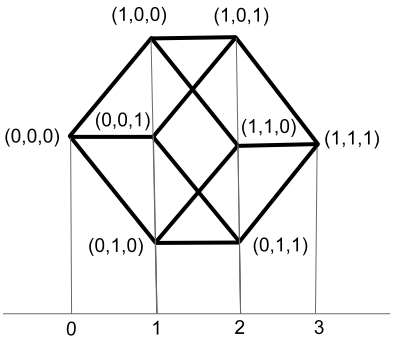

Figure 3.3: The Hamming weight $|\vec{x}|$ measures the distance between $\vec{x}$ and $\vec{0}$. To illustrate this, all vertices with the same Hamming weight are projected to one point.

changes continuously from the quantum random walk to the operator of the search. Then the eigenvectors and eigenvalues of the thus defined operator are discussed. One can easily see, the eigenvectors and eigenvalues can be organised such that only a few of them change as a function of the parameter λ. Thus the subspace of the problem will be reduced and restricted to the set of these eigenvectors only. The spectrum with respect to this reduced subspace is finally be plotted in figure 3.4.

3.2.1 Introduction of a one parameter family of unitary operators and a reduced space

To analyse the search algorithm U' a one parameter family of unitary operators U_λ with parameter λ is defined. The family is organised such that U_λ is unitary for all values of λ and changes continuously from U to U' as λ changes from 0 to 1. These conditions are met by

$$U_\lambda = U + \left(e^{i\lambda\pi} - 1\right) U |sv\rangle \langle sv| . \tag{3.2.1}$$

Note that U_λ is 2-periodic in λ and equals U for all even integers whereas $U_\lambda = U'$ if λ is odd. If analysed in the canonical basis, the operation $\lambda \to 2\pi - \lambda$ acts as complex

conjugation since U and $|sv\rangle\langle sv|$ are both real operators.

To define a maximal symmetry-reduced space it is crucial to note that all eigenvectors of U orthogonal to $|sv\rangle$ remain eigenvectors of U_λ when λ is chosen away from zero and their eigenvalues remain unchanged. That these eigenvectors and their eigenvalues are constant when varying λ follows trivially from the definition $U_\lambda = U + \left(e^{i\lambda\pi} - 1\right) U |sv\rangle\langle sv|$. Therefore it is sufficient to concentrate the investigation on eigenvectors that are not orthogonal to $|sv\rangle$.

These vectors are obtained by a reorganisation of eigenvectors in each of the eigenspaces of U such that the vectors are orthonormalised and there is only one eigenvector not orthogonal to $|sv\rangle$. Let the subspace $\mathcal{H}'$ be the space spanned by this set of vectors. Note that this definition ensures that $\mathcal{H}'$ is the relevant subspace for the investigation because the operators U and U_λ are different only in their action on this subspace.

Up to a normalisation constant, these vectors are obtained by a projection of $|sv\rangle$ into the corresponding subspace.

To continue with the analysis, the scalar product of $|sv\rangle$ with an arbitrary eigenvector according to (3.1.6) is regarded. To simplify notation, a phase factor $e^{\pm i\varphi_k} = \frac{\sqrt{k} \pm i\sqrt{d-k}}{\sqrt{d}}$ can be defined and

$$\left\langle v_{\vec{l}}^{\pm} \mid sv \right\rangle = (-1)^{\vec{l}\vec{v}} \sqrt{2^{-d-1}} \beta_l e^{\pm i\varphi_k}. \tag{3.2.2}$$

The span vectors of the subspace $\mathcal{H}'$ can now be evaluated as

$$|\omega_k^{\pm\prime}\rangle = \sum_{\substack{\vec{l} \\ |\vec{l}|=k}} \left| v_{\vec{l}}^{\pm} \right\rangle \left\langle v_{\vec{l}}^{\pm} \mid sv \right\rangle \tag{3.2.3}$$

$$= \sqrt{2^{-d-1}} e^{\pm i\varphi_k} \beta_k \sum_{\substack{\vec{l} \\ |\vec{l}|=k}} (-1)^{\vec{l}\cdot\vec{v}} \left| v_{\vec{l}}^{\pm} \right\rangle. \tag{3.2.4}$$

This gives normalised vectors of the form

$$\left|\omega_k^{\pm}\right\rangle := \frac{1}{\sqrt{\binom{d}{k}}} \sum_{\substack{\vec{l} \\ |\vec{l}|=k}} (-1)^{\vec{l}\cdot\vec{v}} \left|v_{\vec{l}}^{\pm}\right\rangle . \tag{3.2.5}$$

Again, the vectors defined for $k = 0$ and $k = d$ will be restricted to $\left|\omega_0^+\right\rangle$ and $\left|\omega_d^+\right\rangle$ as these vectors are up to a phase factor identical to the corresponding $^-$ cases. $\left|\omega_0^-\right\rangle$ and $\left|\omega_d^-\right\rangle$ will not be considered.

There are $2d-2$ remaining vectors of type $\left|\omega_k^{\pm}\right\rangle$ and it can be seen that the $2d-2$ dimensional subspace $\mathcal{H}'$ is significantly smaller than $\mathcal{H}$ which is $d2^d$-dimensional.

Note that the vectors in this set are not independent of the marked vertex $\vec{v}$. To simplify the calculation $\vec{v} = \vec{0}$ can be chosen without loss of generality and

$$\left|\omega_k^{\pm}\right\rangle = \frac{1}{\sqrt{\binom{d}{k}}} \sum_{\substack{\vec{l} \\ |\vec{l}|=k}} \left|v_{\vec{l}}^{\pm}\right\rangle \tag{3.2.6}$$

is obtained for the normalised eigenvectors of $U_{\lambda=0}$. The general case $\vec{v} = \vec{a}$ for some vector $\vec{a} \neq \vec{0}$ can easily be reconstructed by performing a transformation in position space $\vec{x} \to \vec{x} \oplus \vec{a}$, where $\oplus$ is again the sum modulo 2.

Although the reasoning for the construction of the subspace $\mathcal{H}'$ is different from the geometrical approach for the subspace in [50], both subspaces are identical since the set of span vectors is the same. Note that per definition, the perturber state $|sv\rangle$ is in $\mathcal{H}'$.

Using the same symbols for the operators in $\mathcal{H}'$ as in the larger space $\mathcal{H}$, U_λ is defined as

$$U_\lambda = U + \left(e^{i\lambda\pi} - 1\right) U \left|sv\right\rangle \left\langle sv\right| \tag{3.2.7}$$

where the operator U is now defined via its action on its eigenvectors in the reduced space

$$U = v_0^+ \left|\omega_0^+\right\rangle \left\langle\omega_0^+\right| + v_d^+ \left|\omega_d^+\right\rangle \left\langle\omega_d^+\right| + \sum_{k=1}^{d-1} \left(v_k^+ \left|\omega_k^+\right\rangle \left\langle\omega_0^+\right| + v_k^- \left|\omega_k^-\right\rangle \left\langle\omega_k^-\right|\right) . \tag{3.2.8}$$

To simplify notation, a new index m is defined to replace both indices, k and $\pm$. Let $m \in \{-d+1, d\}$ such that $\{|m|, sgn(m)\} = \{k, \pm\}$. This leads to an identification $\left|\omega_k^{\pm}\right\rangle = |\omega_m\rangle$, $v_k^{\pm} = e^{\pm i\omega_k} = e^{i\omega_m} = v_m$ and for the phase factor defined in equation (3.2.2) $e^{\pm i\varphi_k} = e^{i\varphi_m}$.

This simplifies the definition of U in the reduced space to

$$U = \sum_{m=-d+1}^{d} e^{i\omega_m} |\omega_m\rangle \langle\omega_m| . \tag{3.2.9}$$

To complete the introduction of the reduced space, $|sv\rangle$ is expanded in the basis of span-vectors of $\mathcal{H}'$.

$$|sv\rangle = \sum_{m=-d+1}^{d} |\omega_m\rangle \langle\omega_m \mid sv\rangle \tag{3.2.10}$$

$$= \sum_{m=-d+1}^{d} |\omega_m\rangle \frac{1}{\sqrt{\binom{d}{k}}} \sum_{\substack{\vec{l} \\ |\vec{l}|=k}} \left\langle v_{\vec{l}}^{\pm} \mid sv \right\rangle , \tag{3.2.11}$$

where the definition (3.2.5) has been used. With equation (3.2.2) the expression simplifies to

$$|sv\rangle = \sum_{m=-d+1}^{d} |\omega_m\rangle \frac{1}{\sqrt{\binom{d}{|m|}}} \sum_{\substack{\vec{l} \\ |\vec{l}|=|m|}} \sqrt{2^{-d-1}} \beta_m e^{i\varphi_m} \tag{3.2.12}$$

$$= \sqrt{2^{-d-1}} \sum_{m=-d+1}^{d} \sqrt{\binom{d}{|m|}} \beta_m e^{i\varphi_m} |\omega_m\rangle . \tag{3.2.13}$$

3.2.2 Spectrum of U_λ

Figure 3.4 shows the numerical calculation of eigenphases of U_λ in $\mathcal{H}'$ for the 20-dimensional hypercube as a function of λ. The spectrum of the reduced space contains several avoided crossings created by two eigenphases crossing diagonal through a spectrum of otherwise nearly constant eigenphases. When varying λ over a small interval near one of the crossings, the eigenvalues not involved in that crossing hardly change. Some of the avoided crossings with small gap have been enlarged in the figure such that the character of the avoided crossing is clearly visible.

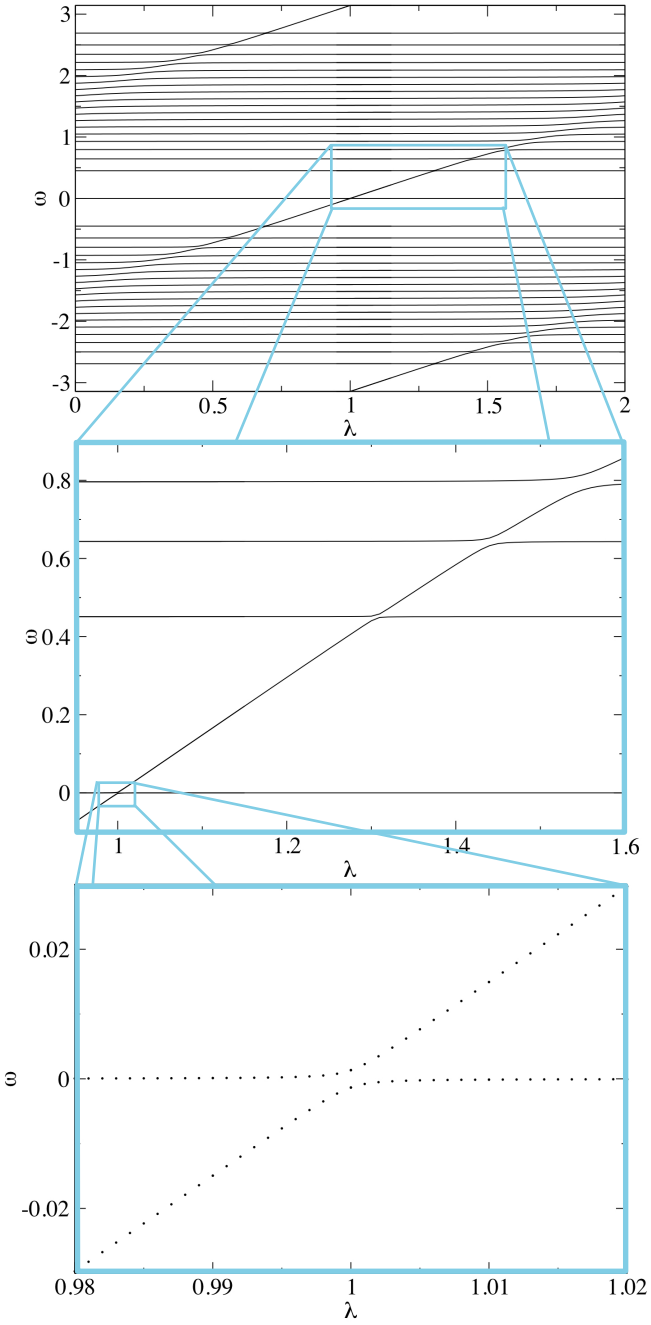

Figure 3.4: Phases of the eigenvalues of U_λ as a function of λ in the reduced space for a 20-dimensional hypercube.

Shenvi, Kempe and Whaley concentrated their analysis on the two eigenvalues of $U_{\lambda=1}$ closest two 1 and this new model reveals that these eigenvalues have an avoided crossing in the region near $\omega = 0$ and $\lambda = 1$.

It is known that oscillations between the two eigenstates of the avoided crossings can occur at the crossing. They appear if the starting state is a linear combination of the two exact eigenvectors, the state will then perform a rotation in the two dimensional subspace of the eigenvectors of the system that are crucial for the crossing. See appendix B for an introduction.

The crossings can be enumerated using the index m according to the eigenphases ω_m participating in the crossings.

Note that the eigenvector $|\omega_0\rangle$ of $U_{\lambda=0}$ with eigenphase 0 is the uniform distribution and therefore identical to the start vector $|\text{start}\rangle$ of the search shown in figure 3.2.

Numerical simulations have shown that the walk U_λ localises at the marked vertex no matter which state $|\omega_m\rangle$ is used as starting state. The only conditions are that it has an avoided crossing, that is, the distance in the crossing is smaller than the distance to neighbouring eigenphases, and that the parameter λ is chosen as the λ-coordinate of the mth crossing.

3.3 Approximative eigenvectors and eigenvalues of U_λ near the mth crossing

Regarding the eigenphases not involved in the crossing as constant can be used to reduce the dimension of the problem even further. Near each crossing, there is one eigenvalue $e^{i\omega_m}$ as defined in (3.1.5). This eigenvalue remains in good approximation constant when λ is varied and belongs to the eigenvector $|\omega_m\rangle$ of $U_{\lambda=0}$. The second

eigenvalue in the crossing changes the phase continuously and is the eigenvalue of an yet unknown vector.

Approximations to these two eigenvectors for each crossing will be used to construct a 2-dimensional subspace spanned by the approximated eigenvectors of U_λ. This subspace can be used to constructed a 2×2 model Hamiltonian H such that e^{-iH} yields a good approximation for U_λ in this 2-dimensional subspace.

For most m's, the eigenphases of $|\omega_m\rangle$ are in good approximation constant when λ is changed and the vectors are assumed to be good approximations for the eigenvectors of U_λ. This can be verified by applying U_λ to the eigenvectors of U

$$U_\lambda |\omega_m\rangle = U |\omega_m\rangle + \left(e^{i\lambda\pi} - 1\right) U |sv\rangle \langle sv \mid \omega_m\rangle \tag{3.3.1}$$

$$= e^{i\omega_m} |\omega_m\rangle + \left(e^{i\lambda\pi} - 1\right) \sqrt{2^{-d-1}} \sqrt{\binom{d}{|m|}} \beta_m e^{-i\varphi_m} U |sv\rangle . \tag{3.3.2}$$

Since the vector $U_\lambda |sv\rangle$ is normalised, the second term on the right hand side scales like $\sqrt{\binom{d}{|m|}/N}$, where N is again the number of vertices of the hypercube. For this reason the error term vanishes for $N \to \infty$ and $|\omega_m\rangle$ is indeed close to an eigenvector of U_λ, as long as $N \gg 1$ and $|m|$ not too close to $\frac{d}{2}$.

To obtain a vector that is approximately an eigenvector of U_λ and has an eigenphase corresponding to one of the diagonal lines in the spectrum, a bit more effort is needed. Let $g^{\pm}(\lambda)$ be a function such that the eigenvalues corresponding to the diagonal lines in the spectrum are given by $e^{ig^{\pm}(\lambda)}$ and let $|\nu_\lambda^{\pm}\rangle$ be the corresponding eigenvectors such that the eigenvalue equation is approximately fulfilled. A rough estimate can be obtained by looking at figure 3.4 and a first approximations to $g^{\pm}(\lambda)$ is obtained as

$$g^{\pm}(\lambda) \approx (\lambda \pm 1) \frac{\pi}{2}. \tag{3.3.3}$$

In this chapter, only the solution crossing at $\omega = 0$ will be considered, that is, $g(\lambda) \approx (\lambda - 1) \frac{\pi}{2}$.

The yet unknown vector $|\nu_\lambda\rangle$ is expanded in the basis of the set of span vectors $|\omega_j\rangle$ of $\mathcal{H}'$. Let

$$|\nu_\lambda\rangle = \sum_{j=-d+1}^{d} a_j |\omega_j\rangle \tag{3.3.4}$$

be the normalised $e^{ig(\lambda)}$-eigenvector with some unknown set of coefficients a_j and let $|\nu_\lambda\rangle$ be orthogonal with respect to the other eigenvector that contributes to the mth crossing, that is $a_m = 0$. The yet arbitrary phase of $|\nu_\lambda\rangle$ will be defined by choosing the scalar product with $|sv\rangle$ real and non-negative and a parameter b is defined such that

$$b = \langle sv \mid \nu_\lambda\rangle \qquad\qquad \in [0,1]\,. \tag{3.3.5}$$

Using these definitions the eigenvalue equation takes the form

$$\begin{aligned}
U_\lambda |\nu_\lambda\rangle =& U|\nu_\lambda\rangle + \left(e^{i\lambda\pi} - 1\right) bU|sv\rangle && (3.3.6)\\
=& e^{ig(\lambda)}|\nu_\lambda\rangle + \left[U|\nu_\lambda\rangle - e^{ig(\lambda)}|\nu_\lambda\rangle + \left(e^{i\lambda\pi} - 1\right) bU|sv\rangle\right] && (3.3.7)\\
=& e^{ig(\lambda)}|\nu_\lambda\rangle + b\left(e^{i\lambda\pi} - 1\right)\sqrt{2^{-d-1}}\sqrt{\binom{d}{|m|}}\beta_m e^{i\varphi_m + i\omega_m}|\omega_m\rangle \\
&+ \sum_{\substack{j=-d+1\\ j\neq m}}^{d}\Bigg[a_j\left(e^{i\omega_k} - e^{ig(\lambda)}\right) \\
&\quad + b\left(e^{i\lambda\pi} - 1\right)\sqrt{2^{-d-1}}\sqrt{\binom{d}{|j|}}\beta_j e^{i\varphi_j + i\omega_j}\Bigg]|\omega_j\rangle\,. && (3.3.8)
\end{aligned}$$

Demanding the term in the square brackets to be zero defines the set of coefficients

$$a_j = \frac{b\left(e^{i\lambda\pi} - 1\right)\sqrt{2^{-d-1}}\sqrt{\binom{d}{|j|}}\beta_j e^{i\varphi_j + i\omega_j}}{e^{ig(\lambda)} - e^{i\omega_j}} \tag{3.3.9}$$

and the approximative eigenvalue equation is

$$U_\lambda|\nu_\lambda\rangle = e^{ig(\lambda)}|\nu_\lambda\rangle + b\left(e^{i\lambda\pi} - 1\right)\sqrt{2^{-d-1}}\sqrt{\binom{d}{|m|}}\beta_m e^{i\varphi_m + i\omega_m}|\omega_m\rangle\,. \tag{3.3.10}$$

As before, the additional term is of the order $\sqrt{\binom{d}{|m|}/N}$ and is small for $N \gg 1$ and $|m|$ not too close to $\frac{d}{2}$.

Since b depends on the a_j's, equation (3.3.9) represents a set of coupled equations and b in (3.3.5) cancels as it appears as a factor in both sides of the equation. Thus (3.3.5) leads

to a sum rule

$$1 \stackrel{!}{=} 2^{-d-1}\left(\mathrm{e}^{\mathrm{i}\lambda\pi}-1\right)\sum_{\substack{j=-d+1\\ j\neq m}}^{d}\binom{d}{|j|}\beta_j^2\frac{\mathrm{e}^{\mathrm{i}\omega_j}}{\mathrm{e}^{\mathrm{i}g(\lambda)}-\mathrm{e}^{\mathrm{i}\omega_j}} \tag{3.3.11}$$

and b remains so far undefined by the above equations and can be used as normalisation parameter if a solution exists. To prove the existence of a solution for the coupled equations (3.3.9), it remains to show that the sum rule (3.3.11) is fulfilled. Note that the sum rule (3.3.11) provides an implicit formula for the phase $g(\lambda)$.

Let λ_m be the value of λ at the avoided crossing of the eigenphases of $|\nu_\lambda\rangle$ and $|\omega_m\rangle$. It remains to prove that the eigenphases cross, that is, the sum rule is at least approximately true in a region for λ near λ_m and for $g(\lambda_m) = \omega_m$.

The technique to solve the equations for λ_m and $g(\lambda)$ at $\lambda \approx \lambda_m$ is to demand $g(\lambda_m) = \omega_m$ and solve (3.3.11) for λ to obtain λ_m. This can in turn be used to verify that $(\lambda_m, g(\lambda_m))$ are indeed the coordinates of the avoided crossing by looking at the spectrum. If this is the case, $g(\lambda)$ near λ_m can be obtained using (3.3.11).

3.3.1 Verification for the $m = 0$ crossing.

The set of equations (3.3.9) has a valid solution if the sum formula (3.3.11) holds. Which will be analysed in the following. For simplicity, the calculation will be done only for the main crossing at $\lambda = 1, \omega = 0$, see figure 3.4.

Note that in the position and coin-space basis of $\mathcal{H}$ both operators, U and the projection on the marking state $|sv\rangle\langle sv|$, are real. Thus the symmetry operation $\lambda \to 2\pi - \lambda$ performs a complex conjugation on U_λ in this basis. Therefore the spectrum of U_λ is symmetric about $(\lambda = 1, \omega = 0)$. Since there is only one crossing in that region of the spectrum, it has to be exactly at that point. Note that the crossings of U_λ in $\mathcal{H}$ are identical to those of U_λ defined for $\mathcal{H}'$ since the space has been reduced by eigenvectors of U_λ with constant eigenphases only.

It remains to show that equation (3.3.11) holds in good approximation at the main crossing for large N. Let

$$S(\lambda) := 2^{-d-1}\left(e^{i\lambda\pi}-1\right)\sum_{\substack{j=-d+1\\ j\neq 0}}^{d}\binom{d}{|j|}\beta_j^2\frac{e^{i\omega_j}}{e^{ig(\lambda)}-e^{i\omega_j}} \tag{3.3.12}$$

and $S(\lambda)\overset{!}{=}1$ needs to be shown.

The sum can be rearranged such that positive and negative j are added up in pairs while the $j=d$ term remains, that is

$$\begin{aligned}S(\lambda) =&2^{-d-1}\left(e^{i\lambda\pi}-1\right)\\ &\cdot\left(\sum_{j=1}^{d-1}\binom{d}{j}\beta_j^2\left(\frac{e^{i\omega_j}}{e^{ig(\lambda)}-e^{i\omega_j}}+\frac{e^{i\omega_{j-d}}}{e^{ig(\lambda)}-e^{i\omega_{j-d}}}\right)+\beta_d^2\frac{e^{i\omega_d}}{e^{ig(\lambda)}-e^{i\omega_d}}\right).\end{aligned} \tag{3.3.13}$$

Using (3.1.7) and (3.1.8), the symmetry of the spectrum of U, $e^{i\omega_{j-d}}=-e^{i\omega_j}$ which follows from (3.1.5) and $e^{i\omega_{j-d}}=e^{-i\omega_{|d-j|}}$ for $0<j<d$, one obtains

$$\begin{aligned}S(\lambda) =&2^{-d-1}\left(e^{i\lambda\pi}-1\right)\left(\sum_{j=1}^{d-1}\binom{d}{j}\left(\frac{e^{i\omega_j}}{e^{ig(\lambda)}-e^{i\omega_j}}+\frac{-e^{i\omega_j}}{e^{ig(\lambda)}+e^{i\omega_j}}\right)\right.\\ &\left.+2\frac{-1}{e^{ig(\lambda)}+1}\right)\end{aligned} \tag{3.3.14}$$

$$=2^{-d-1}\left(e^{i\lambda\pi}-1\right)\left(\sum_{j=1}^{d-1}\binom{d}{j}2\frac{e^{2i\omega_j}}{e^{2ig(\lambda)}-e^{2i\omega_j}}-\frac{2}{e^{ig(\lambda)}+1}\right). \tag{3.3.15}$$

Some trigonometry leads to $\frac{e^{2i\omega_j}}{e^{2ig(\lambda)}-e^{2i\omega_j}}=\frac{i}{2}\cot\left(\omega_j-g(\lambda)\right)-\frac{1}{2}$ and

$\frac{1}{e^{ig(\lambda)}+1}=\frac{1}{2}-\frac{i}{2}\tan\frac{g(\lambda)}{2}$. Now the sum formula simplifies to

$$\begin{aligned}S(\lambda) =&2^{-d-1}\left(e^{i\lambda\pi}-1\right)\left(\sum_{j=1}^{d-1}\binom{d}{j}\left(i\cot\left(\omega_j-g(\lambda)\right)-1\right)-1\right.\\ &\left.+i\tan\frac{g(\lambda)}{2}\right)\end{aligned} \tag{3.3.16}$$

$$\begin{aligned}=&2^{-d-1}\left(e^{i\lambda\pi}-1\right)\left(i\sum_{j=1}^{d-1}\binom{d}{j}\cot\left(\omega_j-g(\lambda)\right)-2^d+1\right.\\ &\left.+i\tan\frac{g(\lambda)}{2}\right),\end{aligned} \tag{3.3.17}$$

where $\sum_{j=0}^{d}\binom{d}{j}=2^d$ has been used.

Inserting $\lambda = 1$ and using $g(1) = 0$, it can be seen that

$$S(1) = 2^{-d-1}(-2)\left(\mathrm{i}\sum_{j=1}^{d-1}\binom{d}{j}\cot(\omega_j) - 2^d + 1\right). \tag{3.3.18}$$

The remaining sum cancels since $\cot\omega_j$ has an odd symmetry about $j = \frac{d}{2}$ and

$$S(1) = 1 - 2^{-d} \tag{3.3.19}$$

confirms the expectation that the sum rule $S(1) = 1$ holds in good approximation.

3.3.2 Taylor expansion of $g(\lambda)$ about 1

To obtain the function $g(\lambda)$ near $\lambda = 1$, a Taylor expansion can be evaluated by calculating the derivatives of $g(\lambda)$ at $\lambda = 1$ iteratively.

The first derivative, $g'(1)$, results from demanding $\frac{dS(\lambda)}{d\lambda}|_{\lambda=1} \stackrel{!}{=} 0$, that is,

$$\begin{aligned} 0 =& \frac{dS(\lambda)}{d\lambda} \qquad (3.3.20)\\ =& 2^{-d-1}\left(\mathrm{e}^{\mathrm{i}\lambda\pi}\mathrm{i}\pi\left(\mathrm{i}\sum_{j=1}^{d-1}\binom{d}{j}\cot(\omega_j - g(\lambda)) - 2^d + 1 + \mathrm{i}\tan\frac{g(\lambda)}{2}\right)\right.\\ &+ \left(\mathrm{e}^{\mathrm{i}\lambda\pi} - 1\right)g'(\lambda)\\ &\left.\cdot\left(\mathrm{i}\sum_{j=1}^{d-1}\binom{d}{j}\left(1 + \cot^2(\omega_j - g(\lambda))\right) + \frac{\mathrm{i}}{2}\left(1 + \tan^2\frac{g(\lambda)}{2}\right)\right)\right), \qquad (3.3.21) \end{aligned}$$

and at $\lambda = 1$

$$0 = 2^{-d-1}\left(-\mathrm{i}\pi\left(-2^d + 1\right) - 2g'(1)\left(\mathrm{i}\sum_{j=1}^{d-1}\binom{d}{j}\left(1 + \cot^2(\omega_j)\right) + \frac{\mathrm{i}}{2}\right)\right), \tag{3.3.22}$$

where the identity $\sum_{j=1}^{d-1}\binom{d}{j}\cot(\omega_j) = 0$ was again useful. Solving this equation for $g'(1)$ results in

$$\begin{aligned} g'(1) &= \frac{2^d\pi - \pi}{2\sum_{j=1}^{d-1}\binom{d}{j}\cot^2(\omega_j) + 2\cdot 2^d - 4 + 1} \qquad (3.3.23)\\ &= \frac{\pi - 2^{-d}\pi}{2 + 2^{-d+1}\sum_{j=1}^{d-1}\binom{d}{j}\cot^2(\omega_j) - 3\cdot 2^{-d}}. \qquad (3.3.24) \end{aligned}$$

Note that both, $g(1) = 0$ and $\lim_{d\to\infty} g'(1) = \frac{\pi}{2}$ agree with the rough estimate $g^{\pm}(\lambda) \approx (\lambda \pm 1)\frac{\pi}{2}$ from equation (3.3.3) if $2^{-d+1}\sum_{j=1}^{d-1}\binom{d}{j}\cot^2(\omega_j) \to 0$ as $d \to \infty$. This remains to be shown.

Let

$$\gamma_d := 2^{-d+1}\sum_{j=1}^{d-1}\binom{d}{j}\cot^2(\omega_j) = 2^{-d+1}\sum_{j=1}^{d-1}\binom{d}{j}\frac{d^2-4jd+4j^2}{4jd-4j^2}, \tag{3.3.25}$$

where the definition of $\omega_{|j|}$ from $e^{i\,\mathrm{sgn}(j)\omega_j} = 1 - \frac{2|j|}{d} - \mathrm{sgn}(j)\,\frac{2i}{d}\sqrt{|j|(d-|j|)}$ was used. For large graphs, that is $d \gg 1$, the de Moivre-Laplace Theorem states $\binom{d}{j}2^{-d} \sim \sqrt{\frac{2}{\pi d}}\exp\left(-\frac{2}{d}\left(j-\frac{d}{2}\right)^2\right)$ [57] and

$$\gamma_d \sim \frac{1}{2}\sqrt{\frac{2}{\pi d}}\sum_{j=1}^{d-1}\frac{d^2-4jd+4j^2}{4jd-4j^2}e^{-\frac{2}{d}\left(j-\frac{d}{2}\right)^2} \tag{3.3.26}$$

is obtained.

To simplify this result Poisson's summation formula is used and the sum is approximated by an integration

$$\gamma_d \sim \sqrt{\frac{1}{2\pi d}}\int_{\frac{1}{2}}^{d-\frac{1}{2}} dx \sum_{n=-\infty}^{\infty}\delta(x-n)\,\frac{d^2-4xd+4x^2}{4xd-4x^2}e^{-\frac{2}{d}\left(x-\frac{d}{2}\right)^2}. \tag{3.3.27}$$

The sum of delta distributions is periodic in x and can be replaced by its Fourier series which leads to

$$\gamma_d \sim \sqrt{\frac{1}{2\pi d}}\int_{\frac{1}{2}}^{d-\frac{1}{2}} dx \sum_{n=-\infty}^{\infty} e^{2\pi i n x}\frac{d^2-4xd+4x^2}{4xd-4x^2}e^{-\frac{2}{d}\left(x-\frac{d}{2}\right)^2}. \tag{3.3.28}$$

Note that this step is not rigorous, since the left hand side of the equation is not a function but a distribution, and convergence of the right hand side has not been shown. However, it can be used in this case [58].

The substitution $y = \frac{2x}{d}$ is employed to obtain

$$\gamma_d \sim \frac{1}{2}\sqrt{\frac{d}{2\pi}}\int_{\frac{1}{d}}^{2-\frac{1}{d}} dy \sum_{n=-\infty}^{\infty} e^{\pi i n y d}\frac{1-2y+y^2}{2y-y^2}e^{-\frac{d}{2}(y-1)^2}. \tag{3.3.29}$$

The approximation above holds for large N only, that is, search algorithms searching a large graph. In the case of $d \gg 1$ all terms $n \neq 0$ can be neglected because there are

fast oscillations in the exponent leading to sub dominant terms only. Furthermore, the symmetry about $y = 1$ allows a reduction of the integration interval.

$$\gamma_d \sim \sqrt{\frac{d}{2\pi}} \int_{\frac{1}{d}}^{1} dy \frac{1-2y+y^2}{2y-y^2} \mathrm{e}^{-\frac{d}{2}(y-1)^2} \tag{3.3.30}$$

$$= \sqrt{\frac{d}{2\pi}} \int_{\frac{1}{d}}^{\frac{1}{2}} dy \frac{1-2y+y^2}{2y-y^2} \mathrm{e}^{-\frac{d}{2}(y-1)^2} + \sqrt{\frac{d}{2\pi}} \int_{\frac{1}{2}}^{1} dy \frac{1-2y+y^2}{2y-y^2} \mathrm{e}^{-\frac{d}{2}(y-1)^2}, \tag{3.3.31}$$

where the integration interval has been split into two regions.

The first integration can easily be estimated by providing an upper and lower bound which equals or approaches 0 for $d \to \infty$. Since the integrand is greater than 0 the lower bound is trivial as the integral has to be greater than 0 as well. An upper bound can be defined by replacing y by $\frac{1}{2}$ in the exponential function,

$$\sqrt{\frac{d}{2\pi}} \int_{\frac{1}{d}}^{\frac{1}{2}} dy \frac{1-2y+y^2}{2y-y^2} \mathrm{e}^{-\frac{d}{2}(y-1)^2} < \sqrt{\frac{d}{2\pi}} \int_{\frac{1}{d}}^{\frac{1}{2}} dy \frac{1-2y+y^2}{2y-y^2} \mathrm{e}^{-\frac{d}{8}} = \sqrt{\frac{d}{2\pi}} \mathrm{e}^{-\frac{d}{8}} \frac{2-d+d\ln(2d-1)-d\ln 3}{2d} \tag{3.3.32}$$

which has a leading term $\sqrt{\frac{d}{2\pi}} \mathrm{e}^{-\frac{d}{8}} \ln(2d-1)$.

For the second integration $f(d) := \sqrt{\frac{d}{2\pi}} \int_{\frac{1}{2}}^{1} dy \frac{1-2y+y^2}{2y-y^2} \mathrm{e}^{-\frac{d}{2}(y-1)^2}$, an upper and lower bound can be found by appointing bounds for the denominator

$$\sqrt{\frac{d}{2\pi}} \int_{\frac{1}{2}}^{1} dy \left(1-2y+y^2\right) \mathrm{e}^{-\frac{d}{2}(y-1)^2} < f(d) < \frac{4}{3} \sqrt{\frac{d}{2\pi}} \int_{\frac{1}{2}}^{1} dy \left(1-2y+y^2\right) \mathrm{e}^{-\frac{d}{2}(y-1)^2}. \tag{3.3.33}$$

The remaining integration that appears in both bounds yields

$$\sqrt{\frac{d}{2\pi}} \int_{\frac{1}{2}}^{1} dy \left(1-2y+y^2\right) \mathrm{e}^{-\frac{d}{2}(y-1)^2} = \sqrt{\frac{d}{2\pi}} \left(-\frac{\mathrm{e}^{-\frac{d}{8}}}{2d} + \sqrt{\frac{\pi}{2}} d^{-\frac{3}{2}} \mathrm{erf}\left(\sqrt{\frac{d}{8}}\right) \right), \tag{3.3.34}$$

where $\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x dt \mathrm{e}^{-t^2}$ is the error function. Since $\mathrm{erf}(x) \approx 1$ for $x \gg 1$, the leading term for $d \gg 1$ is $\frac{1}{2d}$. As both sides of (3.3.33) differ only by a constant, and all

the remaining terms of (3.3.31) decay faster to 0 as $e^{-\frac{d}{8}}d$ the leading order and its error are

$$\gamma_d = \frac{1}{2d} + \mathcal{O}\left(e^{-\frac{d}{8}}d\right). \tag{3.3.35}$$

To conclude this section, it has been shown that the sum formula (3.3.11) is fulfilled for $m = 0$ as it is true for $\lambda = 1$ and $g(1) = 0$ in good approximation and that the first derivative fits to the rough approximation (3.3.3) taken from the spectrum in figure 3.4 since

$$g'(1) = \frac{\pi - 2^{-d}\pi}{2 + \frac{1}{2d} - 3\cdot 2^{-d} + \mathcal{O}\left(e^{-\frac{d}{8}}d\right)} \approx \frac{\pi}{2} \qquad \text{for } d \gg 1. \tag{3.3.36}$$

3.3.3 The crossing eigenvector

So far only the sum formula for the $m = 0$ crossing has been discussed. It remains to show that there also exist solutions for the crossings at $m \neq 0$. Since the discussion in section 3.6 will show that $m = 0$ is the only important case, a detailed discussion of the non-relevant crossings will be skipped and for now it will be assumed that the sum formula holds for all crossings and the set of equations in (3.3.9) has a valid solution.

However, for $m = 0$ the results obtained stand on firm ground. For $m = 0$ there exists a valid solution for the coefficients a_j and the eigenvalue equation for U_λ has been solved approximately resulting in the approximate eigenvector

$$|\nu_\lambda\rangle = \sum_{\substack{j=-d+1\\ j\neq m}}^{d} a_j \left|\omega_j\right\rangle \tag{3.3.37}$$

$$= b\left(e^{i\lambda\pi} - 1\right)\sqrt{2^{-d-1}} \sum_{\substack{j=-d+1\\ j\neq m}}^{d} \frac{\sqrt{\binom{d}{|j|}}\beta_j e^{i\varphi_j + i\omega_j}}{e^{ig(\lambda)} - e^{i\omega_j}} \left|\omega_j\right\rangle. \tag{3.3.38}$$

It is interesting to note that the vector has a coefficient $\left(e^{i\lambda\pi} - 1\right)$ and thus, the reasoning breaks down whenever λ approaches an even integer. Hence the solution exists only if

U is perturbed by the marking coin. This is indeed no surprise as the diagonal lines in the spectrum disappear when $\lambda = 0$ or 2 as can be seen in figure 3.4.

The yet unknown parameter b gives the normalisation of the vector and can therefore be defined by demanding $\langle \nu_\lambda \mid \nu_\lambda \rangle = 1$.

3.3.4 Normalisation of the vector of the localised state

The eigenvector $|\nu_\lambda\rangle$ as defined in (3.3.38) has a so far unknown normalisation constant b. Since b was originally defined as scalar product of two normalised vectors (3.3.5) and the phase was defined such that b is real and positive, $0 \leq b \leq 1$ can be deduced.

To start with, $\langle \nu_\lambda \mid \nu_\lambda \rangle = 1$ results in the condition

$$\frac{1}{|b|^2} = \left| e^{i\lambda\pi} - 1 \right|^2 2^{-d-1} \sum_{\substack{j=-d+1 \\ j\neq m}}^{d} \binom{d}{|j|} \frac{\beta_j^2}{\left| e^{ig(\lambda)} - e^{i\omega_j} \right|^2}. \tag{3.3.39}$$

It is a hard challenge to estimate the sum but it is related to the sum formula discussed in section 3.3.1. Remember that the sum formula and its derivatives with respect to λ define the function $g(\lambda)$ via a Taylor expansion about the crossing and that the first two terms about $\lambda = 1$ at $m = 0$ have been calculated in section 3.3.1. The condition $\frac{dS(\lambda)}{d\lambda}|_{\lambda=1} \stackrel{!}{=} 0$ will be used to calculate the demanded sum and $0 = \frac{dS}{d\lambda}$ gives

$$0 = \frac{d}{d\lambda} \left(2^{-d-1} \left(e^{i\lambda\pi} - 1 \right) \sum_{\substack{j=-d+1 \\ j\neq m}}^{d} \binom{d}{|j|} \beta_j^2 \frac{e^{i\omega_j}}{e^{ig(\lambda)} - e^{i\omega_j}} \right) \tag{3.3.40}$$

$$\begin{aligned} = & 2^{-d-1} i\pi e^{i\lambda\pi} \sum_{\substack{j=-d+1 \\ j\neq m}}^{d} \binom{d}{|j|} \beta_j^2 \frac{e^{i\omega_j}}{e^{ig(\lambda)} - e^{i\omega_j}} \\ & - 2^{-d-1} \left(e^{i\lambda\pi} - 1 \right) i g'(\lambda) \sum_{\substack{j=-d+1 \\ j\neq m}}^{d} \binom{d}{|j|} \beta_j^2 \frac{e^{i\omega_j + ig(\lambda)}}{\left(e^{ig(\lambda)} - e^{i\omega_j} \right)^2} \end{aligned} \tag{3.3.41}$$

The first term can easily be calculated using the sum formula and for the second term it is sufficient to note that the phase factor in the nominator of the fraction can be moved

to the denominator

$$0 = \frac{\mathrm{i}\pi \mathrm{e}^{\mathrm{i}\lambda\pi}}{\mathrm{e}^{\mathrm{i}\lambda\pi}-1} - 2^{-d-1}\left(\mathrm{e}^{\mathrm{i}\lambda\pi}-1\right)\mathrm{i}g'(\lambda) \cdot \sum_{\substack{j=-d+1 \\ j\neq m}}^{d} \binom{d}{|j|} \beta_j^2 \frac{1}{\left(\mathrm{e}^{\mathrm{i}g(\lambda)}-\mathrm{e}^{\mathrm{i}\omega_j}\right)\left(\mathrm{e}^{-\mathrm{i}\omega_j}-\mathrm{e}^{-\mathrm{i}g(\lambda)}\right)}. \tag{3.3.42}$$

This results in an expression for the demanded sum

$$\frac{\pi \mathrm{e}^{\mathrm{i}\lambda\pi}}{g'(\lambda)\left(\mathrm{e}^{\mathrm{i}\lambda\pi}-1\right)^2} = -2^{-d-1} \sum_{\substack{j=-d+1 \\ j\neq m}}^{d} \binom{d}{|j|} \frac{\beta_j^2}{\left|\mathrm{e}^{\mathrm{i}g(\lambda)}-\mathrm{e}^{\mathrm{i}\omega_j}\right|^2}. \tag{3.3.43}$$

Therefore, the normalisation constant b is ruled by the first derivative of $g(\lambda)$ and

$$\frac{1}{|b|^2} = -\left|\mathrm{e}^{\mathrm{i}\lambda\pi}-1\right|^2 \frac{\pi \mathrm{e}^{\mathrm{i}\lambda\pi}}{g'(\lambda)\left(\mathrm{e}^{\mathrm{i}\lambda\pi}-1\right)^2} \tag{3.3.44}$$

$$= \frac{\pi}{g'(\lambda)}. \tag{3.3.45}$$

Demanding $0 \leq b \leq 1$ now defines the phase factor of the normalisation constant of $|\nu_\lambda\rangle$ and results in

$$b = \sqrt{\frac{g'(\lambda)}{\pi}}. \tag{3.3.46}$$

Note that from the rough estimate for $g(\lambda)$ in (3.3.3) $b \approx \frac{1}{\sqrt{2}}$ can be estimated which agrees with $0 \leq b \leq 1$.

3.3.5 Localisation at the target state.

There are two approximate eigenvectors at the crossing, that is, $|\omega_m\rangle$ which is used as start state and the other, the perturber state $|\nu_\lambda\rangle$ whose eigenphase crosses diagonal through the spectrum (figure 3.4) which has ideally a big overlap with the marked state $|sv\rangle$ of the search algorithm. To analyse if this is indeed the case, it is once more important to remember that b has been defined as the scalar product $b = \langle sv \mid \nu_\lambda\rangle$ in (3.3.5). For $\lambda = 1$ and $d \gg 1$, the first derivative of $g(\lambda)$ as calculated in (3.3.36) results in

$$\langle sv \mid \nu_\lambda\rangle|_{\lambda=1} \approx \sqrt{\frac{1}{2+\frac{1}{2d}}}. \tag{3.3.47}$$

This is in fact a crucial point as the approach in this thesis is to characterise U_λ by concentrating on the avoided crossings in the spectrum and to reduce the problem to a two dimensional subspace spanned by $|\omega_m\rangle$ and $|\nu_\lambda\rangle$. Since $|\omega_m\rangle$ is the starting state of the algorithm, and for $m = 0$ also the uniform distribution, this result proves that the target state $|sv\rangle$ has a notable overlap with the other approximate eigenvector. Only a relatively large b enables this approach to be appropriate since it proves that the perturber state $|sv\rangle$ has a sufficiently large contribution in the 2 dimensional subspace in question.

Although the overlap given in (3.3.47) does not seem much, it leads to a probability of nearly $\frac{1}{2}$ which is in particular not d-dependent.

This result is notable since the algorithm for the $m = 0$ crossing starts in the uniform distribution in position space. Half of the vertices have a distance to $\vec{v} = \vec{0}$ with even and half of them a distance with odd Hamming weight. Each application of $U_{\lambda=1}$ changes the parity of the Hamming weight; the amplitude at vertices in even distances are moved to vertices with odd distance and vice versa. To localise more than half of the probability on the target vertex would mean that there exists a time T such that after T applications of $U_{\lambda=1}$ more than half of the probability is localised on vertices with an even distance, that is the target vertex $|v\rangle$ itself, which is impossible. As a consequence the search algorithm that starts in the uniform distribution cannot achieve a localisation probability higher than $\frac{1}{2}$.

However, numerical simulations as shown in figure 3.2 suggest that $|\nu_\lambda\rangle$ is localised on the hypercube *somewhere near the target vertex* $|v\rangle$. To specify this *somewhere near the target vertex* analytical, it is interesting to calculate the scalar product $\langle \nu_\lambda \mid U \mid sv\rangle$ which gives a lower bound to the amplitude localised on the nearest neighbours.

Obviously, the state $U|sv\rangle$ is localised on the d nearest neighbours of the target vertex

since the state $|sv\rangle$ has been propagated once. An ad hoc argument predicts that the probability at the nearest neighbours is equal or larger than the probability on the marked vertex. It is known from numerics (figure 3.2) that the localisation at the target vertex holds for a few time steps. It can be deduced that the amplitude on the target vertex will propagate to the nearest neighbours when U_λ is applied once and some of the amplitude of the nearest neighbours propagates to the target vertex to replace the amplitude that was shifted away. To keep the probability nearly constant the amplitude localised on the nearest neighbours cannot be less than b.

However, the calculation is straight forward and yields

$$\begin{aligned}
\langle \nu_\lambda \mid U \mid sv\rangle =& b\left(\mathrm{e}^{-\mathrm{i}\lambda\pi}-1\right)2^{-d-1} \\
&\cdot \sum_{\substack{j=-d+1\\ j\neq m}}^{d}\binom{d}{|j|}\beta_j^2\frac{\mathrm{e}^{-\mathrm{i}\varphi_j-\mathrm{i}\omega_j}}{\mathrm{e}^{-\mathrm{i}g(\lambda)}-\mathrm{e}^{-\mathrm{i}\omega_j}}\mathrm{e}^{+\mathrm{i}\varphi_j}\left\langle\omega_j\right| U\left|\omega_j\right\rangle \qquad (3.3.48)\\
=& b\mathrm{e}^{-\mathrm{i}\lambda\pi}\left(\mathrm{e}^{\mathrm{i}\lambda\pi}-1\right)2^{-d-1}\mathrm{e}^{\mathrm{i}g(\lambda)}\sum_{\substack{j=-d+1\\ j\neq m}}^{d}\binom{d}{|j|}\beta_j^2\frac{\mathrm{e}^{\mathrm{i}\omega_j}}{\mathrm{e}^{\mathrm{i}g(\lambda)}-\mathrm{e}^{\mathrm{i}\omega_j}} \qquad (3.3.49)\\
=& b\mathrm{e}^{\mathrm{i}\omega_m}\mathrm{e}^{-\mathrm{i}\lambda\pi} \qquad (3.3.50)
\end{aligned}$$

which provides a more rigorous proof for the ad hoc argument discussed above. In the last step, the sum formula (3.3.11) and $\mathrm{e}^{\mathrm{i}g(\lambda)}=\mathrm{e}^{\mathrm{i}\omega_m}$ have been used. Note that $U|sv\rangle$ is only one state out of d^2 states localised on the nearest neighbours and the probability that is actually localised on the nearest neighbours is very likely to extend the probability in $U|sv\rangle$.

For the main crossing at $\lambda=1$ the result is remarkable since the amplitude of $|\nu_\lambda\rangle$ in the two orthogonal states $|sv\rangle$ and $U|sv\rangle$ is, up to a phase, equal to b and since it approaches $\frac{1}{\sqrt{2}}$ for $d\to\infty$. The probability P, to measure $|\nu_{\lambda=1}\rangle$ either at the marked vertex $|v\rangle$ or on its nearest neighbours is

$$P=2b^2=\frac{1-2^{-d}}{1+\frac{1}{4d}-3\cdot 2^{-d-1}+\mathcal{O}\left(\mathrm{e}^{-\frac{d}{8}}d\right)} \qquad (3.3.51)$$

and approaching 1 if $d \to \infty$. Therefore the state $|\nu_\lambda\rangle$ can be considered localised on the marked vertex and its nearest neighbours. A measurement once the search has localised in $|\nu_\lambda\rangle$ gives either the target vertex or one of its nearest neighbours with a probability close to 1. Note that for the crossings where $m \neq 0$, $g'(\lambda) \approx \frac{\pi}{2}$ is known from the numerics only and the result has not been verified analytically.

Notably, the contribution on the nearest neighbours is not an arbitrary state but $U|sv\rangle$. As a consequence, if the outcome of the measurement is not the target vertex itself, the coin state at the measured vertex points into the direction of the target vertex as can be verified by the definition of the shift operator S in (3.1.3). Therefore the target vertex can be identified and confirmed after the measurement with at most two more oracle queries first on the vertex determined by the measurement and, if this fails to be the target state, on the neighbouring vertex of this vertex singled out by the coin state.

So far, it has been shown that the two dimensional subspace contains the starting state and a state localised at the target vertex. To prove that the search algorithm on the hypercube indeed succeeds to localise at the marked vertex in a time $\mathcal{O}\left(\frac{1}{\sqrt{N}}\right)$, it remains to show that there exists an integer T, such that $U_\lambda^T|\omega_m\rangle$ has a strong overlap with $|\nu_\lambda\rangle$ and to calculate the time T.

3.4 Model of avoided crossings

The analysis in the previous sections identified two approximate eigenvectors $|\omega_m\rangle$ and $|\nu_\lambda\rangle$ and their eigenvalues. The two vectors span an approximately invariant subspace relevant for the avoided crossing of U_λ. In fact the approximate eigenvalues of these vectors do not avoid the crossing at $\lambda = \lambda_m$.

Since the subspace is identified and starting and localised state are contained in this space the next step is to construct a 2×2 model for the search algorithm U_λ. This can

be done using a Hamiltonian, that is a hermitian matrix H that characterises the time propagation through $\left(U_{\lambda_m}^{2\times 2}\right)^t = \mathrm{e}^{-\mathrm{i}Ht}$, where Planck's constant was chosen as $\hbar = 1$.

In general, an avoided crossing can be described using a Hamiltonian containing the two eigenvalues and a small coupling constant $\epsilon > 0$ with an arbitrary phase $\mathrm{e}^{\mathrm{i}\delta}$. Thus the crossing in the search algorithm for the mth crossing at λ_m can be analysed using the model Hamiltonian

$$H = \begin{pmatrix} -\omega_m & \epsilon\,\mathrm{e}^{\mathrm{i}\delta} \\ \epsilon\,\mathrm{e}^{-\mathrm{i}\delta} & -\omega_m \end{pmatrix}, \tag{3.4.1}$$

where the identity $g(\lambda_m) = \omega_m$, the so far unknown real constants ϵ and δ and the canonical basis is defined as $|1\rangle = |\omega_m\rangle$, $|2\rangle = |\nu_{\lambda_m}\rangle$ is used. Since ω_m is real, the matrix is Hermitian.

Eigenvectors and eigenvalues of H are easily calculated

$$\begin{array}{ll} \text{eigenvector} & \text{eigenvalue} \\ |u_1\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -\mathrm{e}^{-\mathrm{i}\delta} \end{pmatrix} & -\omega_m - \epsilon \\ |u_2\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ \mathrm{e}^{-\mathrm{i}\delta} \end{pmatrix} & -\omega_m + \epsilon. \end{array} \tag{3.4.2}$$

Starting and localised state are calculated in the $|u_1\rangle$, $|u_2\rangle$ basis, that is,

$$|\omega_m\rangle = \frac{1}{\sqrt{2}}\left(|u_1\rangle + |u_2\rangle\right) \tag{3.4.3}$$

$$|\nu_{\lambda_m}\rangle = \frac{-\mathrm{e}^{\mathrm{i}\delta}}{\sqrt{2}}\left(|u_1\rangle - |u_2\rangle\right). \tag{3.4.4}$$

3.4.1 Time of the search

Since the preliminary calculations are done, the 2×2 model for U_{λ_m} can be applied t times to the starting state

$$\left(U_{\lambda_m}^{2\times 2}\right)^t |\text{start}\rangle = \mathrm{e}^{-\mathrm{i}Ht} |\omega_m\rangle \tag{3.4.5}$$

$$= \mathrm{e}^{-\mathrm{i}Ht} \frac{1}{\sqrt{2}} \left(|u_1\rangle + |u_2\rangle\right) \tag{3.4.6}$$

$$= \frac{1}{\sqrt{2}} \left(\mathrm{e}^{\mathrm{i}(\omega_m+\epsilon)t} |u_1\rangle + \mathrm{e}^{\mathrm{i}(\omega_m-\epsilon)t} |u_2\rangle\right). \tag{3.4.7}$$

Defining the time $t := \frac{\pi}{2\epsilon}$ leads to $\mathrm{e}^{\pm\mathrm{i}\epsilon t} = \pm\mathrm{i}$ and

$$\left(U_{\lambda_m}^{2\times 2}\right)^t |\text{start}\rangle = \frac{\mathrm{e}^{\mathrm{i}\omega_m t}}{\sqrt{2}} \left(\mathrm{i} |u_1\rangle - \mathrm{i} |u_2\rangle\right) \tag{3.4.8}$$

$$= -\mathrm{i}\mathrm{e}^{\mathrm{i}\omega_m t - \mathrm{i}\delta} |\nu_{\lambda_m}\rangle \tag{3.4.9}$$

which is the localised state. Thus the time the walk needs to localise at the marked vertex is inversely proportional to the parameter ϵ in the Hamiltonian.

Since the two eigenvalues of H given in equation (3.4.2) differ by 2ϵ, the coupling constant ϵ is half the distance of the eigenvalues in the spectrum of U_{λ_m} at the crossing. According to figure 3.4 the search algorithm for $m = 0$ can be expected to take longest since it describes the algorithm at the crossing with the smallest gap in the avoided crossing.

3.4.2 Size of the gap

The size of the gap and the time the search algorithm needs is ruled by the coupling constant ϵ in the model Hamiltonian. The entries of the model Hamiltonian H can be determined using the relation $U_{\lambda_m}^{2\times 2} = \mathrm{e}^{-\mathrm{i}H}$. Let T be the transformation matrix to diagonalise H, then

$$T^{-1}HT = \text{diag}\left(-\omega_m - \epsilon, -\omega_m + \epsilon\right) \tag{3.4.10}$$

and

$$U_{\lambda_m}^{2\times 2} = T\mathrm{diag}\left(\mathrm{e}^{\mathrm{i}\omega_m+\mathrm{i}\epsilon}, \mathrm{e}^{\mathrm{i}\omega_m-\mathrm{i}\epsilon}\right) T^{-1} \tag{3.4.11}$$

for the 2×2 model of the unitary operator $U_{\lambda_m}^{2\times 2}$ and after a transformation into the original basis

$$U_{\lambda_m}^{2\times 2} = \begin{pmatrix} \mathrm{e}^{\mathrm{i}\omega_m}\cos\epsilon & -\mathrm{i}\mathrm{e}^{\mathrm{i}\omega_m}\mathrm{e}^{\mathrm{i}\delta}\sin\epsilon \\ -\mathrm{i}\mathrm{e}^{\mathrm{i}\omega_m}\mathrm{e}^{-\mathrm{i}\delta}\sin\epsilon & \mathrm{e}^{\mathrm{i}\omega_m}\cos\epsilon \end{pmatrix}. \tag{3.4.12}$$

These matrix elements can be calculated more easily. In fact most of the work is done by using equations (3.3.2), (3.3.10) and (3.3.50). Since $g(\lambda_m) = \omega_m$ the diagonal entries are

$$\langle \omega_m \mid U_{\lambda_m} \mid \omega_m \rangle = \mathrm{e}^{\mathrm{i}\omega_m} + \left(\mathrm{e}^{\mathrm{i}\lambda\pi} - 1\right) 2^{-d-1} \binom{d}{|m|} \beta_m^2 \mathrm{e}^{\mathrm{i}\omega_m} \tag{3.4.13}$$

$$\langle \nu_{\lambda_m} \mid U_{\lambda_m} \mid \nu_{\lambda_m} \rangle = \mathrm{e}^{\mathrm{i}\omega_m} \tag{3.4.14}$$

and for the off-diagonal terms

$$\langle \omega_m \mid U_{\lambda_m} \mid \nu_{\lambda_m} \rangle = b\left(\mathrm{e}^{\mathrm{i}\lambda\pi} - 1\right) \sqrt{2^{-d-1}} \sqrt{\binom{d}{|m|}} \beta_m \mathrm{e}^{\mathrm{i}\varphi_m + \mathrm{i}\omega_m} \tag{3.4.15}$$

$$\langle \nu_{\lambda_m} \mid U_{\lambda_m} \mid \omega_m \rangle = -b\left(\mathrm{e}^{-\mathrm{i}\lambda\pi} - 1\right) \sqrt{2^{-d-1}} \sqrt{\binom{d}{|m|}} \beta_m \mathrm{e}^{-\mathrm{i}\varphi_m + \mathrm{i}\omega_m} \tag{3.4.16}$$

are obtained.

By comparing these matrix elements to the ones in (3.4.12) the yet unknown constant ϵ can be estimated using $\sin\epsilon = b\left|\mathrm{e}^{\mathrm{i}\lambda\pi} - 1\right| \sqrt{2^{-d-1}} \sqrt{\binom{d}{|m|}} \beta_m$ and $\cos\epsilon = 1 + \mathcal{O}\left(2^{-d}\right)$ resulting from the diagonal and off-diagonal terms respectively.

Thus the constant

$$\epsilon = b\left|\mathrm{e}^{\mathrm{i}\lambda\pi} - 1\right| \sqrt{2^{-d-1}} \sqrt{\binom{d}{|m|}} \beta_m + \mathcal{O}\left(2^{-d}\right) \tag{3.4.17}$$

and the phase factor

$$\mathrm{e}^{\mathrm{i}\delta} = \mathrm{i}\mathrm{e}^{\mathrm{i}\phi_m} \frac{\mathrm{e}^{\mathrm{i}\lambda_m\pi} - 1}{\left|\mathrm{e}^{\mathrm{i}\lambda_m\pi} - 1\right|} \tag{3.4.18}$$

are obtained. Since 2^d is the number of vertices of the d-dimensional hypercube, the gaps $\Delta_m = 2\epsilon$ for fixed m scale like $\frac{1}{\sqrt{N}}$ whereas the error is of order $\frac{1}{N}$.

3.4.3 Analogy to Grover's search algorithm

The search algorithms for the $m = 0$th crossing on the hypercube discussed above and the search algorithm on a regular d-dimensional square lattice as discussed in chapter 4 have some analogy to Grover's algorithm. Grover's algorithm has been discussed in references [5, 23] and introduced in section 2.3.1 of this thesis. Like in Grover's algorithm, the main effect of the search algorithms discussed here take place in a 2-dimensional subspace spanned by the uniform superposition in position space and a state localised at the target vertex. Ambainis, Kempe and Rivosh noted that Grover's algorithm can be viewed as a search algorithm on a fully connected graph [25].

The crucial part of Grover's algorithm consists of the two steps (2a) and (2b) discussed in section 2.3.1. The first step, (2a), is a reflection about the marked state and the second, (2b), a reflection about the uniform state that acts as starting state. Both operations are performed alternatingly. The same can be said for the search algorithm on the hypercube for $m = 0$ at $\lambda = 1$ and the search algorithm on the regular lattice that will be discussed in chapter 4 since both algorithms can be written as

$$U_1 = U\left(\mathbb{1} - 2\left|sv\right\rangle\left\langle sv\right|\right). \tag{3.4.19}$$

Here $\mathbb{1} - 2\left|sv\right\rangle\left\langle sv\right|$ is a reflection about the target state $\left|sv\right\rangle$, so there is a one to one correspondence to the equivalent step in Grover's algorithm.

The second step of Grover's algorithm is a reflection about the uniform distributed state. Unlike in Grover's algorithm, the algorithms on graphs have a spatial order of the vertices since they are arranged on a graph and transition can only take place to connected vertices. Taking this into account, there is a correspondence of U to step (2b) since the matrix U performs a local coin flip that is equivalent to the matrix D as defined in (2a) and then shifts the walk to neighbouring vertices.

3.5 Results for the reduced space

To summarise the results for the reduced space $\mathcal{H}'$, the previous sections are recapitulated.

In section 3.4.1 it was shown that

$$T = \text{integer}\,(t) = \text{integer}\left(\frac{\pi}{2\epsilon}\right) \tag{3.5.1}$$

applications of U_{λ_m} to the starting state $|\text{start}\rangle = |\omega_m\rangle$ are needed. After this time the walk is localised on the target vertex with probability b^2 and with at least the same probability the walk can be found on the nearest neighbours of the target vertex $|v\rangle$ and b has been estimated for large N as $b \approx \sqrt{\frac{1}{2}}$, see equations (3.3.46) and (3.3.3) [1]. For $\lambda = 1$, a more detailed expression for b can be obtained from (3.3.46) and (3.3.36).

All in all the search needs a time

$$t_m = \frac{\sqrt{2^{d-1}}\pi}{b\left|e^{i\lambda_m\pi} - 1\right|\sqrt{\binom{d}{|m|}\beta_m}} + \mathcal{O}\,(1) \tag{3.5.2}$$

to localise at the target vertex. Thus the result for the leading order scales like $\sqrt{N}$, where $N = 2^d$ is the number of vertices of the hypercube. This coincides with the scaling obtained by Shenvi, Kempe and Whaley [50]. Additionally to the scaling of the leading order term, the coefficient has also been obtained here and it can be seen that the walk succeeds faster the higher m.

Figure 3.5 shows numerical results for the time the search needs as a function of the number of dimensions d for $m = 0,\ 1,\ 2$ and 3 drawn as circles, squares and triangles, respectively.

[1] Strictly speaking, $b \approx \sqrt{\frac{1}{2}}$ has not been shown to hold for general $m \neq 0$ since the estimate for the slope of $g\,(\lambda)$ was obtained as observation from numerical calculations of the spectrum of U_λ for some d. Only for the central crossing at $\lambda_0 = 1$, the leading order contributions for $d \gg 1$ has been calculated and the thus obtained results agrees with the rough estimate taking from figure 3.4

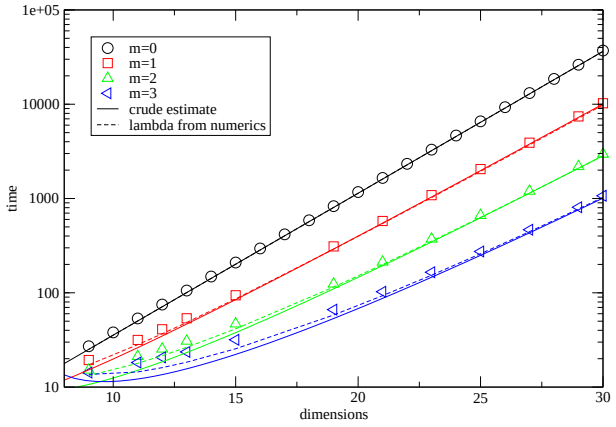

Figure 3.5: Comparison of numerical and theoretical results for the time of the search. The symbols correspond to numerical results while the lines show the theoretical results. The continuous line was determined using the rough estimate (3.3.3) for the values λ_m where the avoided crossings occur. The dashed lines use values for λ_m obtained numerically from the spectrum.

The straight and dashed lines show different approximations of the theoretical results in equation (3.5.2). Since the evaluations of b and λ_m are hard, the approximations discussed in section 3.3 for $d \gg 1$ have been used.

Both approximations for the theoretical results make use of $b \approx \sqrt{\frac{1}{2}}$, whereas λ_m was estimated differently. The rough estimate

$$\omega_m = g(\lambda_m) \approx (\lambda_m - 1)\frac{\pi}{2} \tag{3.5.3}$$

was used for the evaluation of the time shown as continuous line. For the approximation shown as dashed lines, the λ_m-values resulting from numerical calculations of the spectra have been employed; that is, the λ-values used where obtained by searching for the minimal distance between the eigenphases in the gaps of the avoided crossings in the spectrum.

Only for the $m = 0$ crossing no approximation was used as $\lambda_0 = 1$ is known exactly.

For large d both analytical results are in good agreement with the numerics. Since the approximations used to generate the analytical results in (3.5.2) hold only for $d \gg 1$, it can be expected to find larger differences for smaller d which is meet by the results shown in the figure. Furthermore, the dashed line can be expected to show a better approximation to the numerical results, since the rough estimate for $g(\lambda)$ has been used only once while λ_m originates directly from the spectrum. Again, the figure follows the expectation.

In order to understand why the differences between the expected times from the analytical calculation and the measured time from numerical experiments show better agreement for smaller $|m|$, the spectrum has to be considered. Figure 3.4 shows the spectrum of a 20 dimensional hypercube and it can be seen that, coming from $\lambda = 1$, the diagonal line disappears somewhere at $\lambda = 1 \pm 0.6$ and the crossings become more and more inaccurate at higher $|m|$ values. The 3rd crossing already has $\lambda_3 > 1.5$ and is close to the region where the diagonal eigenphase disappears and the analysis breaks down.

For smaller dimensions, the difference between the horizontal lines is larger since the density of eigenphases decreases for there are only $2d$ eigenvectors in the reduced space. A consequence is that this leads to the crossings being further away from $\lambda = 1$ and an earlier breakdown of the analysis. The only crossing that is not affected by this is the crossing for $m = 0$ at $\lambda_0 = 1$, $\omega = 0$ for which the figure agrees very well even for small d.

Figures 3.6 and 3.7 show how the probability to measure $U_\lambda^t |\omega_m\rangle$ at the target vertex $|v\rangle$ changes as a function of t for different m and a fixed number of dimensions, that is $d = 25$. As expected, the localisation occurs faster the higher m. The algorithm for $m = 2$ finds the marked vertex in 1/5th of the time needed for $m = 0$ but there is a price to pay since the walk has a slightly lower probability at its maximum. Regarding the algorithms for higher and higher m in figure 3.7, an optimal search algorithm can be

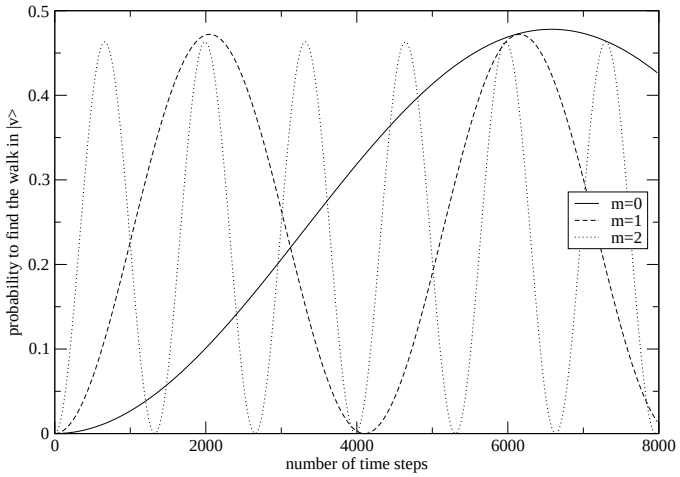

Figure 3.6: The probability to measure the state at the marked vertex in $d = 25$ dimensions for the central crossing and the first two crossings for higher m.

identified before the loss due to amplitude drop is bigger than the gain of a faster search: The maximum of the algorithm using $m = 4$ does not extend the curve for $m = 3$, so there is nothing gained and only the loss due to lower amplitude.

Numerical observations lead to the conclusion that the quantum search algorithm fails to localise at the marked vertex for higher $|m|$ as soon as the gap is of the same magnitude as the distance between two neighbouring eigenvalues next to the crossing.

3.6 Results for the original space $\mathcal{H}$

Unfortunately, the fast algorithms for $m \neq 0$ can not be used in the original space $\mathcal{H}$. To understand why, one has to regard the construction of the reduced Hilbert space. The introduction of the marked vertex at $\vec{v} = \vec{0}$ broke the symmetry of the system and reduced the degeneracy of the eigenvalues. For the then following reduction of the

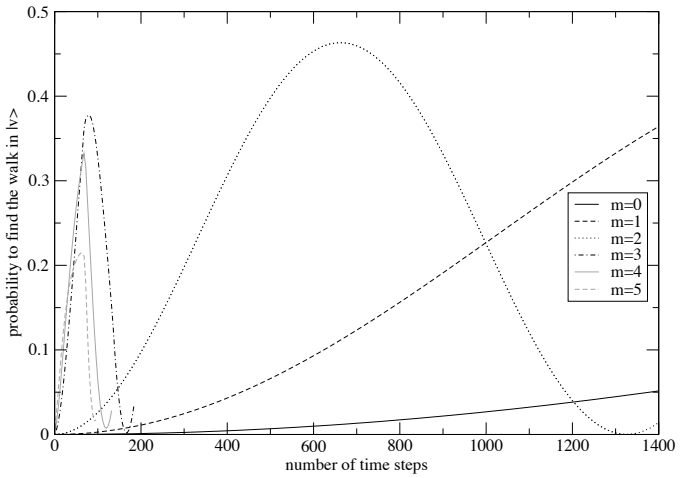

Figure 3.7: Detail of figure 3.6 with more crossings for higher m. To concentrate on the relevant maxima, each curve has been cut off after its first maximum.

space all eigenvectors corresponding to eigenvalues that do not change in λ where dropped. The remaining eigenvectors are given in equation (3.2.5). The $|\omega_m\rangle$ are linear combinations of eigenvectors in the corresponding $e^{i\omega_m}$-eigenspace but there is a phase factor that depends on the position of the target vertex $\vec{v}$, see (3.2.5). Therefore the linear combination is not known for unknown $\vec{v}$. Since the thus defined eigenvector acts as starting vector for the search at the mth crossing, the demanded starting state is not defined for unknown $\vec{v}$. In fact, a search algorithm can be employed to identify the right state in the $\binom{d}{|m|}$ dimensional eigenspace. This can only be done in a time of the order $\sqrt{\binom{d}{|m|}}$, as it is not possible to construct a faster search [23, 24]. To construct such an algorithm, the previously discussed search algorithm can be used as oracle to determine whether a state leads to a successful search or not. However, this search algorithm succeeds not faster than in $\Omega\left(\sqrt{\binom{d}{|m|}}\right)$ time steps and takes away the speed up gained by choosing $m \neq 0$.

This negative result justifies the sloppy analysis for the search algorithms for $m \neq 0$. The only search algorithm that can be used for arbitrary $\vec{v}$ is the one starting in the uniform distribution $|\omega_0\rangle$ for which a careful analysis was presented in this chapter.

To summarise it has been shown in this chapter that the search starting in the uniform distributions leads to a localisation after T time steps, where $T = \text{integer}\,(t)$ and

$$t = \frac{\sqrt{2^{d-1}}\pi}{b\left|e^{i\lambda_0\pi} - 1\right|\sqrt{\binom{d}{0}\beta_0}} + \mathcal{O}(1) \tag{3.6.1}$$

$$= \frac{\sqrt{N}\pi}{4}\sqrt{\frac{2 + \frac{1}{2d} - 3\cdot 2^{-d} + \mathcal{O}\left(e^{-\frac{d}{8}}d\right)}{1 - 2^{-d}}} + \mathcal{O}(1)\,. \tag{3.6.2}$$

After T time steps it has a probability

$$P_{\vec{v}} = \frac{1 - 2^{-d}}{2 + \frac{1}{2d} - 3\cdot 2^{-d} + \mathcal{O}\left(e^{-\frac{d}{8}}d\right)} \tag{3.6.3}$$

to be measured at the target vertex and with probability

$$P_{nn} = \frac{1 - 2^{-d}}{1 + \frac{1}{4d} - 3\cdot 2^{-d-1} + \mathcal{O}\left(e^{-\frac{d}{8}}d\right)} \tag{3.6.4}$$

the walk is found either at the target vertex or at one of its nearest neighbours.

In the case of large d the probabilities reach their maximum and the search algorithm can be optimised if the outcome of the measurement is verified using the oracle once more and check, if the resulting vertex is indeed marked. The search state will be measured at the target vertex with a probability close to $\frac{1}{2}$ and the answer is 'yes'. If the outcome is negative, the target vertex can be found after one additional oracle call as described in section 3.3.5 and without repeating the search. Therefore at most 2 additional oracle calls are needed if the search is stopped and the target vertex will be obtained with a probability close to 1.

Recent result on the hypercube search

Potoček *et al.* have developed an algorithm for the search on the d-dimensional hypercube that demonstrates a speed-up of $\frac{1}{\sqrt{2}}$ compared to the Shenvi Kempe Whaley algorithm discussed here by adding an auxiliary dimension to the coin space [28].

From equation (3.6.2), it follows that the search algorithm introduced by Potoček et al. localises at the marked vertex in

$$t = \frac{\sqrt{N}\pi}{4}\sqrt{\frac{1+\frac{1}{4d}-3\cdot 2^{-d-1}+\mathcal{O}\left(\mathrm{e}^{-\frac{d}{8}}d\right)}{1-2^{-d}}}+\mathcal{O}\left(1\right) \tag{3.6.5}$$

time steps.

CHAPTER 4

Search algorithm on a d-dimensional square lattice

The second model for a search algorithm to be discussed in this thesis, is a search on a d-dimensional square lattice previously discussed by Ambainis, Kempe and Rivosh in [25].

In their paper, Ambainis, Kempe and Rivosh analyse two search algorithms and estimate the order of search time. In this chapter, the more efficient of the two is analysed using the technique of avoided crossings introduced in chapter 3 of this thesis. Like the search algorithm on the hypercube, the lattice search can be regarded as a perturbed quantum random walk where the target vertices are singled out by a marking coin flip.

In this chapter, the method introduced in chapter 3 is applied to the different system and the search time and localisation probability of this search are calculated. That is like before, a family of unitary operators with parameter λ is defined and an avoided crossing is found in the spectrum at $\lambda = 1$ as shown in figure 4.2. Again two approximate eigenvectors near the crossing are being calculated (eqs (4.2.7) and (4.2.18) give their

approximate eigenvalue equations).

However, the approximate eigenvectors lead to a construction of a two dimensional, nearly invariant subspace and again it is shown that this subspace contains starting state and a state localised at the target vertex.

Most of the work in this chapter is spend on the estimate of the normalisation constant b of the approximate eigenvector $|\nu_\lambda\rangle$ finally obtained in equation (4.3.42), which turns out to be crucial for the analysis of the search time (4.4.12) and amplitude of the localised state at the target vertex (4.3.43). As in the previous chapter, the analytical results are compared to some results obtained by numerical simulations, figure 4.4 and are found in good agreement.

4.1 Definition and model

Let the graph be a d-dimensional lattice with n equally partitioned vertices in each dimension, making overall $N = n^d$ vertices, and periodic boundary conditions. The positions of the vertices in the lattice are defined by the set of vectors $|x\rangle = \vec{x} = (x_1, x_2, \ldots, x_d)$, where the coordinates $x_i \in \{0, 1, \ldots, n-1\}$ are integers.

Each vertex has bonds to its $2d$ nearest neighbours, two in each direction, and two vertices at positions $\vec{x}$ and $\vec{y}$ are connected, if and only if, $x_i - y_i = 0$ for all but one i and the absolute value of the difference in the remaining dimension is either 1 or $n-1$.

The quantum walk that will serve as foundation for the search algorithm is basically the same as discussed in section 3.1.1 but adapted to the different graph.

4.1.1 Quantum random walk

The local coin flip σ is defined as in (3.1.1) for the hypercube, that is,

$$\sigma = 2\left|s\right\rangle\left\langle s\right| - \mathbb{1}_{2d}. \tag{4.1.1}$$

As before $|s\rangle$ is defined as the uniform distribution in coin space. Using the vectors $|i^{\pm}\rangle$ as coin space basis vectors pointing to the nearest neighbours located in positive and negative i-direction and $i = 1, \ldots, d$ labels the dimensions, the uniform distribution is defined as $|s\rangle = \frac{1}{\sqrt{2d}} \sum_{i=1}^{d} \left(|i^{+}\rangle + |i^{-}\rangle\right)$. The global coin flip is again defined as the local coin flip applied at all vertices using the tensor product and the identity operator in position space

$$C = \sigma \otimes \mathbb{1}_N. \tag{4.1.2}$$

Using the unit vector in i^{+}-direction $|e_i\rangle$ and $|i^{\pm}x\rangle = |i^{\pm}\rangle \otimes |x\rangle$, the moving shift S is defined according to

$$S = \sum_{\vec{x}} \sum_{i=1}^{d} \left(\left|i^{+}x - e_i\right\rangle\left\langle i^{-}x\right| + \left|i^{-}x + e_i\right\rangle\left\langle i^{+}x\right|\right). \tag{4.1.3}$$

At first glance, S looks counter intuitive as it performs permutations in coin space as well as in position space, but in fact it is a very natural definition for a moving shift as a walker leaving vertex $\vec{x}$ in direction i^{+}, enters vertex $\vec{x} + \vec{e}_i$ from direction i^{-}. This obviously justifies the permutation in coin space.

As discussed in section 2.2, the dimension of the Hilbert space $\mathcal{H}$ of the quantum random walk is the number of vertices times the number of bonds at each vertex. Hence, the d-dimensional lattice has $\dim \mathcal{H} = 2dn^d$.

The quantum random walk itself is defined according to

$$U = SC, \tag{4.1.4}$$

where the first operator applied to the quantum walker is the coin flip C and then the walker is moved to the neighbouring vertices by the shift operator S.

4.1.2 Eigenvectors and eigenvalues of U

Ambainis, Kempe and Rivosh analysed the eigenvalues and eigenvectors of the quantum walk defined in the previous section and in [25]. Some of the properties of eigenvectors and eigenvalues are important for the following analysis, these properties are outlined in this section. A more detailed discussion can be found in appendix C.

Using the tensor product, the eigenvectors can be divided into a vector in coin space $|u\rangle$ and a position space vector $|X\rangle$ according to

$$\left|\phi_{\vec{k}}\right\rangle = |u\rangle \otimes \left|X_{\vec{k}}\right\rangle, \tag{4.1.5}$$

where $\vec{k}$ is a vector with entries $k_i \in \{0, 1, \ldots, n-1\}$. Using the tensor product once more, the vector in position space can be factorised into contributions for each dimension, $\left|X_{\vec{k}}\right\rangle = \bigotimes_{i=1}^{d} \left|\chi_{k_i}\right\rangle$, where the vectors $\left|\chi_{k_i}\right\rangle$ are obtained from the canonical basis vectors of position space using a Fourier transformation

$$\left|\chi_{k_i}\right\rangle = \frac{1}{\sqrt{n}} \sum_{j=0}^{n-1} \omega^{k_i j} |j\rangle \qquad \text{where } \omega = \mathrm{e}^{2\pi \mathrm{i}/n}. \tag{4.1.6}$$

The thus obtained basis provides a convenient way to denote the eigenvectors. Although the set of eigenvectors, especially for arbitrary d, is hard to calculate some properties can be verified. Appendix C.1 analyses the complete set of eigenvalues and discusses important properties of the eigenvectors that are outlined below in more detail.

Only some of the eigenvectors are important for the scope of this thesis which are the ones having a coin space component not orthogonal to $|s\rangle$. These vectors are the 1-eigenvector $|\phi_0\rangle = |s\rangle \otimes \left|X_{\vec{0}}\right\rangle$ which is the uniform distribution and two eigenvectors $\left|\phi_{\vec{k}}^{\pm}\right\rangle = \left|u_{\vec{k}}^{\pm}\right\rangle \otimes \left|X_{\vec{k}}\right\rangle$ for each $\vec{k} \neq \vec{0}$ with complex conjugated eigenvalues $\mathrm{e}^{\pm \mathrm{i}\theta_{\vec{k}}}$, where $\cos\theta_{\vec{k}} = \frac{1}{d}\sum_{i=1}^{d} \cos\frac{2\pi k_i}{n}$.

These $2N-1$ eigenvectors are the ones that are important for the later analysis. Although these eigenvectors are hard to calculate, it is shown in appendix C.2 that $\left\langle s \mid u_{\vec{k}}^{\pm} \right\rangle = \frac{1}{\sqrt{2}}$. The remaining eigenvectors are $(d-1)N+1$ eigenvectors with eigenvalue 1 and $(d-1)N$ eigenvectors with eigenvalue -1.

4.2 The quantum search algorithm

Following the model of the abstract search algorithm in section 2.3.2 and Ambainis, Kempe and Rivosh [25], a target vertex $|v\rangle$ is introduced and the coin flip of the target vertex is changed to $\sigma' = -\mathbb{1}_{2d}$ leading to a perturbed coin flip

$$C' = C - (\sigma - \sigma') \otimes |v\rangle \langle v| . \tag{4.2.1}$$

The effect of this perturbation is analogue to the perturbation discussed for the hypercube in equations (3.1.9 - 3.1.12) and the resulting search algorithm is characterised by

$$U' = U - 2U |sv\rangle \langle sv| . \tag{4.2.2}$$

Figure 4.1 shows the localisation of the probability at the target vertex for four different times, $T = 0,\ 19,\ 38$ and 57. That is, the resulting states after the application of $(U')^T$ on the uniformly distributed state $|\phi_0\rangle$ on a logarithmic scale.

The mechanism behind the Ambainis Kempe Rivosh search algorithm can be analysed following the treatment developed for the hypercube in chapter 3. In analogy to the definition (3.2.1) a unitary one parameter operator U_λ is defined for the d-dimensional lattice:

$$U_\lambda = U + \left(e^{i\pi\lambda} - 1\right) U |sv\rangle \langle sv| . \tag{4.2.3}$$

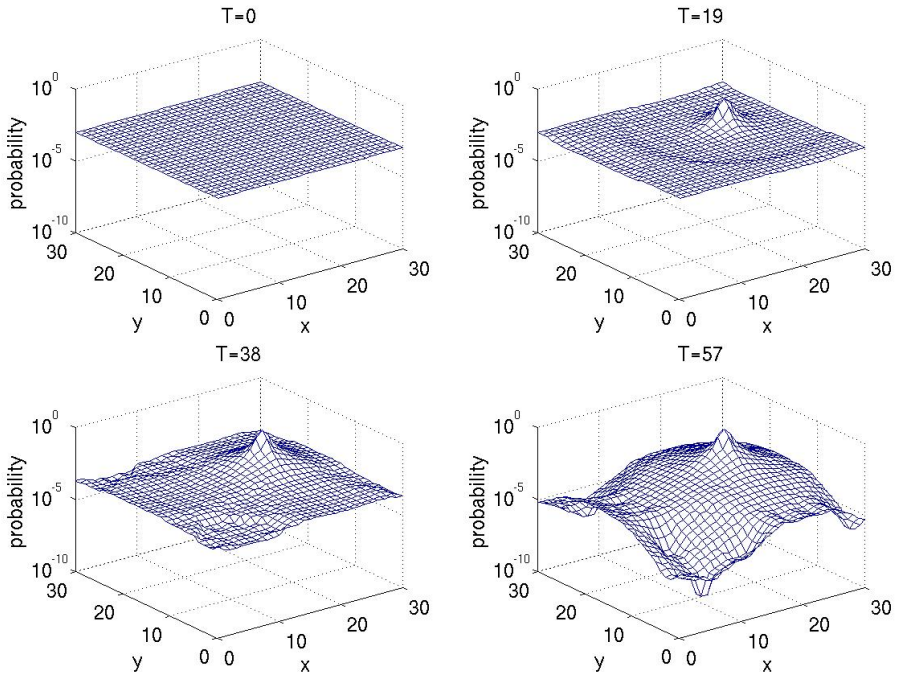

Figure 4.1: Probability distribution of the quantum walk on a 31×31-lattice for several times T (on a logarithmic scale).

4.2.1 Reduced space

A reduced space $\mathcal{H}'$ can be defined to keep the system as simple as possible. This is done by dropping all ± 1-eigenvectors of U expect $|\phi_0\rangle$ and the ones represented by $\left|\phi_{\vec{k}}^{\pm}\right\rangle$ which occur for even n. These vectors can be dropped since they are orthogonal to $|sv\rangle$ and neither eigenvalue nor eigenvector change by a variation of λ. Having done so, a reduced subspace for the problem can be defined. This subspace is the $2N-1$ dimensional Hilbert space $\mathcal{H}'$ spanned by $|\phi_0\rangle$ and $\left|\phi_{\vec{k}}^{\pm}\right\rangle$ for $\vec{k} \neq \vec{0}$.

This reduction is by far not as complete as the reduction for the search on the hypercube in section 3.2.1. For the hypercube the reduction removed all degeneracies from the spectrum while in this case some degeneracies remain. For example the vectors $\left|\phi_{\vec{k}}^{\pm}\right\rangle$ for $\vec{k}_1 = (1,0,0,\dots)$ and $\vec{k}_2 = (0,1,0,\dots)$ give rise to the same eigenvalues $\mathrm{e}^{\pm \mathrm{i}\theta_{\vec{k}}}$.

The scalar product $\left\langle \phi_{\vec{k}}^{\pm} \mid sv \right\rangle$ can be calculated using $\left\langle s \mid u_{\vec{k}}^{\pm} \right\rangle = \frac{1}{\sqrt{2}}$ (appendix C.2)

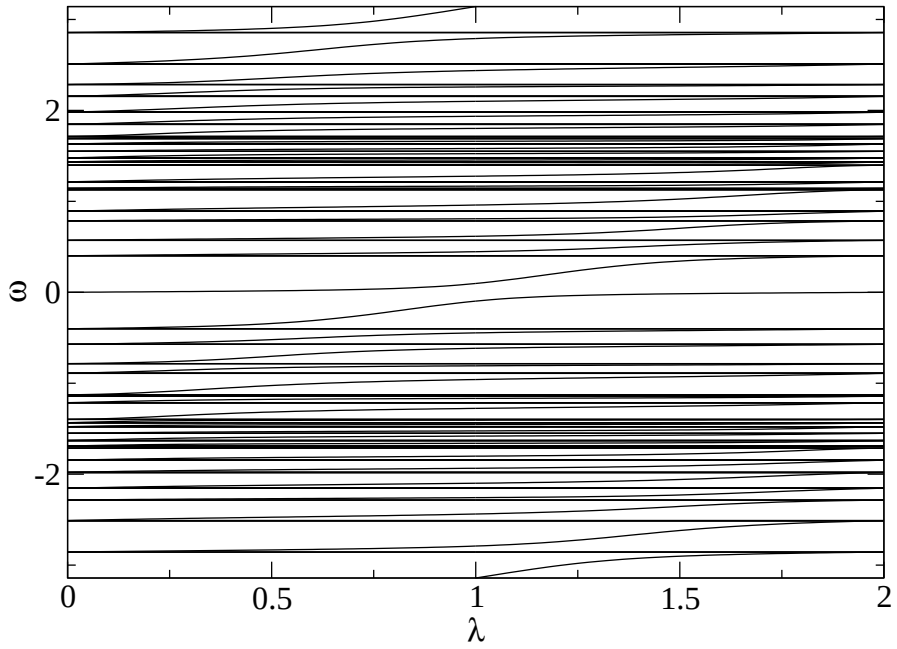

Figure 4.2: The eigenphases as functions of λ for $n = 11$, $d = 2$.

and thus $|sv\rangle$ can be expanded in terms of eigenvectors in the reduced space, that is

$$|sv\rangle = \mathbb{1}_{\mathcal{H}'} |sv\rangle \tag{4.2.4}$$

$$= |\phi_0\rangle \langle \phi_0 \mid sv\rangle + \sum_{\vec{k}\neq\vec{0}} \left(\left|\phi_{\vec{k}}^{+}\right\rangle \left\langle \phi_{\vec{k}}^{+}\right| + \left|\phi_{\vec{k}}^{-}\right\rangle \left\langle \phi_{\vec{k}}^{-}\right| \right) |sv\rangle \tag{4.2.5}$$

$$= \frac{1}{\sqrt{N}} |\phi_0\rangle + \frac{1}{\sqrt{2N}} \sum_{\vec{k}\neq\vec{0}} \omega^{-\vec{k}\vec{v}} \left(\left|\phi_{\vec{k}}^{+}\right\rangle + \left|\phi_{\vec{k}}^{-}\right\rangle \right). \tag{4.2.6}$$

Since $|sv\rangle$ is orthogonal to all eigenvectors of U not contained in $\mathcal{H}'$ this expansion also holds for the non-reduced space.

4.2.2 Approximative eigenvectors of U_λ

The technique to calculate the search time T introduced in chapter 3 for the hypercube search can also be used in order to estimate the search time T for the lattice search

algorithm.

The spectrum of U_λ for a 11×11-lattice in the reduced space $\mathcal{H}'$ is plotted in figure 4.2. Again, the perturbation leads to an avoided crossing in the spectrum. Other than for the $m \neq 0$-crossings in the spectrum of the hypercube search, the coordinates of the crossing $(\lambda = 1, \omega = 0)$ are well known which simplifies the discussion. Following the reasoning in section 3.4 the search time can be estimated using a 2×2 model Hamiltonian H that is constructed using the subspace spanned by the two approximate eigenvectors of U_λ with eigenvalues that participate in the crossing.

Thus, the approximate eigenvectors have to be found. The first guess for one approximative eigenvector is the uniform distribution $|\phi_0\rangle$ and indeed, up to an error term of $\mathcal{O}\left(N^{-1/2}\right)$ this vector is an eigenvector with eigenvalue 1 since

$$U_\lambda |\phi_0\rangle = |\phi_0\rangle + \left(\mathrm{e}^{\mathrm{i}\lambda\pi} - 1\right) U |sv\rangle \frac{1}{\sqrt{N}} \tag{4.2.7}$$

$$= |\phi_0\rangle + \mathcal{O}\left(N^{-1/2}\right). \tag{4.2.8}$$

Since the actions of the unitary operator U and the projection $|sv\rangle \langle sv|$ on the set of eigenvectors of U in the reduced space is well known, the eigenvector corresponding to the line that crosses diagonal through the point $(\lambda = 1, \omega = 0)$ in figure 4.2 can be calculated using an expansion in the eigenvectors of U, that is,

$$|\nu_\lambda\rangle = a_0 |\phi_0\rangle + \sum_{\vec{k} \neq \vec{0}} \left(a^+_{\vec{k}} \left|\phi^+_{\vec{k}}\right\rangle + a^-_{\vec{k}} \left|\phi^-_{\vec{k}}\right\rangle\right) \tag{4.2.9}$$

with a yet unknown set of coefficients a_0 and $a^\pm_{\vec{k}}$, and

$$|a_0|^2 + \sum_{\vec{k} \neq \vec{0}} \left(\left|a^+_{\vec{k}}\right|^2 + \left|a^-_{\vec{k}}\right|^2\right) = 1 \tag{4.2.10}$$

is demanded such that $|\nu_\lambda\rangle$ is normalised.

The condition $U_\lambda |\nu_\lambda\rangle \approx \mathrm{e}^{\mathrm{i}g(\lambda)} |\nu_\lambda\rangle$, that is, the approximate eigenvalue equation with an unknown eigenvalue $\mathrm{e}^{\mathrm{i}g(\lambda)}$ gives a linear system of equations for the $2N - 1$ unknown

coefficients. Since both, $|\nu_\lambda\rangle$ and $|\phi_0\rangle$, are approximate eigenvectors of U_λ the two vectors can be assumed to be orthogonal. Demanding $|\nu_\lambda\rangle \perp |\phi_0\rangle$ is equivalent to $a_0 = 0$.

A straight forward application of U_λ on $|\nu_\lambda\rangle$ results in

$$\begin{aligned} U_\lambda |\nu_\lambda\rangle = &\sum_{\vec{k}\neq\vec{0}} \left(a^+_{\vec{k}} \mathrm{e}^{\mathrm{i}\theta_{\vec{k}}} \left|\phi^+_{\vec{k}}\right\rangle + a^-_{\vec{k}} \mathrm{e}^{-\mathrm{i}\theta_{\vec{k}}} \left|\phi^-_{\vec{k}}\right\rangle \right) + \left(\mathrm{e}^{\mathrm{i}\pi\lambda} - 1\right) \\ &\cdot \left(\frac{1}{\sqrt{N}} |\phi_0\rangle + \frac{1}{\sqrt{2N}} \sum_{\vec{k}\neq\vec{0}} \omega^{-\vec{k}\vec{v}} \left(\mathrm{e}^{\mathrm{i}\theta_{\vec{k}}} \left|\phi^+_{\vec{k}}\right\rangle + \mathrm{e}^{-\mathrm{i}\theta_{\vec{k}}} \left|\phi^-_{\vec{k}}\right\rangle \right) \right) \langle sv \mid \nu_\lambda\rangle . \end{aligned} \tag{4.2.11}$$

For simplicity, a constant

$$b := \langle sv \mid \nu_\lambda\rangle \tag{4.2.12}$$

is defined. The overall phase factor of $|\nu_\lambda\rangle$ can be defined such that $b \geq 0$. Now the right hand side of the approximate eigenvalue equation, that is $\mathrm{e}^{\mathrm{i}g(\lambda)} |\nu_\lambda\rangle$, is added and subtracted to obtain

$$\begin{aligned} U_\lambda |\nu_\lambda\rangle = &\mathrm{e}^{\mathrm{i}g(\lambda)} |\nu_\lambda\rangle + \frac{b\left(\mathrm{e}^{\mathrm{i}\pi\lambda} - 1\right)}{\sqrt{N}} |\phi_0\rangle \\ &+ \sum_{\vec{k}\neq\vec{0}} \left(\left(a^+_{\vec{k}} \left(\mathrm{e}^{\mathrm{i}\theta_{\vec{k}}} - \mathrm{e}^{\mathrm{i}g(\lambda)} \right) + \frac{b\left(\mathrm{e}^{\mathrm{i}\pi\lambda} - 1\right) \omega^{-\vec{k}\vec{v}} \mathrm{e}^{\mathrm{i}\theta_{\vec{k}}}}{\sqrt{2N}} \right) \left|\phi^+_{\vec{k}}\right\rangle \right. \\ &\left. + \left(a^-_{\vec{k}} \left(e^{-\mathrm{i}\theta_{\vec{k}}} - \mathrm{e}^{\mathrm{i}g(\lambda)} \right) + \frac{b\left(\mathrm{e}^{\mathrm{i}\pi\lambda} - 1\right) \omega^{-\vec{k}\vec{v}} \mathrm{e}^{-\mathrm{i}\theta_{\vec{k}}}}{\sqrt{2N}} \right) \left|\phi^-_{\vec{k}}\right\rangle \right). \end{aligned} \tag{4.2.13}$$

Demanding the sum to be zero defines the coefficients

$$a^\pm_{\vec{k}} = \frac{b\left(\mathrm{e}^{\mathrm{i}\pi\lambda} - 1\right) \omega^{-\vec{k}\vec{v}} \mathrm{e}^{\pm\mathrm{i}\theta_{\vec{k}}}}{\sqrt{2N}\left(\mathrm{e}^{\mathrm{i}g(\lambda)} - e^{\pm\mathrm{i}\theta_{\vec{k}}}\right)} \tag{4.2.14}$$

and gives the error term of the eigenvalue equation as $\frac{b\left(\mathrm{e}^{\mathrm{i}\pi\lambda}-1\right)}{\sqrt{N}} |\phi_0\rangle$. Therefore the error term is $\mathcal{O}\left(\frac{b}{\sqrt{N}}\right)$ and due to $b \leq 1$ it can be considered small.

Using the set of coefficients $a^\pm_{\vec{k}}$ from (4.2.14) the constant b appears on both sides of $b = \langle sv \mid \nu_\lambda\rangle$. Hence the system of equations has a solution, if and only if

$$b = \langle sv \mid \nu_\lambda\rangle \tag{4.2.15}$$

is at least approximatively fulfilled at the crossing. Using equations (4.2.6) and (4.2.14), the condition can be denoted as

$$1 = \frac{\left(\mathrm{e}^{\mathrm{i}\pi\lambda} - 1\right)}{2N} \sum_{\vec{k}\neq\vec{0}} \left(\frac{\mathrm{e}^{\mathrm{i}\theta_{\vec{k}}}}{\mathrm{e}^{\mathrm{i}g(\lambda)} - e^{\mathrm{i}\theta_{\vec{k}}}} + \frac{\mathrm{e}^{-\mathrm{i}\theta_{\vec{k}}}}{\mathrm{e}^{\mathrm{i}g(\lambda)} - e^{-\mathrm{i}\theta_{\vec{k}}}} \right). \tag{4.2.16}$$

The crossing occurs at $\lambda = 1$ and $\mathrm{e}^{\mathrm{i}g(1)} = 1$ for which the sum formula (4.2.16) can be calculated explicitly, that is,

$$1 = \frac{-2}{2N} \sum_{\vec{k}\neq\vec{0}} (-1) = \frac{1}{N} (N-1) = 1 - \frac{1}{N}. \tag{4.2.17}$$

Therefore no exact solution exists. This is not surprising as $|\nu_\lambda\rangle$ is not an exact eigenvector. Since the error term $\frac{1}{N}$ is small and decays to 0 as $N \to \infty$, equation (4.2.17) holds in good approximation and it is verified that the set of coefficients from equation (4.2.14) define an approximate solution of the set of coupled equations given in (4.2.14). Therefore, the vector defined as

$$\begin{aligned} |\nu_\lambda\rangle =& \frac{b\left(\mathrm{e}^{\mathrm{i}\pi\lambda} - 1\right)}{\sqrt{2N}} \sum_{\vec{k}\neq\vec{0}} \omega^{-\vec{k}\vec{v}} \\ & \cdot \left(\frac{\mathrm{e}^{\mathrm{i}\theta_{\vec{k}}}}{\mathrm{e}^{\mathrm{i}g(\lambda)} - e^{\mathrm{i}\theta_{\vec{k}}}} \left|\phi_{\vec{k}}^{+}\right\rangle + \frac{\mathrm{e}^{-\mathrm{i}\theta_{\vec{k}}}}{\mathrm{e}^{\mathrm{i}g(\lambda)} - e^{-\mathrm{i}\theta_{\vec{k}}}} \left|\phi_{\vec{k}}^{-}\right\rangle \right), \end{aligned} \tag{4.2.18}$$

fulfils the approximate eigenvalue equation

$$U_\lambda |\nu_\lambda\rangle = \mathrm{e}^{\mathrm{i}g(\lambda)} |\nu_\lambda\rangle + \frac{b\left(\mathrm{e}^{\mathrm{i}\pi\lambda} - 1\right)}{\sqrt{N}} |\phi_0\rangle \tag{4.2.19}$$

$$= \mathrm{e}^{\mathrm{i}g(\lambda)} |\nu_\lambda\rangle + \mathcal{O}\left(\frac{1}{\sqrt{N}}\right) \tag{4.2.20}$$

and can be considered an approximate eigenvector of U_λ with eigenvalue $\mathrm{e}^{\mathrm{i}g(\lambda)}$, where $\mathrm{e}^{\mathrm{i}g(1)} = 1$.

The only unknown in $|\nu_\lambda\rangle$ is the constant b as it dropped out of the equations. Since the vector $|\nu_\lambda\rangle$ is demanded to be normalised b actually turns out to be the normalisation constant.

4.3 Normalisation of the approximated crossing eigenvector

The normalisation $|b|$ of the approximate eigenvector of $|v_\lambda\rangle$ is important for estimating the search time because the normalisation of the vectors is crucial for evaluating the matrix elements of the 2×2 model for U_λ.

To determine the normalisation requires a lengthy and technical calculation. First the expression for the normalisation will be expanded in a sum and then the first terms will be approached one by one. The result, up to leading order, for $d = 2$ and 3 and an estimate of the scaling for $d \geq 4$ is finally presented in equation (4.3.42). In principle, the leading order term for the normalisation constants for $d \geq 4$ can be obtained using numerical integration methods as in the $d = 3$ case.

Demanding normalisation of $|v_\lambda\rangle$ and regarding $\lambda = 1$ and $e^{ig^{(1)}} = 1$ only, results in

$$\frac{1}{|b|^2} = \frac{4}{2N} \sum_{\vec{k} \neq \vec{0}} \frac{2}{\left|1 - e^{i\theta_{\vec{k}}}\right|^2} \tag{4.3.1}$$

$$= \frac{2}{N} \sum_{\vec{k} \neq \vec{0}} \frac{1}{1 - \cos\theta_{\vec{k}}} \tag{4.3.2}$$

$$= \frac{2d}{N} \sum_{\vec{k} \neq \vec{0}} \frac{1}{d - \sum_{i=1}^{d} \cos\frac{2\pi k_i}{n}}, \tag{4.3.3}$$

where the sum over all vectors $\vec{k}$ in a d-dimensional cube has to be calculated for $k_i \in \{0, 1, \ldots, n-1\}$, $\vec{k} \neq \vec{0}$. This can be rearranged to a summation over lower dimensional objects where only a limited number of entries of $\vec{k}$ are not 0.

That is, all one dimensional edges, two dimensional faces, 3-dimensional cubes etc. with k_i varying from 1 to $n-1$. The edges are obtained by choosing $d-1$ entries of $\vec{k}$ equal 0, while the remaining entry varies from 1 to $n-1$, the faces have $d-2$ entries of $\vec{k}$ equal 0 and two in $[1, n-1]$ and higher dimensions correspondingly. This new arrangement

results in

$$\begin{aligned}
\frac{1}{|b|^2} =& \frac{2d}{N} \sum_{\vec{k}\neq\vec{0}} \frac{1}{d - \sum_{i=1}^{d} \cos\frac{2\pi k_i}{n}} && (4.3.4)\\
=& \frac{2d}{N} \underbrace{d}_{\text{number of edges}} \underbrace{\sum_{j_1=1}^{n-1} \frac{1}{1 - \sum_{l=1}^{1} \cos\frac{2\pi j_l}{n}}}_{\text{contribution of one edge}} \\
&+ \frac{2d}{N} \underbrace{\binom{d}{2}}_{\text{number of faces}} \underbrace{\sum_{j_1,j_2=1}^{n-1} \frac{1}{2 - \sum_{l=1}^{2} \cos\frac{2\pi j_l}{n}}}_{\text{contribution of one face}} \\
&+ \dots && (4.3.5)\\
=& \frac{2d}{N} \sum_{i=1}^{d} \binom{d}{i} \sum_{j_1,j_2,\dots,j_i=1}^{n-1} \frac{1}{i - \sum_{l=1}^{i} \cos\frac{2\pi j_l}{n}} && (4.3.6)\\
=& \frac{2d}{N} \sum_{i=1}^{d} \binom{d}{i} \left(\frac{n}{\pi}\right)^i I_i, && (4.3.7)
\end{aligned}$$

where in the last step $I_i := \left(\frac{\pi}{n}\right)^i \sum_{j_1,\dots,j_i=1}^{n-1} \left(i - \sum_{l=1}^{i} \cos\frac{2\pi j_l}{n}\right)^{-1}$ was defined.

To obtain the normalisation constant $|b|$ for the approximate eigenvector of the search on the d-dimensional lattice, all sums from I_1 to I_d have to be calculated.

The identity $1 - \cos x = 2\sin^2\frac{x}{2}$ can be used to simplify these sums. One obtains

$$I_i = \frac{1}{2}\left(\frac{\pi}{n}\right)^i \sum_{j_1,\dots,j_i=1}^{n-1} \left(\sum_{l=1}^{i} \sin^2\frac{\pi j_l}{n}\right)^{-1}. \tag{4.3.8}$$

In section 3.3.2 Poisson's summation formula was used to approximate a sum by an integral. The same can be done for the I_i's. First each sum is replaced by an integration and a sum over delta functions is added such that the integrand contributes only at the old j_l's

$$\begin{aligned}
I_i = \frac{1}{2}\left(\frac{\pi}{n}\right)^i & \int_{\frac{1}{2}}^{n-\frac{1}{2}} dx_1 \sum_{m_1=-\infty}^{\infty} \delta(x_1 - m_1) \dots \\
& \int_{\frac{1}{2}}^{n-\frac{1}{2}} dx_i \sum_{m_i=-\infty}^{\infty} \delta(x_i - m_i) \left(\sum_{l=1}^{i} \sin^2\frac{\pi x_l}{n}\right)^{-1}. \qquad (4.3.9)
\end{aligned}$$

In a second step, the sums of delta functions can again be replaced by their Fourier series $\sum_{m_j=-\infty}^{\infty} \delta\left(x_j - m_j\right) = \sum_{m_j=-\infty}^{\infty} \mathrm{e}^{2\pi \mathrm{i} m_j x_j}$. This results in

$$\begin{aligned} I_i =& \frac{1}{2}\left(\frac{\pi}{n}\right)^i \int_{\frac{1}{2}}^{n-\frac{1}{2}} dx_1 \sum_{m_1=-\infty}^{\infty} \mathrm{e}^{2\pi \mathrm{i} m_1 x_1} \dots \\ & \int_{\frac{1}{2}}^{n-\frac{1}{2}} dx_i \sum_{m_i=-\infty}^{\infty} \mathrm{e}^{2\pi \mathrm{i} m_i x_i} \left(\sum_{l=1}^{i} \sin^2 \frac{\pi x_l}{n}\right)^{-1} \qquad (4.3.10) \\ =& \frac{1}{2} \int_{\frac{\pi}{2n}}^{\pi-\frac{\pi}{2n}} dy_1 \dots \int_{\frac{\pi}{2n}}^{\pi-\frac{\pi}{2n}} dy_i \\ & \sum_{m_1=-\infty}^{\infty} \dots \sum_{m_i=-\infty}^{\infty} \frac{\mathrm{e}^{2\mathrm{i}n(m_1 y_1 + \dots + m_i y_i)}}{\sum_{l=1}^{i} \sin^2 y_l}, \qquad (4.3.11) \end{aligned}$$

where the last equality is due to a reordering of terms and the substitution $y_j = \frac{\pi}{n} x_j$ for all integrals.

Using the same arguments as before (section 3.3.2), the contributions are neglected if one of the m_j's is not zero, since the exponential function leads to rapid oscillations if $n \gg 1$. Hence only the $m_j = 0$ terms are taken into account and for $n \gg 1$

$$I_i \approx \frac{1}{2} \int_{\frac{\pi}{2n}}^{\pi-\frac{\pi}{2n}} dy_1 \dots \int_{\frac{\pi}{2n}}^{\pi-\frac{\pi}{2n}} dy_i \frac{1}{\sum_{l=1}^{i} \sin^2 y_l}. \qquad (4.3.12)$$

In the remainder of this chapter equality will be assumed.

Using the symmetry of the sine squared, the integration simplifies to

$$I_i = 2^{i-1} \int_{\frac{\pi}{2n}}^{\frac{\pi}{2}} dy_1 \dots \int_{\frac{\pi}{2n}}^{\frac{\pi}{2}} dy_i \frac{1}{\sum_{l=1}^{i} \sin^2 y_l} \qquad (4.3.13)$$

and the integrations can finally be calculated.

4.3.1 First integration I_1

The first integration is quickly done and one obtains

$$I_1 = \int_{\frac{\pi}{2n}}^{\frac{\pi}{2}} dy \frac{1}{\sin^2 y} = \left[-\cot y\right]_{y=\frac{\pi}{2n}}^{\frac{\pi}{2}} = \cot \frac{\pi}{2n}. \qquad (4.3.14)$$

4.3.2 Second integration I_2

The second integration

$$I_2 = 2\int_{\frac{\pi}{2n}}^{\frac{\pi}{2}} dx \int_{\frac{\pi}{2n}}^{\frac{\pi}{2}} dy \frac{1}{\sin^2 x + \sin^2 y} \tag{4.3.15}$$

is by far more difficult. It is important to note that the integrand is symmetric with respect to exchanged x and y and therefore

$$I_2 = 4\int_{\frac{\pi}{2n}}^{\frac{\pi}{2}} dx \int_{x}^{\frac{\pi}{2}} dy \frac{1}{\sin^2 x + \sin^2 y}. \tag{4.3.16}$$

Now the y integration can be solved by observing that

$$\frac{d}{dx} \frac{\arctan\left(\frac{\tan y}{\tan x}\sqrt{2\tan^2 x + 1}\right)}{\sin x \sqrt{\sin^2 x + 1}} = \frac{1}{\sin^2 x + \sin^2 y}. \tag{4.3.17}$$

Thus

$$I_2 = 4\int_{\frac{\pi}{2n}}^{\frac{\pi}{2}} dx \left[\frac{\arctan\left(\frac{\tan y}{\tan x}\sqrt{2\tan^2 x + 1}\right)}{\sin x \sqrt{\sin^2 x + 1}}\right]_{y=x}^{\frac{\pi}{2}} \tag{4.3.18}$$

$$= 4\int_{\frac{\pi}{2n}}^{\frac{\pi}{2}} dx \frac{\frac{\pi}{2} - \arctan\left(\sqrt{2\tan^2 x + 1}\right)}{\sin x \sqrt{\sin^2 x + 1}}. \tag{4.3.19}$$

The ansatz $I_2 = 4\int_{\frac{\pi}{2n}}^{\frac{\pi}{2}} dx f g'$ and the functions

$$f = \frac{\pi}{2} - \arctan\sqrt{2\tan^2 x + 1} \tag{4.3.20}$$

and

$$g = -\ln\left(\sqrt{2 + \cot^2 x} + \cot x\right) \tag{4.3.21}$$

are used to integrate by parts

$$\begin{aligned} I_2 =& 4\ln\left(\sqrt{2 + \cot^2\frac{\pi}{2n}} + \cot\frac{\pi}{2n}\right)\left(\frac{\pi}{2} - \arctan\sqrt{2\tan^2\frac{\pi}{2n} + 1}\right) \\ & - 4\int_{\frac{\pi}{2n}}^{\frac{\pi}{2}} dx \frac{\ln\left(\sqrt{2 + \cot^2 x} + \cot x\right)}{\sqrt{2 + \cot^2 x}}. \end{aligned} \tag{4.3.22}$$

This is simplified using the substitution $z = \tan x$ and

$$\begin{aligned} I_2 =& 4 \ln \left(\sqrt{2 + \cot^2 \frac{\pi}{2n}} + \cot \frac{\pi}{2n} \right) \left(\frac{\pi}{2} - \arctan \sqrt{2 \tan^2 \frac{\pi}{2n} + 1} \right) \\ & - 4 \int_{\tan \frac{\pi}{2n}}^{\infty} dz \frac{z \left(\ln \left(1 + \sqrt{2z^2 + 1} \right) - \ln z \right)}{\sqrt{2z^2 + 1} \left(1 + z^2 \right)}. \end{aligned} \tag{4.3.23}$$

This last integration can finally be expanded[1] to

$$I_2 = \pi \ln n + \pi \ln \frac{4}{\pi} - 2K - \frac{\pi}{2} \ln 2 + \mathcal{O}\left(\frac{1}{n^2} \right), \tag{4.3.24}$$

where $K \approx 0.916$ is Catalan's constant.

Normalisation for $d = 2$

The formula for the normalisation is given in (4.3.7) and the result for the $d = 2$ dimensional lattice is

$$\frac{1}{|b|^2} = \frac{4}{N} \sum_{i=1}^{2} \binom{2}{i} \left(\frac{n}{\pi} \right)^i I_i \tag{4.3.25}$$

$$= \frac{4}{N} \left(2 \frac{n}{\pi} I_1 + \frac{N}{\pi^2} I_2 \right) \tag{4.3.26}$$

$$= \frac{8}{\pi n} \cot \frac{\pi}{2n} + \frac{4}{\pi} \ln n + \frac{4}{\pi} \ln \frac{4}{\pi} - \frac{8K}{\pi^2} - \frac{2}{\pi} \ln 2 + \mathcal{O}\left(\frac{1}{n^2} \right) \tag{4.3.27}$$

$$= \frac{2}{\pi} \ln N + \frac{16}{\pi^2} + \frac{4}{\pi} \ln \frac{4}{\pi} - \frac{8K}{\pi^2} - \frac{2}{\pi} \ln 2 + \mathcal{O}\left(\frac{1}{N} \right). \tag{4.3.28}$$

4.3.3 Third integration I_3

For the third integration only the limit $n \to \infty$ will be evaluated. Starting with

$$I_3 = 4 \int_{\frac{\pi}{2n}}^{\frac{\pi}{2}} dx \int_{\frac{\pi}{2n}}^{\frac{\pi}{2}} dy \int_{\frac{\pi}{2n}}^{\frac{\pi}{2}} dz \frac{1}{\sin^2 x + \sin^2 y + \sin^2 z} \tag{4.3.29}$$

[1] The expansion was calculated by MATHEMATICA.

a substitution $x' = \tan x$ in all three variables leads to

$$I_3 = 4 \int_{\tan \frac{\pi}{2n}}^{\infty} dx \int_{\tan \frac{\pi}{2n}}^{\infty} dy \int_{\tan \frac{\pi}{2n}}^{\infty} dz \frac{1}{x^2 (1+y^2)(1+y^2) + (1+x^2) y^2 (1+z^2) + (1+x^2)(1+y^2) z^2} \tag{4.3.30}$$

which converges to[2]

$$\lim_{n \to \infty} I_3 = 15.672\ldots. \tag{4.3.31}$$

Thus $I_3 = 15.672 \cdots + f(n)$ where $\lim_{n\to\infty} f(n) = 0$.

Normalisation for $d = 3$

The result for the normalisation of the $d = 3$ dimensional lattice is computed as

$$\frac{1}{|b|^2} = \frac{6}{N} \sum_{i=1}^{3} \binom{3}{i} \left(\frac{n}{\pi}\right)^i I_i \tag{4.3.32}$$

$$= \frac{6}{N} \left(\frac{n}{\pi}\right)^3 I_3 + \mathcal{O}\left(\frac{\ln N}{n}\right) \tag{4.3.33}$$

$$= \frac{6}{\pi^3} I_3 + \mathcal{O}\left(\frac{\ln N}{n}\right) \tag{4.3.34}$$

$$= \Theta(1). \tag{4.3.35}$$

4.3.4 Higher integrations and normalisations for $d > 3$

Although some results used for this section will not be proven until section 4.4, the calculation shall be presented here because it completes the results of the previous sections by providing an estimate for b for dimensions $d \geq 4$.

Since the integrations get more and more difficult, this section aims at proving that T scales like $\sqrt{N}$ for $d > 3$. It will be shown in section 4.4, equation (4.4.11), that the search time for arbitrary d is given by $T = \frac{\pi\sqrt{N}}{4b}$. This later result and the lower bound for

[2]This result was obtained using MAPLE.

the scaling $T = \Omega\left(\sqrt{N}\right)$ proven by Grover and Bennett *et. al.* [23, 24], lead to a lower bound for the scaling of $\frac{1}{|b|^2}$ for any d. One obtains $\frac{1}{|b|^2} = \Omega(1)$. It remains to show that this bound is tight for $d > 3$ by proving the existence of an upper bound that is independent of N and leads to $\frac{1}{|b|^2} = \mathcal{O}(1)$.

This can be shown by induction.

- By (4.3.35) it is evident that $\frac{1}{|b|^2} = \mathcal{O}(1)$ holds for $d = 3$.
- Assuming that $\frac{1}{|b|^2} = \mathcal{O}(1)$ holds for some $d \geq 3$, it follows by

$$\frac{1}{|b|^2} = 2d \sum_{i=1}^{d} \binom{d}{i} \frac{n^{i-d}}{\pi^i} I_i \tag{4.3.36}$$

  (equation (4.3.7)) that $I_d = \mathcal{O}(1)$.

- It remains to show that $\frac{1}{|b|^2} = \mathcal{O}(1)$ also holds for $d + 1$.

  The starting point is again provided by equation (4.3.7) but this time for for $d + 1$, that is,

$$\frac{1}{|b|^2} = 2(d+1)\left(\sum_{i=1}^{d} \binom{d+1}{i} \frac{n^{i-d-1}}{\pi^i} I_i + \frac{1}{\pi^{d+1}} I_{d+1}\right). \tag{4.3.37}$$

  The sum of the first d terms adds to a leading order of $\frac{1}{n}$, since $I_i = \mathcal{O}(1)$ for $3 \leq i \leq d$ and the lower order terms $I_2 = \Theta(\ln N)$ and $I_1 = \Theta(n)$ have prefactors n^{1-d} and n^{-d}, respectively. Therefore these contributions vanish for $n \to \infty$.

  It remains to show that $I_{d+1} = \mathcal{O}(1)$ and this can be done using equation (4.3.13), that is,

$$I_{d+1} = 2^d \int_{\frac{\pi}{2n}}^{\frac{\pi}{2}} dy_1 \ldots \int_{\frac{\pi}{2n}}^{\frac{\pi}{2}} dy_{d+1} \frac{1}{\sum_{l=1}^{d} \sin^2 y_l + \sin^2 y_{d+1}}. \tag{4.3.38}$$

  As the sines are greater than zero, an upper bound is obtained by dropping the last one. Now the y_{d+1} integration can be performed

$$I_{d+1} \leq 2^d \left(\frac{\pi}{2} - \frac{\pi}{2n}\right) \int_{\frac{\pi}{2n}}^{\frac{\pi}{2}} dy_1 \ldots \int_{\frac{\pi}{2n}}^{\frac{\pi}{2}} dy_d \frac{1}{\sum_{l=1}^{d} \sin^2 y_l} \tag{4.3.39}$$

$$= \left(\pi - \frac{\pi}{n}\right) I_d \tag{4.3.40}$$

and this results in $I_{d+1} = \mathcal{O}(1)$. Thus, the normalisation constant is of the demanded order $\frac{1}{|b|^2} = \mathcal{O}(1)$.

Lower and upper bound together prove that the scaling of b for $d \geq 3$ dimensions is characterised by

$$\frac{1}{b^2} = \Theta(1) \tag{4.3.41}$$

and the scaling of the number of time steps is $T = \Theta\left(\sqrt{N}\right)$.

Thus, the overall result for the normalisation constant is

$$\frac{1}{b^2} = \begin{cases} \frac{2}{\pi}\ln N + \frac{16}{\pi^2} + \frac{4}{\pi}\ln\frac{4}{\pi} - \frac{8K}{\pi^2} - \frac{2}{\pi}\ln 2 + \mathcal{O}\left(\frac{1}{N}\right) & \text{for } d = 2 \\ \frac{6}{\pi^3} I_3 + \mathcal{O}\left(\frac{\ln N}{\sqrt[3]{N}}\right) & \text{for } d = 3\,, \\ \Theta(1) & \text{for } d \geq 4 \end{cases} \tag{4.3.42}$$

where $K \approx 0.916$ is Catalans constant and I_3 has a leading order of $I_3 \approx 15.672$ (4.3.31). It is important to note that the result for $d \geq 4$ relies on equation (4.4.11) which was used to calculate the bounds for $d \geq 4$. Therefore the scaling for $d \geq 4$ remains unclear until it has been shown that equation (4.4.11) holds.

Since it has been shown that for $i \geq 3$, the leading order of I_i in the limit $n \to \infty$ does not depend on n. Thus it can be calculated from equation (4.3.13) by replacing $\frac{\pi}{2n}$ with 0 and using numerical integration methods.

4.3.5 Success probability

Since the model introduced in section 3.4 leads to a localisation on $|\nu_\lambda\rangle$, the scalar product in (4.2.12) is crucial to estimate how much probability will be localised at the target vertex $|v\rangle$. $b = \langle sv \mid \nu_{\lambda=1}\rangle$ does give a lower bound only as there is no need for the vector $|\nu_\lambda\rangle$ to have a coin space contribution parallel to $|s\rangle$. Therefore the probability to measure $|\nu_\lambda\rangle$ at the target vertex $|v\rangle$ is greater or equal b^2.

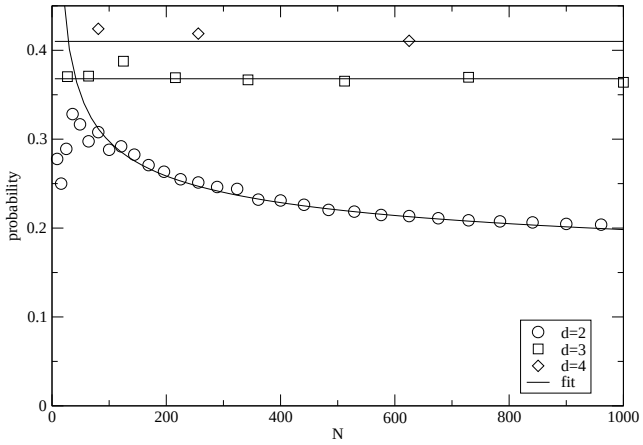

Figure 4.3: Numerical results for the maximal probability at the target vertex as a function of N for $d = 2, 3$ and 4. The fitted curve is a one-parameter fit with a constant times the scaling given in equation (4.3.43).

From equation (4.3.42), it is known

$$b = \begin{cases} \frac{1}{\sqrt{\frac{2}{\pi}\ln N + \frac{16}{\pi^2} + \frac{4}{\pi}\ln\frac{4}{\pi} - \frac{8K}{\pi^2} - \frac{2}{\pi}\ln 2 + \mathcal{O}\left(\frac{1}{N}\right)}} = \Theta\left(\frac{1}{\sqrt{\ln N}}\right) & \text{for } d = 2 \\ \frac{1}{\sqrt{\frac{6}{\pi^3} I_3 + \mathcal{O}\left(\frac{\ln N}{\sqrt[3]{N}}\right)}} = \Theta(1) & \text{for } d = 3 \\ \Theta(1) & \text{for } d \geq 4 \end{cases} . \tag{4.3.43}$$

The localisation at the target vertex is not N-dependent for $d \geq 3$, but in the $d = 2$ case the amplitude of $|\nu_\lambda\rangle$ on $|v\rangle$ may decrease like $\frac{1}{\sqrt{\ln N}}$. However, the state $|\nu_{\lambda=1}\rangle$ is localised on the marked vertex for $d = 2$ since the average amplitude of an eigenstate of the random walk U deceases like $\frac{1}{\sqrt{N}}$. Amplitude amplification methods can be used to increase the success probability to a constant in N by repeating the search algorithm $\mathcal{O}\left(\sqrt{\ln N}\right)$ times [59].

Numerical results of how the probability to find the walk at the target vertex scales as a function of N are shown in figure 4.3. It can be expected that b gives a lower bound for

the scaling only as it measures the amplitude in the coin space state $|s\rangle$ at $|v\rangle$. However, the estimates represent the behaviour of the localisation on the target vertex very well.

The shape of the localised state $|\nu_{\lambda=1}\rangle$ on a 31×31 lattice can be seen in figure 4.1.

Like for the hypercube, section 3.3.5, the probability in $|sv\rangle$ is of the same order as the probability in $U|sv\rangle$ as

$$\langle v_{\lambda=1} \mid U \mid sv\rangle = -\frac{2b}{2N} \sum_{\vec{k}\neq\vec{0}} \left(\frac{1}{\mathrm{e}^{-\mathrm{i}g(1)} - e^{\mathrm{i}\theta_{\vec{k}}}} + \frac{1}{\mathrm{e}^{-\mathrm{i}g(1)} - e^{-\mathrm{i}\theta_{\vec{k}}}} \right) \tag{4.3.44}$$

$$= -b\left(1 - \frac{1}{N}\right). \tag{4.3.45}$$

To obtain the last equality $\mathrm{e}^{\mathrm{i}g(1)} = 1$ has been used. Due to the definition of the shift operator S (4.1.3), the state $U|sv\rangle$ is localised on the vertex adjacent to the target vertex and the state in the coin space points into the direction of the target vertex.

4.4 Model in the two-level subspace.

In the neighbourhood of the crossing, the dynamics of the search algorithm can be described in terms of the two-level subspace spanned by the orthogonal vectors $|\Phi_0\rangle$ and $|\nu_\lambda\rangle$ which are identified with the standard basis vectors $|1\rangle$ and $|2\rangle$ respectively.

Let the 2×2 Hamiltonian H of the search be defined as

$$H = \begin{pmatrix} -\omega & \epsilon\,\mathrm{e}^{\mathrm{i}\delta} \\ \epsilon\,\mathrm{e}^{-\mathrm{i}\delta} & -\omega \end{pmatrix}, \tag{4.4.1}$$

where ω is the phase of the eigenvalue at the crossing, that is, an integer multiple of 2π since $\mathrm{e}^{\mathrm{i}\omega} \stackrel{!}{=} 1$ and the entry $\epsilon\,\mathrm{e}^{-\mathrm{i}\delta}$ is defined such that $\epsilon, \delta \in \mathbb{R}$. ϵ is a small constant enabling the coupling of the two states. Following the calculation in section 3.4 this

ansatz leads to the 2×2 model of U_λ

$$U_\lambda^{2\times 2} = \begin{pmatrix} \cos\epsilon & -\mathrm{i}e^{\mathrm{i}\delta}\sin\epsilon \\ -\mathrm{i}e^{-\mathrm{i}\delta}\sin\epsilon & \cos\epsilon \end{pmatrix}, \tag{4.4.2}$$

where the matrix elements can be calculated explicitly.

The eigenvalues and eigenvectors of H are discussed in section 3.4 and it has been shown that the walk localises on the perturber state $|\nu_\lambda\rangle$ in a time $t = \frac{\pi}{2\epsilon}$ by applying $\left(U_\lambda^{2\times 2}\right)^t$ to the starting state, that is, the uniform distribution

$$\left(U_\lambda^{2\times 2}\right)^t |\text{start}\rangle = -\mathrm{i}e^{-\mathrm{i}\delta} |\nu_\lambda\rangle \tag{4.4.3}$$

as shown in section 3.4.1.

Since $t = \frac{\pi}{2\epsilon}$ does not necessarily need to be integer, the number of time steps necessary for localisation is denoted by T and defined as the integer closest to t.

The calculation of matrix elements for $\lambda = 1$ uses equations (4.2.7), (4.2.19) and (4.2.6), and results in

$$\langle \phi_0 \mid U_1 \mid \phi_0 \rangle = 1 - \frac{2}{N} \tag{4.4.4}$$

$$\langle \nu_1 \mid U_1 \mid \nu_1 \rangle = 1 \tag{4.4.5}$$

for the diagonal elements and

$$\langle \phi_0 \mid U_1 \mid \nu_1 \rangle = \frac{-2b}{\sqrt{N}} \tag{4.4.6}$$

$$\langle \nu_1 \mid U_1 \mid \phi_0 \rangle = \frac{-2}{\sqrt{N}} \langle \nu_1 \mid U \mid sv \rangle \tag{4.4.7}$$

$$= \frac{2b}{\sqrt{N}} \left(1 - \frac{1}{N}\right) \tag{4.4.8}$$

for the off-diagonal entries, where (4.3.45) has been used for the last equality.

Comparing the results for the matrix elements to those of equation (4.4.2), the constants

$$e^{\mathrm{i}\delta} = -\mathrm{i} \qquad \text{and} \qquad \epsilon = \frac{2b}{\sqrt{N}} + \mathcal{O}\left(\frac{1}{N}\right) \tag{4.4.9}$$

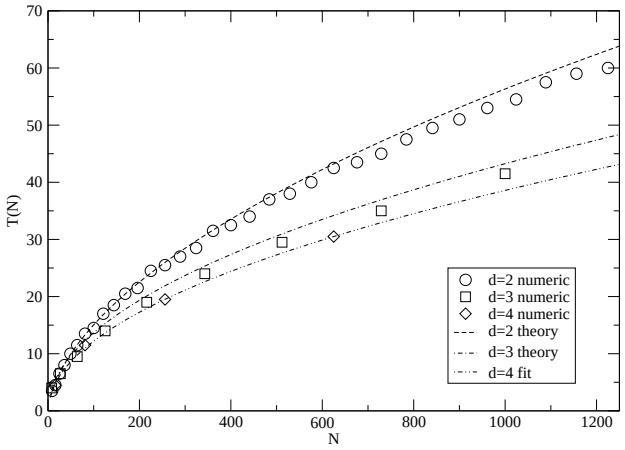

Figure 4.4: The localisation time t after which the walk is localised at the marked vertex for several N. Numerical result compared to analytical for $d = 2, 3$ and 4.

are obtained and the Hamiltonian results in

$$H = \begin{pmatrix} -\omega & -\mathrm{i}\epsilon \\ \mathrm{i}\epsilon & -\omega \end{pmatrix}. \tag{4.4.10}$$

Therefore the time of the search algorithm is identified as

$$t = \frac{\pi}{2\epsilon} = \frac{\pi\sqrt{N}}{4b} + \mathcal{O}(1). \tag{4.4.11}$$

Note that this equation finally verifies the results for $\frac{1}{b^2}$ in the case $d \geq 4$ as stated in (4.3.42).

Overall, the number of time steps T specified by the analysis is the integer closest to t where

$$t = \begin{cases} \frac{\pi\sqrt{N}}{4}\sqrt{\frac{2}{\pi}\ln N + \frac{16-8K}{\pi^2} + \frac{4\ln\frac{4}{\pi} - 2\ln 2}{\pi}} + \mathcal{O}(1) & \text{for } d = 2 \\ \frac{\pi\sqrt{N}}{4}\sqrt{\frac{6I_3}{\pi^3}} + \mathcal{O}(1) & \text{for } d = 3 \\ \Theta\left(\sqrt{N}\right) & \text{for } d \geq 4. \end{cases} \tag{4.4.12}$$

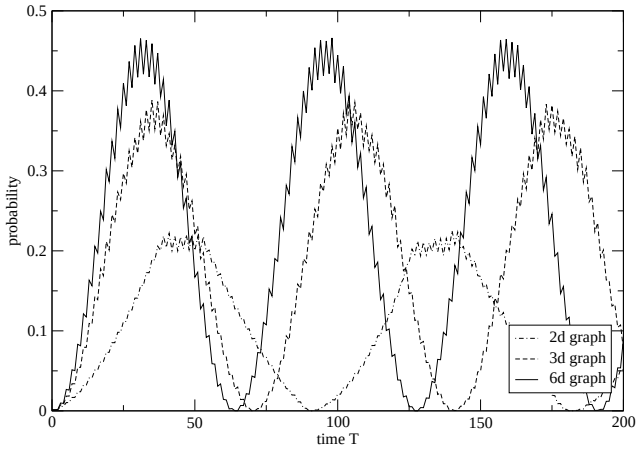

Figure 4.5: The time evolution of the probability on the marked vertex for three graphs containing 729 vertices in $d = 2, 3$ and 6 organised as 27^2, 9^3 and 3^6.

Figure 4.4 compares the analytical results for the localisation time to times obtained from numerical simulations. The theoretical results agree very well with the numerical outcome. The general behaviour suggested by this figure is that the walk for a fixed number of vertices is faster the higher the dimension. In fact, it is advisable to add some vertices and to reorganise the underlying database such that the search can take place in a higher dimension.

For example the figure shows clearly that the search on a 10^3 graph is faster than the search on a 30^2 graph although 100 vertices have been added. In addition to that equation (4.3.43) suggests that not only the search time but also the localisation is better for the larger system. Figure 4.5 gives a different example as it shows the probability at the target vertex as a function of T for three different graphs containing 729 vertices each, organised on lattices in 2, 3 or 6 dimensions. The search time as well as the localisation on the target vertex improve as d gets higher. A very naive reasoning to explain this effect is to realise that the graph is far better connected in larger dimensions since each vertex has $2d$ outgoing bonds. The price to pay for this speed up is an enlarged Hilbert

space since $\dim \mathcal{H} = 2dN$, where $N = n^d$.

The results of the analytical calculation as given in equations (4.4.12) and (4.3.43) shows a strikingly different behaviour for $d = 2$ and $d \geq 3$ which is confirmed by the simulations shown in figure 4.5.

In the following chapter 5, some applications of the search algorithm on the lattice are discussed.

CHAPTER 5

Applications of the lattice search and Grover's algorithm

The search algorithm on the lattice introduced in chapter 4 possesses some interesting features which will be discussed in this chapter. Some surprising effects occur when more than one marked vertex is located on the lattice. To start with, the search algorithm is generalised to search for m targets, where $m \ll N$. Then it will be shown that the search algorithm can be employed to send a signal from one vertex that therefore acts as 'sender' to another vertex or a set of vertices acting as 'receivers'. To the best of my knowledge such sender and receiver models have not been discussed before; apart from [2], where the results presented here have been published. The algorithm starts in a state localised at the sender. If sender and receivers are marked as for the search algorithm, the signal will localise at the receivers as the algorithm proceeds as shown in figure 5.4. This sender-receiver setting allows one to send a signal from one vertex of the graph to a set of receivers without localisation at other vertices. It should also be noted, that neither the sender nor the receivers need to know the others positions.

This effect can now be used to introduce a new kind of search algorithm that introduces

an additional marked vertex to the set of target vertices which acts as sender where the target vertices act as receivers. Doing so, a continuous signal can be send from the sender which will localise at the target vertices, see figure 5.6. This way a measurement of the signal will reveal the target vertices. Other than for the search algorithms of the previous chapters, it is not crucial to know the search time (for which knowledge of the number of receivers is needed) as the localisation on the target vertices does not disappear.

Numerical simulations are shown to visualise the effects and to verify the analytical results. Most figures for the lattices search shown in this chapter result from simulations for the 31×31 lattice although the analysis holds for arbitrary $d > 1$.

After the detailed discussion for the algorithms on the lattice, both effects, the sender and receiver model as well as the continuous signal search algorithm, will then be shown to hold also for the setting of Grover's search algorithm. To do so, a more general introduction of Grover's search algorithm as presented in section 2.3.1 will be given such that the algorithm searches for m target states. Then one of the marked states is singled out as sender. As for the lattice, the signal localises at the receiving states for the single impulse as well as the continuous signal which is used for the new search algorithm.

To close this chapter, some similarities of the lattice underlying the search algorithm discussed chapter 4 to periodic structures as in solid states or optical crystals are discussed. Furthermore, the band structure of the unperturbed 2-dimensional lattice is calculated.

5.1 m marked vertices

The analysis of the lattice search can easily be adapted to search for more than one marked vertex. The reduced space $\mathcal{H}'$ introduced in section 4.2.1 is defined such that

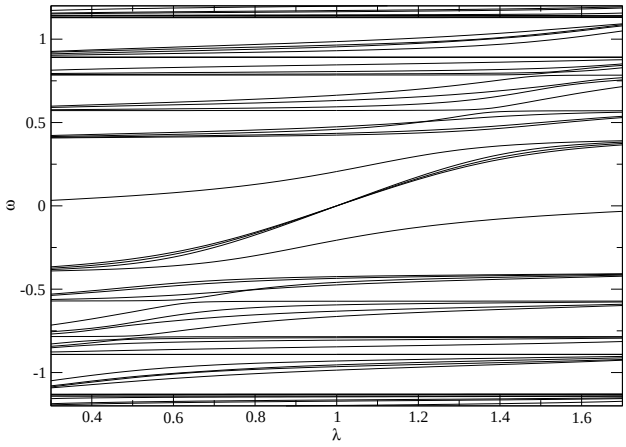

Figure 5.1: The crossing for $m = 4$ on a $d = 2$, $N = 121$ lattice, magnification.

$\mathcal{H} \setminus \mathcal{H}'$ is orthogonal to all vectors of type $|s\rangle \otimes |w\rangle$, where $|s\rangle$ is equally distributed in coin space and $|w\rangle$ an arbitrary vector in position space.

Consider $m \ll N$ marked vertices and denote these $\left|v^i\right\rangle$, where $i = 1, \ldots, m$ and $\vec{v^i} \neq \vec{v^j}$ for $i \neq j$. The operator U_λ can be adapted by changing the coin matrix analogous to the case with one marked vertex, that is

$$U_\lambda = U + \sum_{i=1}^{m} \left(\mathrm{e}^{\mathrm{i}\pi\lambda_i} - 1\right) U \left|sv^i\right\rangle \left\langle sv^i\right|, \tag{5.1.1}$$

where the index λ consists of m parameters $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_m)$. To simplify the notation $\lambda = 1$ will be used for $\lambda_i = 1$ for all i. The marking states $\left|sv^i\right\rangle$ can be calculated like in (4.2.6) and

$$\left|sv^i\right\rangle = \frac{1}{\sqrt{N}} |\phi_0\rangle + \frac{1}{\sqrt{2N}} \sum_{\vec{k} \neq \vec{0}} \omega^{-\vec{k}\vec{v^i}} \left(\left|\phi_{\vec{k}}^+\right\rangle + \left|\phi_{\vec{k}}^-\right\rangle\right). \tag{5.1.2}$$

The operator U_λ is now defined in an m-parameter space. Figure 5.1 shows the eigenphases at the avoided crossing along the line in parameter space where all λ_i's are equal. For the 2-dimensional lattice containing 117 unmarked and 4 marked vertices, there are

three eigenphases crossing at the point $(\lambda = 1, \omega = 0)$ and two outer eigenphases in an avoided crossing.

Again, a model Hamiltonian for the crossing is constructed. It can be reckoned that the important vectors are m localised states, one for each marked vertex, and the uniform distribution. This results in a model Hamiltonian specified by an $(m+1) \times (m+1)$ matrix. The spectra shown in figures 5.1 and 4.2 support this approach as there are $m+1$ eigenphases near $(\lambda = 1, \omega = 0)$.

Furthermore, the overlap between different localised states will be neglected, that is, $\langle \nu_\lambda^i | \nu_\lambda^j \rangle = \delta_{ij}$ is assumed. A second assumption that will be employed is that the localised state for a single marked vertex as given in (4.2.18) provides a good approximation for the localised states. Therefore the m vectors

$$\left| \nu_\lambda^i \right\rangle = \frac{b\left(\mathrm{e}^{\mathrm{i}\pi\lambda} - 1\right)}{\sqrt{2N}} \sum_{\vec{k} \neq \vec{0}} \omega^{-\vec{k}\vec{v^i}} \left(\frac{\mathrm{e}^{\mathrm{i}\theta_{\vec{k}}}}{\mathrm{e}^{\mathrm{i}g(\lambda)} - e^{\mathrm{i}\theta_{\vec{k}}}} \left| \phi_{\vec{k}}^+ \right\rangle + \frac{\mathrm{e}^{-\mathrm{i}\theta_{\vec{k}}}}{\mathrm{e}^{\mathrm{i}g(\lambda)} - e^{-\mathrm{i}\theta_{\vec{k}}}} \left| \phi_{\vec{k}}^- \right\rangle \right) \tag{5.1.3}$$

will be used for the m localised states.

Note that especially the assumption $\langle \nu_\lambda^i | \nu_\lambda^j \rangle = \delta_{ij}$ does not necessarily hold. It can be seen from the definition (5.1.3) and from figure 4.1 that the amplitude of $\left| \nu_\lambda^i \right\rangle$ is not zero everywhere but on $|v_i\rangle$. However, it will be assumed that interaction can be neglected as a relatively large probability is localised at the target vertex and its $2d$ nearest neighbours, see section 4.3.5.

Finally, it will be assumed that the matrix elements $\left\langle \nu_\lambda^i \mid H \mid \nu_\lambda^j \right\rangle = 0$ for $i \neq j$, that is, interaction takes place through the uniform distribution $|\phi_0\rangle$ only which can be regarded as carrier state. The remaining non-diagonal matrix elements $\pm \mathrm{i}\epsilon$ have been calculated in (4.4.9). Hence $\epsilon = \frac{2b}{\sqrt{N}} + \mathcal{O}\left(\frac{1}{N}\right)$. The diagonal entries ω are again integer multiples of 2π as they are defined by the phase of the crossing. Now, the Hamiltonian H can be determined using the canonical basis defined as $\{|1\rangle, |2\rangle, \ldots, |m+1\rangle\} =$

$\left\{|\Phi_0\rangle, \left|\nu^1_{\lambda=1}\right\rangle, \ldots, \left|\nu^m_{\lambda=1}\right\rangle\right\}$ that is

$$H = \begin{pmatrix} -\omega & -\mathrm{i}\epsilon & -\mathrm{i}\epsilon & \ldots & -\mathrm{i}\epsilon \\ \mathrm{i}\epsilon & -\omega & 0 & \ldots & 0 \\ \mathrm{i}\epsilon & 0 & -\omega & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ \mathrm{i}\epsilon & 0 & \ldots & 0 & -\omega \end{pmatrix} \tag{5.1.4}$$

in analogy to the definition in (4.4.1).

5.1.1 Eigenvalues and eigenvectors of the model Hamiltonian

To agree with numerical calculations of the spectrum of U_λ in the space $\mathcal{H}'$ as shown in figure 5.1 H needs $m-1$ eigenvalues $-\omega$ and 2 eigenvalues $-\omega \pm \frac{\Delta}{2}$. Here Δ is defined as the distance between the two outer eigenphases of U_λ.

Eigenvectors and eigenvalues of H are easily calculated:

$$\begin{array}{cc} \text{eigenvector} & \text{eigenvalue} \\ |u^\pm\rangle = \frac{\mathrm{i}}{\sqrt{2m}}\left(\mp\mathrm{i}\sqrt{m}, 1, 1, \ldots, 1\right)^\mathsf{T} & -\omega \pm \sqrt{m}\epsilon, \\ \left|u^i\right\rangle = \frac{1}{\sqrt{i(i-1)}}\left(\sum_{j=2}^{i}|j\rangle + (1-i)\,|i+1\rangle\right) & -\omega \end{array} \tag{5.1.5}$$

where $\left|u^i\right\rangle$ is defined for $i = 2, 3, \ldots, m$. Hence the analytical results coincide with the numerics.

5.2 Search for m target vertices

The uniform distribution $|\phi_0\rangle = |1\rangle$ can be expressed using the eigenvectors of H, that is, $|\phi_0\rangle = \frac{1}{\sqrt{2}}\left(|u^+\rangle - |u^-\rangle\right)$. Following the calculation in section 3.4.1, the time of the search is computed by applying $U_{\lambda=1}$ t-times to the starting state, where t remains

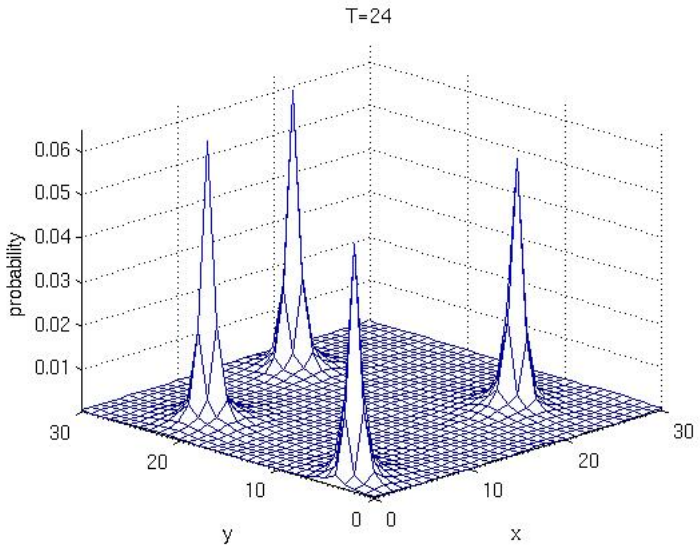

Figure 5.2: The localisation at $T = 24$ of the search for 4 marked vertices on a 31×31-lattice.

unknown. The resulting state is

$$\left(U_{\lambda=1}^{(m+1)\times(m+1)}\right)^t |\phi_0\rangle = \mathrm{e}^{-\mathrm{i}Ht}\frac{1}{\sqrt{2}}\left(|u^+\rangle - |u^-\rangle\right) \tag{5.2.1}$$

$$= \frac{1}{\sqrt{2}}\left(\mathrm{e}^{\mathrm{i}\left(\omega-\sqrt{m\epsilon}\right)t}|u^+\rangle - \mathrm{e}^{\mathrm{i}\left(\omega+\sqrt{m\epsilon}\right)t}|u^-\rangle\right). \tag{5.2.2}$$

The yet unknown time t is now defined such that $\mathrm{e}^{\mathrm{i}\sqrt{m\epsilon}t} = \mathrm{i}$, that is,

$$t = \frac{\pi}{2\sqrt{m\epsilon}} \tag{5.2.3}$$

and therefore

$$\left(U_{\lambda=1}^{(m+1)\times(m+1)}\right)^t |\phi_0\rangle = \frac{\mathrm{e}^{\mathrm{i}\omega t}}{\sqrt{2}}\left(-\mathrm{i}|u^+\rangle - \mathrm{i}|u^-\rangle\right) \tag{5.2.4}$$

$$= -\frac{\mathrm{i}\mathrm{e}^{\mathrm{i}\omega t}}{\sqrt{m}}(0,1,1,\ldots,1)^{\mathsf{T}}. \tag{5.2.5}$$

It can now be seen that the search algorithm localises on all m marked vertices simultaneously after T time steps, where T is the integer closest to t. This coincides with the result for $m = 1$ in (4.4.11).

In figure 5.2 the resulting state of the search algorithm for 4 target vertices is displayed while figure 5.3 shows the time evolution of the probability on several vertices. For a

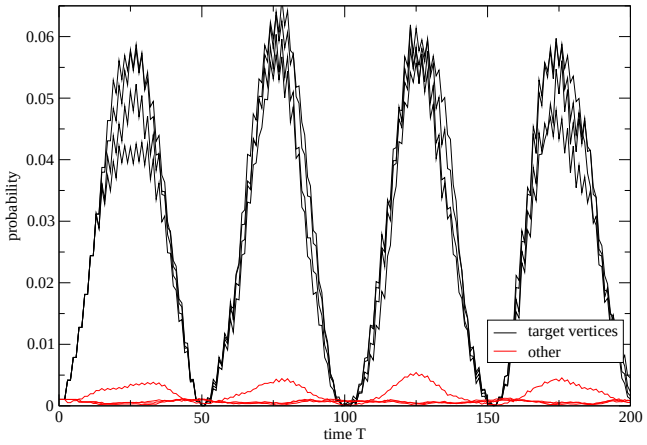

Figure 5.3: Time evolution on the 4 marked vertices and 4 randomly chosen unmarked vertices on a 31×31-lattice.

31×31-lattice, the search localises on the marked positions after $T = 22$ times steps. To visualise the time evolution, figure 5.3 shows how the probabilities on the 4 target vertices evolve compared to the probabilities on 4 randomly chosen unmarked vertices in the same setting, that is, the same configuration of target vertices and the same lattice.

Equation (4.3.43) shows that, for the 2-dimensional lattice, the probability decays with increasing N. This explains the small probability to measure the walk at one of the target vertices in figures 5.2 and 5.3, but the probability is still large compared to the probability at the unmarked vertices. However, better results can be expected for the lattice in $d \geq 3$ dimensions (compare figure 4.3).

5.3 Transfer of a signal

The quantum search with $m \geq 2$ marked vertices can be used to transmit a signal across the graph without further changes to the search operator $U_{\lambda=1}$. It is also remarkably that neither sender or receiver need to now each others position.

5.3.1 Single impulse

If the algorithm starts not in the uniform distribution but at vertex $|v^m\rangle$ the starting state is $\left|\nu^m_{\lambda=1}\right\rangle \mathrel{\widehat{=}} |m+1\rangle$. After applying $U_{\lambda=1}$ t times, the resulting state is of the form

$$\left(U^{(m+1)\times(m+1)}_{\lambda=1}\right)^t |\nu^m_\lambda\rangle \tag{5.3.1}$$

$$= -\frac{\mathrm{i}\mathrm{e}^{-\mathrm{i}Ht}}{\sqrt{2m}}\left(\left|u^+\right\rangle + \left|u^-\right\rangle - \mathrm{i}\sqrt{2\,(m-1)}\,|u^m\rangle\right) \tag{5.3.2}$$

$$= -\frac{\mathrm{i}\mathrm{e}^{\mathrm{i}\omega t}}{\sqrt{2m}}\left(\mathrm{e}^{-\mathrm{i}\sqrt{m}\epsilon t}\left|u^+\right\rangle + \mathrm{e}^{\mathrm{i}\sqrt{m}\epsilon t}\left|u^-\right\rangle - \mathrm{i}\sqrt{2\,(m-1)}\,|u^m\rangle\right). \tag{5.3.3}$$

Choosing $t = \frac{\pi}{\sqrt{m}\epsilon}$

$$\left(U^{(m+1)\times(m+1)}_{\lambda=1}\right)^t |\nu^m_{\lambda=1}\rangle = \frac{\mathrm{i}\mathrm{e}^{\mathrm{i}\omega t}}{\sqrt{2m}}\left(\left|u^+\right\rangle + \left|u^-\right\rangle + \mathrm{i}\sqrt{2\,(m-1)}\,|u^m\rangle\right) \tag{5.3.4}$$

$$= -\mathrm{e}^{\mathrm{i}\omega t}\left(0, \frac{2}{m}, \frac{2}{m}, \ldots, \frac{2}{m}, \frac{2}{m}-1\right)^{\mathsf{T}} \tag{5.3.5}$$

is obtained and the state is localised on all $m-1$ remaining marked vertices with the same amplitude and a remaining contribution on the mth marked vertex. For further reference, the vertex $|v^m\rangle$ will be referred to as 'sender' and the remaining $m-1$ marked vertices as 'receivers'.

Note that in the special case $m = 2$, the whole signal is transmitted and the amplitude on the sender vanishes completely. Figure 5.4 shows how the intensity on the sender, a single receiver and 4 randomly chosen vertices evolves in time. The noise on top of the signal of the sender seems to be due to the way the starting state has been generated. In an experiment it would be difficult to generate $\left|\nu^{\text{sender}}_\lambda\right\rangle$. Therefore the starting state used for the numerics was constructed as the localised state of the search algorithm that started in the uniform distribution where $\left|\nu^{\text{sender}}_\lambda\right\rangle$ acted as the only marked vertex. Only after this first localisation, when the probability at the sender had reached its first maximum, the second marked vertex was introduced.

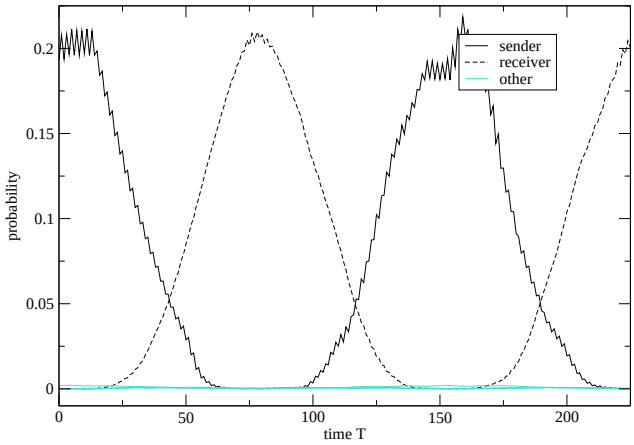

Figure 5.4: A signal is transferred from one vertex, acting as sender to another one, the receiver, on a 31×31-lattice. At the bottom of the figure, the probability evolution on 4 randomly chosen vertices is plotted.

The whole process is periodic and once the walk is localised at the receiver the rôles of sender and receiver change and the receiver returns the signal back to the sender.

The set of marked vertices splits in one sender and $m-1$ receivers. At each receiver $\frac{4}{m^2}$ of the probability will localise and a probability of $\frac{(2-m)^2}{m^2}$ remains at the sender. Figure 5.5 shows some results for various simulations on a 31×31 lattice. The general behaviour agrees with the theoretical result although there are differences. The theoretical model is based on the assumption that there are no interactions between different marked vertices, that is, the configuration of sender and receivers does not have any influence on the result. This assumption is just an approximation and figure 5.5 shows that the configuration of sender and receivers does play a rôle. This can be seen as there are different intensities for different configurations. In the case of two receivers the differences obtained for different configurations are most striking.

Furthermore the plot showing the localisation of the search for 4 marked vertices printed in figure 5.2 shows that the distribution is likely to affect the resulting state: It can be

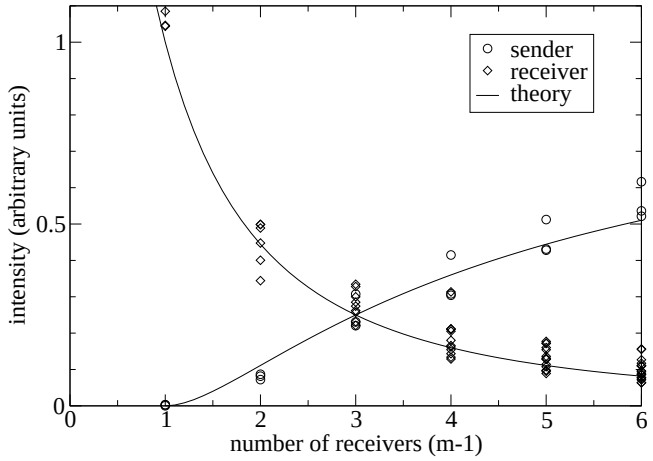

Figure 5.5: Amplitude of the signal at the sender and the receivers after the localisation on the receivers on a 31×31 lattice. Three different receiver configurations have been considered for each fixed m. The bold lines give the predictions obtained from equation (5.3.5).

seen that the amplitude on the neighbours of the marked vertices is notably away from zero and if two receivers are close together, e.g. neighbours, interference effects are expected. Thus the different results for different configurations of marked vertices shown in figure 5.5 are explained.

Interestingly, the sender can estimate how many receivers are listening to the signal by tracking the signal locally at the senders position. There are two indicators that enable an estimate, the time and the minimal amplitude. The time after which the signal at the sender reaches its minimum scales like $t = \frac{\pi}{\sqrt{m\epsilon}}$, but since the minimum are quite broad, the time measurements are inaccurate. Figure 5.5 suggests that by tracking the magnitude locally to measure the minimal amplitude, the sender is able to distinguish between $m - 1 = 1, 2$ or ≥ 3 receivers. This technique is not accurate for more than 3 receivers because the differences due to different sender-receiver configurations are on the same scale as the difference between different numbers of receivers. However, if the

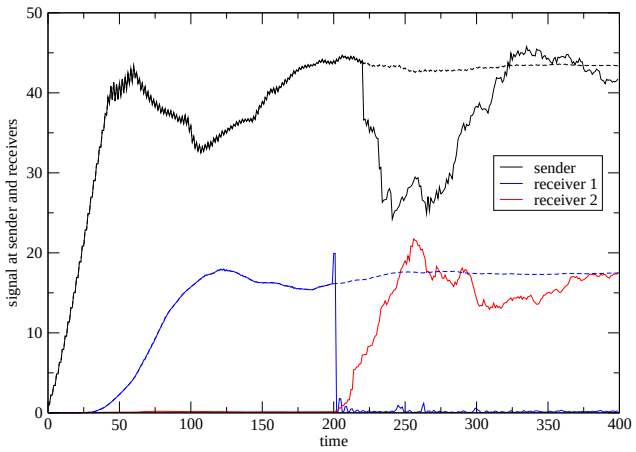

Figure 5.6: A signal is fed in continuously at a sender and observed at the receiver. At time $t = 200$ the receiver changes position and the dashed lines show the signals for the case of an unchanged receiver. A 31×31 lattice was used and to keep the signal finite, the system was damped and 10% of the signals at sender and receiver and 1% at all other vertices was reduced.

sender knows where the receivers are located on the lattice, an estimate seems possible.

5.3.2 Continuous signal

Since it is possible to transfer a signal from one marked vertex to some receivers without knowing the number or positions of the receivers, it is interesting to investigate if the mechanism works for a continuous signal as well. Numerical simulations suggest that this is the case.

Figure 5.6 shows the results for two different settings. In both simulations the signal is fed into the lattice by adding the state $|sv\rangle$ at a sender $\vec{v}$ at each time step. A single sender was introduced at a randomly chosen position and the signal at sender and receiver has been investigated. Furthermore, some damping was introduced to keep the signal finite. Up to time $T = 200$ the figure shows how the signal evolves at sender and

receiver. The system takes some time until an equilibrium between damping and added signal occurs. After $T = 200$, the system is almost stable as shown by the dashed line for $T > 200$.

The second simulation, shown by the continuous line, starts identically but at $T = 200$ the position of the receiver was changed and the figure for $T \geq 200$ shows how the system adapts to this sudden change. The result is that the signal at the old receiver position breaks down almost immediately whereas a new peak arises at the new position in less time than the peak at the initial position.

The faster time is easy to explain, since according to the model introduced in section 5.1, the localised states at sender and receiver interact through a carrier state, the uniform distribution. Therefore the peak at the first receiver can not be built up, until a sufficient amount of the signal has been transferred into the carrier state. When the position of the receiver is changed, this is not necessary as the carrier state is already occupied.

The overshooting of the signal at the old position shortly after the receiver has been moved is due to the special choice of damping. That is, a general damping of 1% at all vertices and a damping of 10% at sender and receiver. This has been done to simulate an additional bond attached to those vertices where 10% of the signal leaves the graph, enabling sender and receiver to perform measurements without affecting the state in the system. Hence the equilibrium that arises after a critical time at the receiver takes this damping into account. Then, when the position is changed the damping at the old position disappeared and the signal overshoots before the system adjusts to the new setting with the now unmarked coin at this position and the signal breaks down.

5.4 Grover's search algorithm

In this section, the sender and receiver model and the continuous search discussed above will be adapted to Grover's search algorithm. In section 2.3.1 Grover's search algorithm has been introduced (see also [5, 23]) but a detailed discussion has been postponed until now. The discussion starts with a repetition of the definition given in the earlier section, here adapted to m target states.

Given a system with N states of which $m \ll N$ states are target states. Grover constructed a search algorithm [23] that will localise on the target states using the following steps.

1. initialise the system in the uniform distribution

$$|\psi\rangle := \left(\frac{1}{\sqrt{N}}, \frac{1}{\sqrt{N}}, \frac{1}{\sqrt{N}}, \ldots, \frac{1}{\sqrt{N}}\right)^{\mathsf{T}} \tag{5.4.1}$$

2. repeat the following unitary operations T times, where T is a particular, but yet unknown time $T = \Omega\left(\sqrt{N}\right)$

   (a) rotate the phase of all states $|v_i\rangle$ by π if $|v_i\rangle$ is a target state and by 0 otherwise.

   (b) apply the unitary matrix D defined by $D_{ii} = -1 + \frac{2}{N}$ and $D_{ij} = \frac{2}{N}$ for $i \neq j$

3. measure the resulting state $|\psi_{\text{result}}\rangle$.

5.4.1 Grover's search algorithm for m marked vertices

The calculation of the search time for m marked vertices follows the presentation in [5]. As discussed in section 2.3.1, Grover's algorithm, consists of a successive application of two reflection operators performing a rotation in a two dimensional subspace.

$$|\psi\rangle = \frac{1}{\sqrt{N}}\,(1, 1, \ldots, 1)^{\mathsf{T}} \tag{5.4.2}$$

is defined as the uniform distribution (5.4.1) and let the uniform distributions of the m target states and the $N-m$ unmarked states be defined as

$$|v_t\rangle = \frac{1}{\sqrt{m}}\sum_{\vec{x}}{}' |x\rangle \tag{5.4.3}$$

$$|v_u\rangle = \frac{1}{\sqrt{N-m}}\sum_{\vec{x}}{}'' |x\rangle\,, \tag{5.4.4}$$

where $\sum'$ sums over all marked and $\sum''$ over all unmarked states $\vec{x}$. Using these distributions, the uniform distribution (5.4.1) can be expressed as $|\psi\rangle = \sqrt{\frac{m}{N}}\,|v_t\rangle + \sqrt{\frac{N-m}{N}}\,|v_u\rangle = \sin\frac{\theta}{2}\,|v_t\rangle + \cos\frac{\theta}{2}\,|v_u\rangle$ for some angle θ with

$$\theta = 2\arcsin\sqrt{\frac{m}{N}}. \tag{5.4.5}$$

Grover's algorithm can be denoted as

$$G = (2\,|\psi\rangle\,\langle\psi| - \mathbb{1})\,O, \tag{5.4.6}$$

where the oracle O switches the sign of all components of the m marked states [5] and therefore

$$O = \mathbb{1} - 2\sum_{\vec{x}}{}' |x\rangle\,\langle x|\,. \tag{5.4.7}$$

For any superposition $|\alpha\rangle = \sin\alpha\,|v_t\rangle + \cos\alpha\,|v_u\rangle$, Grover's algorithm performs a rotation in the 2-dimensional space spanned by $|v_t\rangle$ and $|v_u\rangle$ since

$$G\,|\alpha\rangle = (2\,|\psi\rangle\,\langle\psi| - \mathbb{1})\,(-\sin\alpha\,|v_t\rangle + \cos\alpha\,|v_u\rangle) \tag{5.4.8}$$

$$= 2\,|\psi\rangle\left(-\sin\frac{\theta}{2}\sin\alpha + \cos\frac{\theta}{2}\cos\alpha\right) + \sin\alpha\,|v_t\rangle - \cos\alpha\,|v_u\rangle \tag{5.4.9}$$

$$= \left(\left(-2\sin^2\frac{\theta}{2}+1\right)\sin\alpha + 2\cos\frac{\theta}{2}\sin\frac{\theta}{2}\cos\alpha\right)|v_t\rangle + \left(\left(2\cos^2\frac{\theta}{2}-1\right)\cos\alpha - 2\sin\frac{\theta}{2}\cos\frac{\theta}{2}\sin\alpha\right)|v_u\rangle \tag{5.4.10}$$

$$= \sin(\theta+\alpha)\,|v_t\rangle + \cos(\theta+\alpha)\,|v_u\rangle \tag{5.4.11}$$

and the angle of the rotation is θ. Note that G does not shift the state out of the plane spanned by $|v_t\rangle$ and $|v_u\rangle$. A sketch of the 2-dimensional subspace and the effect of G is shown in figure 5.7.

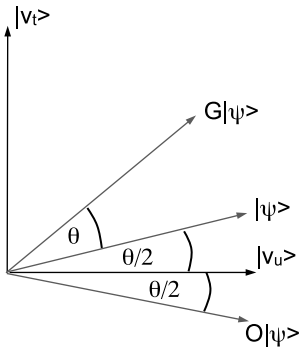

Figure 5.7: The action of G in the 2-dimensional subspace.

In particular, this is true for $|\psi\rangle$ and t applications of G lead to

$$G^t |\psi\rangle = \sin\left(\frac{\theta}{2} + t\theta\right) |v_t\rangle + \cos\left(\frac{\theta}{2} + t\theta\right) |v_u\rangle . \tag{5.4.12}$$

It is obvious that for small angles θ, the algorithm reaches a state close to $|v_t\rangle$ for some t and will be measured in one of the m marked states after t applications of G with high probability. The optimal number of applications of the search algorithm G is defined as the integer T which is closest to t defined by

$$\sin\left(\frac{\theta}{2} + t\theta\right) = 1, \tag{5.4.13}$$

that is,

$$t = \frac{\pi}{2\theta} - \frac{1}{2}. \tag{5.4.14}$$

If only a small fraction of states are target states and $m \ll N$, then $\theta \approx 2\sqrt{\frac{m}{N}}$ follows from (5.4.5) and

$$t \approx \frac{\pi\sqrt{N}}{4\sqrt{m}} - \frac{1}{2}. \tag{5.4.15}$$

Thus the number of time steps T scales like

$$T = \Theta\left(\sqrt{\frac{N}{m}}\right). \tag{5.4.16}$$

5.4.2 Sender and receiver model for Grover's algorithm

This thesis introduced a sender-receiver model for the lattice search. As will be shown in this section, the same can be done for Grover's search algorithm.

To transfer the sender and receiver model to Grover's algorithm the definitions of the previous section need to be changed. Additional to the m marked vertices, a special marked state $|v_0\rangle$ called 'sender' is introduced making the number of unmarked vertices $N - m - 1$.

The new set of vectors is defined as

$$|v_t\rangle = \frac{1}{\sqrt{m}} {\sum_{\vec{x} \neq \vec{v}_0}}' |x\rangle \tag{5.4.17}$$

$$|v_u\rangle = \frac{1}{\sqrt{N-m-1}} {\sum_{\vec{x}}}'' |x\rangle \tag{5.4.18}$$

$$|\psi\rangle = \sqrt{\frac{m}{N}} |v_t\rangle + \sqrt{\frac{N-m-1}{N}} |v_u\rangle + \frac{1}{\sqrt{N}} |v_0\rangle , \tag{5.4.19}$$

where $\sum'$ and $\sum''$ sum over all marked and unmarked states respectively. Note that $|v_t\rangle$ is the uniform distribution of all marked states but the sender.

To simplify the analysis, two auxiliary states are introduced

$$|\alpha\rangle = \sqrt{\frac{m}{m+1}} |v_t\rangle + \frac{1}{\sqrt{m+1}} |v_0\rangle \tag{5.4.20}$$

$$|\beta\rangle = -\frac{1}{\sqrt{m+1}} |v_t\rangle + \sqrt{\frac{m}{m+1}} |v_0\rangle \tag{5.4.21}$$

such that $|\alpha\rangle$ is the uniform distribution of all marked states including the sender and $|\beta\rangle$ is a second state in the $|v_t\rangle$-$|v_0\rangle$-plane orthogonal to $|\alpha\rangle$. Note that this implies $|\beta\rangle$ orthogonal to $|\psi\rangle$, the uniform distribution of all states.

The uniform distribution lies in the span of $|u_u\rangle$ and $|\alpha\rangle$ and an angle θ is defined such

that

$$|\psi\rangle = \sqrt{\frac{N-m-1}{N}}\,|v_u\rangle + \sqrt{\frac{m+1}{N}}\,|\alpha\rangle \tag{5.4.22}$$

$$= \cos\frac{\theta}{2}\,|v_u\rangle + \sin\frac{\theta}{2}\,|\alpha\rangle\,. \tag{5.4.23}$$

The action of G on an arbitrary vector in the span of the vectors $|v_u\rangle$, $|\alpha\rangle$ and $|\beta\rangle$ will be analysed using a normalised test vector $|a\rangle = a_u\,|v_u\rangle + a_\alpha\,|\alpha\rangle + a_\beta\,|\beta\rangle$ with $|a_u|^2 + |a_\alpha|^2 + |a_\beta|^2 = 1$:

$$G\,|a\rangle = (2\,|\psi\rangle\,\langle\psi| - \mathbb{1})\left(a_u\,|v_u\rangle - a_\alpha\,|\alpha\rangle - a_\beta\,|\beta\rangle\right) \tag{5.4.24}$$

$$=2\,|\psi\rangle\left(a_u\cos\frac{\theta}{2} - a_\alpha\sin\frac{\theta}{2}\right) - a_u\,|v_u\rangle + a_\alpha\,|\alpha\rangle + a_\beta\,|\beta\rangle \tag{5.4.25}$$

$$= (a_u\cos\theta - a_\alpha\sin\theta)\,|v_u\rangle + (a_u\sin\theta + a_\alpha\cos\theta)\,|\alpha\rangle + a_\beta\,|\beta\rangle\,. \tag{5.4.26}$$

As before, this subspace does not mix with its complement and the algorithm once started in this subspace is trapped there.

If the canonical basis is identified with these three basis vectors, that is $|v_u\rangle = |1\rangle$, $|\alpha\rangle = |2\rangle$ and $|\beta\rangle = |3\rangle$, G is the rotation matrix

$$G = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{5.4.27}$$

as can be seen from (5.4.26), and hence

$$G^t = \begin{pmatrix} \cos(t\theta) & -\sin(t\theta) & 0 \\ \sin(t\theta) & \cos(t\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{5.4.28}$$

The uniform distribution of all unmarked states is $|v_u\rangle = (1,0,0)^{\mathsf{T}}$. An angle ϕ is

defined such that $\sin\frac{\phi}{2} = \frac{1}{\sqrt{1+m}}$ and $\cos\frac{\phi}{2} = \sqrt{\frac{m}{1+m}}$ and simple algebra leads to

$$|v_0\rangle = \left(0, \sin\frac{\phi}{2}, \cos\frac{\phi}{2}\right)^{\mathsf{T}} \tag{5.4.29}$$

$$|v_t\rangle = \left(0, \cos\frac{\phi}{2}, -\sin\frac{\phi}{2}\right)^{\mathsf{T}}. \tag{5.4.30}$$

Now, the effect of t applications of G on the sender $|v_0\rangle$ can easily be analysed and one observes oscillations in all three states

$$\langle v_0 \mid G^t \mid v_0\rangle = \sin^2\frac{\phi}{2}\cos(t\theta) + \cos^2\frac{\phi}{2} \tag{5.4.31}$$

$$\langle v_t \mid G^t \mid v_0\rangle = \frac{1}{2}\sin\phi\,(\cos(t\theta) - 1) \tag{5.4.32}$$

$$\langle v_u \mid G^t \mid v_0\rangle = -\sin\frac{\phi}{2}\sin(t\theta)\,. \tag{5.4.33}$$

Figure 5.8 shows the probability for $N = 50$, and $m = 1$ and $m = 5$ in two different plots in one state of the set of receiving and unmarked states. Note that the curve for the unmarked state is close to zero and barely visible.

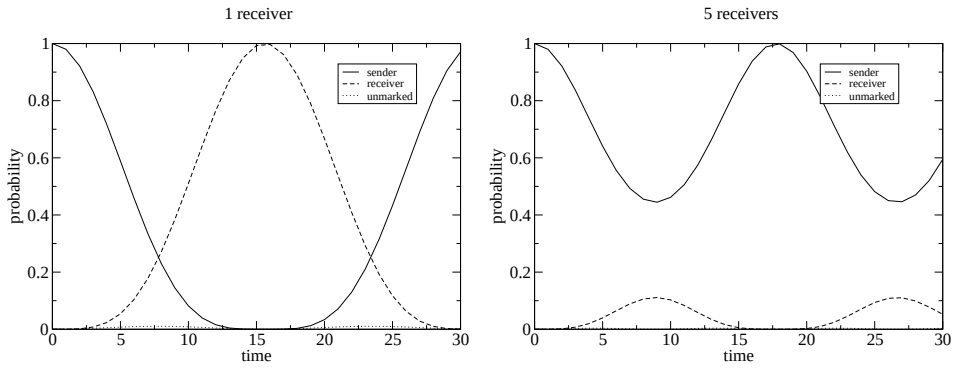

Figure 5.8: Results for Grover's algorithm in a setting with 50 states and one sender and one and five receivers respectively.

5.4.3 Continuous sender in Grover's algorithm

To send an continuous signal from the sender to the receivers, some damping needs to be introduced such that the system reaches a constant saturation state. Let p be the

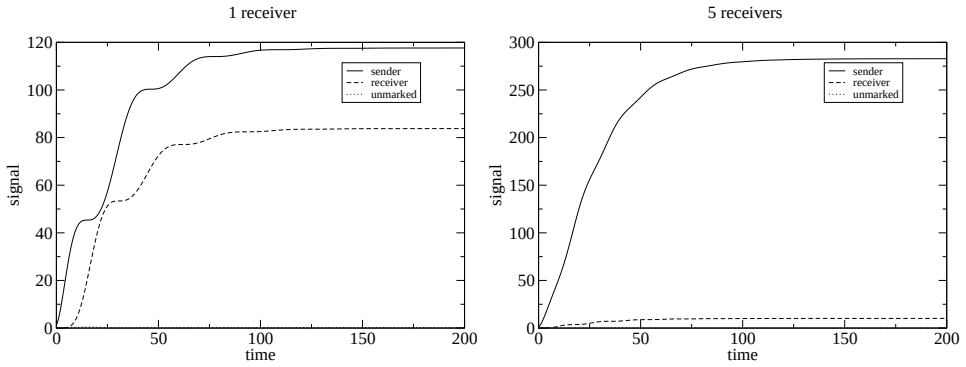

Figure 5.9: Results for Grover's algorithm with 50 states and continuous sender. The figures show the signal of a system with 5% damping and one receiver and five receivers respectively.

probability of the undamped part, then $(100 - 100p)$ % of the amplitude in each state is damped. The state $|A(T)\rangle$ after T time steps can be calculated using

$$|A(T)\rangle = \sum_{i=0}^{T-1} p^i G^i |a\rangle , \tag{5.4.34}$$

where $|a\rangle$ is an arbitrary state that is added every time step.

Using the amplitudes given in equations (5.4.31), (5.4.32) and (5.4.33) the signal $\frac{\langle test|A(t)\rangle^2}{k}$ has been plotted in figure 5.9, where k is the number of states, that is 1 for the sender, m for the receiver and $N - m - 1$ for the unmarked state and $|test\rangle$ was chosen equal to the uniform distributions of sender, receivers and unmarked states.

5.5 Continuous model as search algorithm

The continuous-sender model for Grover's search and the lattice search as shown in sections 5.3.2 and 5.4.3 represents an improvement to both search algorithms. Additional to the target vertices of the search setting, a sender is introduced and sends a continuous signal to the target vertices which act as receivers. This model has the advantages that

no uniform distribution is required to search for the target vertices and the number of target vertices does not need to be known.

Unlike the original algorithm from Grover, there is no need to know the search time exactly since once the walk is localised at the receivers, the signal remains there. A critical point of Grover's algorithm is knowing the localisation time T close to $t = \frac{\pi\sqrt{N}}{4\sqrt{m}} - \frac{1}{2}$ at which one needs to perform the measurement. Since T depends on the overall number of states N and the number of target states m, Grover's algorithm requires the knowledge of the size of the system N and the number of target states in advance. If the measurement is not performed at T, the algorithm is not well localised and the result of the measurement can be any arbitrary vertex.

The algorithm with continuous sender produces a high signal at all marked vertices, that is sender and target vertices. It reaches a stable localisation such that the time is not a critical factor for the algorithm and neither the number of target states. It might seem to be a disadvantage that the time at which the localisation reaches a stationary state takes significantly longer. In the example given in figure 5.9 for one receiver, no major changes occur after 100 time steps while the localisation time defined by (5.4.15) is 5 time steps. However, the numerics suggests that a localisation emerges with an amplitude notably higher than at the unmarked states after a much shorter time.

A different approach leading to a somewhat similar result was introduced in [60]. Mizel introduced a search algorithm similar to Grover's algorithm that succeeds to localise at an unknown number of target vertices in a time $\sqrt{N}$. Due to dissipation the rotation in the two dimensional subspace is damped and does eventually stop in the localised state. Although the method used is quite different from the approach discussed in this thesis, both algorithms are able to find an unknown number of target vertices and the localised state becomes stationary.

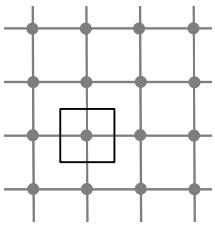

Figure 5.10: Unit cell (black) within the 2 dimensional periodic lattice (grey).

5.6 Conditions for an experimental realisation

A promising implementation for an experimental realisation is provided by systems with periodic structures like crystals in solid states or photonic crystals. An introduction to photonic crystals can be found in [61]. These systems can be regarded as periodic structures consisting of unit cells and it is possible to introduce perturbations by altering single unit cells.

The system of the search on a regular square lattice as defined in section 4.1.1 draws a similar picture since an otherwise periodic system is perturbed by a marked vertex or defect.

As sketched in figure 5.10 the lattice can be understood to be composed of unit cells whose boundaries cross the bonds of the graph in the middle. The operator U introduced in section 4.1.1 will, for the scope of this section, be redefined such that it now propagates waves from one unit cell to its nearest neighbours. Other than in solid states or photonic crystals, the lattice takes only interactions between nearest neighbours into account and neglects dissipation.

5.6.1 Band structure of the unperturbed lattice

Assuming that $U_{\lambda=0}$ does not depend on the frequency of the quantum particle propagating through the 2-dimensional lattice, the band structure of the unperturbed lattice

can be evaluated by the condition

$$\det\left(\mathbb{1}_{2dn^d} - \mathrm{e}^{\mathrm{i}ka} U_{\lambda=0}\right) \equiv 0, \tag{5.6.1}$$

where a is the geometrical width of the unit cells and k the wave length of the walker. Hence $\mathrm{e}^{\mathrm{i}ka}$ is the phase shift between incoming and outgoing waves of the unit cells.

Note that condition (5.6.1) is equivalent to demanding $\mathrm{e}^{-\mathrm{i}ka}$ being an eigenvalue of $U_{\lambda=0}$ as it is equivalent to $\det\left(\mathbb{1}_{2dn^d}\mathrm{e}^{-\mathrm{i}ka} - U_{\lambda=0}\right) \equiv 0$. The phase factor can be obtained by calculating the eigenstates, that is, asking for stationary solutions of $U_{\lambda=0}$.

Consider the 2-dimensional lattice and let ψ_b^x, ψ_d^y be the wave functions on bond b and d respectively, where b is an index varying over all bonds pointing in x direction and d enumerating the bonds in y direction. Let the coordinate on the bonds be defined such that the direction of increasing coordinates agrees with the directions of x and y.

Wave propagation on the bonds can be characterised using plane waves and a set of coefficients A_b, B_b, C_d, D_d according to

$$\psi_b^x = A_b \mathrm{e}^{\mathrm{i}kx} + B_b \mathrm{e}^{-\mathrm{i}kx} \tag{5.6.2}$$

$$\psi_d^y = C_d \mathrm{e}^{\mathrm{i}ky} + D_d \mathrm{e}^{-\mathrm{i}ky}. \tag{5.6.3}$$

Regard the wave functions in the neighbourhood of a vertex v. Let b and d be the bonds leaving v in negative x and y directions and $b+1$, $d+1$ the bonds leaving v in positive x and y directions respectively. Due to the periodicity of the system stationary solutions are translational invariant and the wave functions ψ_{b+1}^x and ψ_{d+1}^y are specified by the wave functions on ψ_b^x and ψ_d^y and the yet unknown phase factors κ_1 and κ_2

$$\begin{aligned} A_{b+1} &= \mathrm{e}^{\mathrm{i}\kappa_1 a} A_b \\ B_{b+1} &= \mathrm{e}^{\mathrm{i}\kappa_1 a} B_b \\ C_{b+1} &= \mathrm{e}^{\mathrm{i}\kappa_2 a} C_b \\ D_{b+1} &= \mathrm{e}^{\mathrm{i}\kappa_2 a} D_b \end{aligned} . \tag{5.6.4}$$

However, time propagation provides an additional mapping between the amplitudes that is governed by the vertex scattering matrix σ that maps the four incoming waves at v to the four outgoing waves

$$\begin{pmatrix} B_b e^{-ika} \\ A_{b+1} \\ D_d e^{-ika} \\ C_{d+1} \end{pmatrix} = \sigma \begin{pmatrix} A_b e^{ika} \\ B_{b+1} \\ C_d e^{ika} \\ D_{d+1} \end{pmatrix}. \tag{5.6.5}$$

The definition of σ from equation (4.1.1) and the set of identities in (5.6.4) leads to a linear system of equations in A_b, B_b, C_d and D_d

$$\begin{pmatrix} -e^{ika} & e^{i\kappa_1 a} - 2e^{-ika} & e^{ika} & e^{i\kappa_2 a} \\ e^{ika} - 2e^{i\kappa_1 a} & -e^{i\kappa_1 a} & e^{ika} & e^{i\kappa_2 a} \\ e^{ika} & e^{i\kappa_1 a} & -e^{ika} & e^{i\kappa_2 a} - 2e^{-ika} \\ e^{ika} & e^{i\kappa_1 a} & e^{ika} - 2e^{i\kappa_2 a} & -e^{i\kappa_2 a} \end{pmatrix} \begin{pmatrix} A_b \\ B_b \\ C_d \\ D_d \end{pmatrix} = \vec{0}. \tag{5.6.6}$$

To obtain stationary solutions for the wave function, the determinant of the matrix of coefficients needs to be zero, and hence

$$0 = \sin(ka)\left(\cos(\kappa_1 a) + \cos(\kappa_2 a) - 2\cos(ka)\right) \tag{5.6.7}$$

defines the band structure of the 2 dimensional lattice.

Figure 5.11 shows a three dimensional plot of the thus obtained band structure for $a = 1$. The band structure is $2k$ periodic and the figure is restricted to $k \in [-\pi, \pi]$.

The finite number of vertices per dimension and periodic boundary conditions of the lattice demand the phase factors for n translations in dimension i to be $e^{i\kappa_i a n} = 1$. Therefore the band structure is discretised according to

$$\kappa_i = \frac{2\pi l_i}{an} \qquad\qquad l_i \in \mathbb{Z}. \tag{5.6.8}$$

A phase shift of more than 2π in positive or negative direction can not be distinguished from a smaller one. Thus $\kappa_i a$ can be restricted to $[-\pi, \pi]$ without loss of generality.

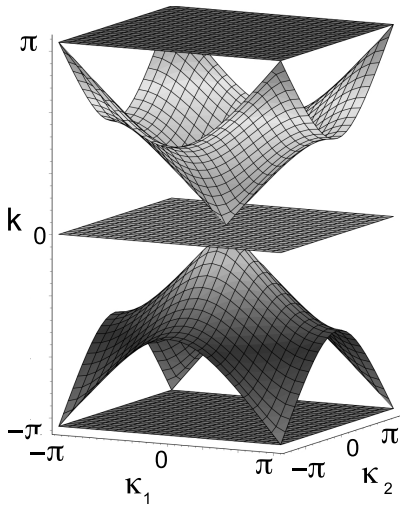

Figure 5.11: The band structure of the two dimensional lattice.

The thus obtained discrete band structure coincides with the eigenphases discussed in section 4.1.2.

5.6.2 Defect states

From the analysis in section 4.2 it is known that the perturbation introduces a new eigenphase, the defect state, that crosses through the spectrum at $k = 0$ modulo 2π.

Defect states that are localised in the band gap are trapped at the defect since no transition through the regular, unperturbed system is possible. Given a system with defect states that are within the band and have a frequency near a uniformly distributed state at $k = 0$ modulo 2π. According to the analysis of the search on the lattice, it should be possible for the defect state of this system to interact with the eigenstates of the unperturbed system and it may be assumed that a localisation from the uniformly distributed state does occur in an experimental setting.

Although it is not yet clear what conditions such a system has to meet, it will be interesting for further research to investigate if an experimental search algorithm in

a crystal can be constructed. Following the reasoning above, one can expect that one condition is that the system has a localised defect with a defect state close to a uniformly distributed state.

CHAPTER 6

Some thoughts on the lattice search

This chapter lists some speculations and ideas why some search algorithms are successful while others are not. There are many open questions about the conditions for successful localisation processes.

The main part of the chapter consists of reasoning and numerical simulations. Based on the lattices search different coin flip operations are tested against the search constructed by Ambainis, Kempe and Rivosh (AKR). Although the lattice search in d dimensions does not scale optimal, that is with $\sqrt{N}$, the coin flips tested in this chapter are not better. On the contrary, two of the three tested algorithms achieve no significant localisation at all, figure 6.3. Only the coin in equation (6.2.1) results in a time which scales better than the AKR-coin, compare figure 6.1. However, the localisation is not as good as for the AKR-coin (figure 6.2) and therefore the overall behaviour does not seem to be better. A detailed analysis of the search algorithm would be necessary to compare both search algorithms but this however, is beyond the scope of this chapter and due to lack of time beyond the scope of this thesis.

6.1 What makes an algorithm efficient?

It is somehow surprising that the search algorithm on the lattices has a different efficiency in 2 than in 3 or more dimensions. Answering the question 'Why?' might provide deeper inside into conditions that enable the algorithm to localise on the target vertex faster. The algorithm in 2 dimensions needs $\mathcal{O}\left(\sqrt{N \ln N}\right)$ time steps, while $\mathcal{O}\left(\sqrt{N}\right)$ provides a lower bound for search algorithms in general [23, 24]. Therefore it is not guaranteed that no algorithm exists that provides a search on the 2-dimensional lattice with a more efficient scaling in N. Furthermore equation (4.3.43) shows that the scaling of the amplitude on the marked vertex is not as good as for lattices with more than two dimensions.

How does the search algorithm for a d-dimensional lattice localise? It can be seen from figure 4.5 and from the theory in section 4.4 that, up to some fluctuations, the probability to measure the walk at the target vertex increases steadily until it reaches its maximum where it remains for a few time steps before it starts do decrease again. Since a notable amount of probability is localised at the marked vertex, the algorithm has to have backscattering at the neighbouring vertices. Otherwise, the walk would not be able to localise at the marked vertex with a probability of more than 40%, as can be seen for the 6-dimensional lattice in figure 4.5, for more than two time steps. Obviously, the 40% intensity at time t_1 would be shifted to the neighbouring vertices at $t_1 + 1$, and on a lattice without backscattering, the probability can not extend 20% at $t_1 + 2$. As the localisation on the target vertex is high for more than two time steps, some contributions must have been backscattered from the nearest neighbours of the target vertex.

The coin flip ruling the backscattering at the nearest neighbours of the target vertex is the unperturbed local coin flip as defined in equation (4.1.1). That is the backscattering is defined through the diagonal terms as $\frac{1}{d} - 1$, while the factors for the amplitudes of

the outgoing waves in all other directions are $\frac{1}{d}$.

It can be seen that the coin flip acts differently in 2 dimensions. In this lattice, backscattering is singled out only in sign and the magnitude for backscattering is equal to the magnitude for scattering in any other direction. The lattices in more dimensions single backscattering out in sign and magnitude as the probability for backscattering is higher than for any other direction.

One may thus conclude that backscattering enables the algorithm to trap the amplitude more efficiently at the marked vertex and that the search will thus have an improved localisation if backscattering is increased. To investigate if the poor result for the search on the 2-dimensional lattice is connected to the lower probability for backscattering some numerical results are considered in section 6.2.

An argument against higher backscattering is that it is important that the walk is able to spread 'fast' through the network. When the walk is started in the uniform distribution and no marked vertex is introduced, it will remain there because the uniform distribution is an eigenstate of the unperturbed walk. As soon as a vertex is marked by a different coin flip, the 'knowledge' of the perturbation starts to spread through the graph. 3-dimensional simulations on the 2-dimensional graph as shown in figure 4.1 and figure 4.5 for several time steps give the impression that the marked vertices attract the probability right from the start. To localise the probability at the target vertex the information has to spread through the graph and to increase backscattering may be the wrong approach to make the search more efficient. Thus the amount of backscattering may have to be balanced between these two effects.

The performance of three different coins will be analysed in this chapter, the 'Kottos-Smilansky coin', the Fourier coin and an algorithm based on a coin without backscattering will be investigated.

6.2 The Kottos-Smilansky coin

The most promising coin flip is the Kottos-Smilansky coin. This coin matrix, introduced by Kottos and Smilansky in 1999 [37], is a generalisation of the coin used in chapter 4 and is defined as

$$\sigma_{KS}(\mu) = \left(1 + \mathrm{e}^{-\mathrm{i}\mu}\right) |s\rangle \langle s| - \mathbb{1}_{2d}, \tag{6.2.1}$$

where $|s\rangle$ is defined as in chapter 4, that is, $|s\rangle = \frac{1}{\sqrt{2d}} \sum_{i=1}^{d} \left(|i^+\rangle + |i^-\rangle\right)$ and μ is a real parameter.

This matrix has the advantage that on the one hand, it is a generalisation of the coin used in chapter 4 since the this coin is represented by the $\mu = 0$ case. On the other hand, the parameter $\mu \in [0, \pi]$ can be varied to increase backscattering continuously. It is certainly not wise to choose $\mu = \pi$, because for this choice the probability for backscattering equals 1 and the algorithm cannot localise anywhere as the amplitude is trapped on the bond where it started.

The global coin flip is defined again by the local coin flip on all vertices, that is $C(\mu) = \sigma_{KS}(\mu) \otimes \mathbb{1}_N$.

Note that in terms of the parameter μ the distance of the local coin and the old perturber coin $-\mathbb{1}_{2d}$ is reduced if $\mu \neq 0$ is chosen since the perturber coin used earlier equals $\sigma' = \sigma_{KS}(\pi)$. To make up for this, a new perturber coin $\sigma'(\mu) = \sigma_{KS}(\pi + \mu)$ is introduced.

To obtain the perturbed global coin, the local coin at the target vertex $|v\rangle$ is replaced by the perturber coin

$$C'(\mu) = C(\mu) - \left(\sigma_{KS}(\mu) - \sigma'(\mu)\right) \otimes |v\rangle \langle v| \tag{6.2.2}$$

$$= C(\mu) - 2\mathrm{e}^{-\mathrm{i}\mu} |s\rangle \langle s| \otimes |v\rangle \langle v|. \tag{6.2.3}$$

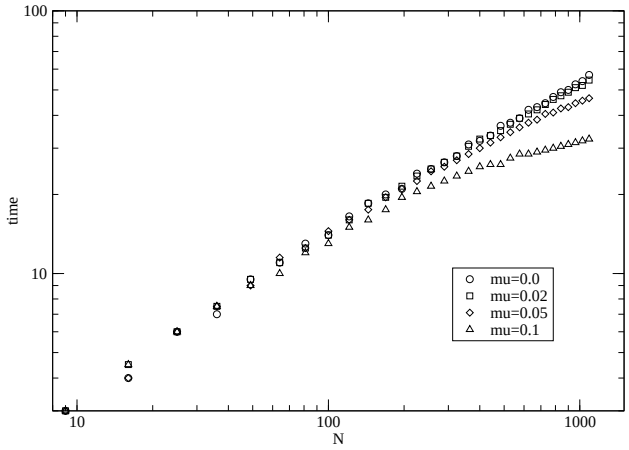

Figure 6.1: Time needed for the localisation on the marked vertex for the search algorithms for several μ as a function of N.

Since $|s\rangle$ is a $e^{-i\mu}$-eigenvector of $\sigma(\mu)$

$$U'(\mu) = SC'(\mu) \tag{6.2.4}$$

$$= S\left(C(\mu) - 2C(\mu)\,|s\rangle\langle s| \otimes |v\rangle\langle v|\right) \tag{6.2.5}$$

$$= U(\mu) - 2U(\mu)\,|sv\rangle\langle sv| \tag{6.2.6}$$

results in a generalised search algorithm where backscattering can easily be varied.

Some numerical simulations for this search algorithm will be presented in the following section.

6.2.1 Numerics

The numerical simulation for the success of the search in 2 dimensions for some μ, including the old search algorithm represented by $\mu = 0$, are shown in figure 6.1. It can be seen that the search algorithms for $\mu \neq 0$ scale faster than $\sqrt{N \ln N}$, since the distance in time between the different algorithms increases rapidly for large N.

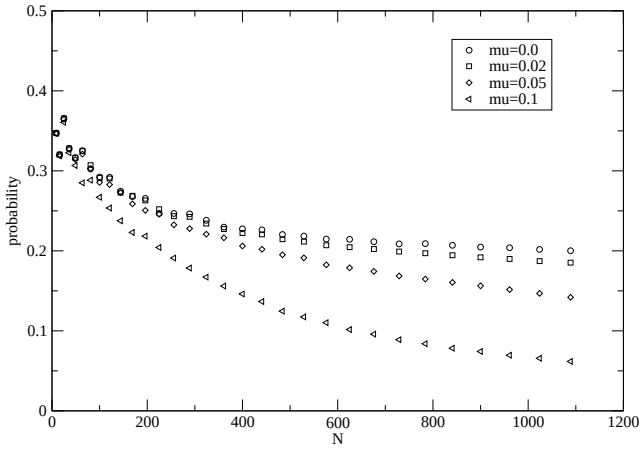

Figure 6.2: Probability at the target vertex as a function of N for the search algorithms shown in figure 6.1.

Thus backscattering affects the localisation process and localisation occurs faster when backscattering is increased.

However, figure 6.2 shows that the search algorithms for $\mu \neq 0$ do not localise as much probability as the $\mu = 0$ algorithm. In the limit $N \to \infty$, the probabilities of the $\mu \neq 0$ algorithms decrease even faster than the algorithm introduced by Ambainis, Kempe and Rivosh. To obtain a constant success probability amplitude amplification methods like in [59] have to be used and the search needs to be repeated several times.

Overall, a detailed calculation is necessary to determine if the algorithm is more efficient for $\mu \neq 0$ since the faster search is balanced by a higher number of repetitions to obtain a success probability for the algorithm that does not change with N.

Thus the numerical results do not confirm the arguments in section 6.1. It was expected that the localisation will improve if backscattering is increased and that backscattering might lead to longer search times. However, the figures display that the scaling behaviour of the time improves and the localisation amplitude decreases when μ was

increased.

To understand this effects a more detailed analysis of the spectrum of the search algorithm would be necessary. Unfortunately there was no time to do this as part of this thesis.

6.3 Other test coins

6.3.1 The Fourier-coin

The second coin to be tested is the Fourier-coin

$$\sigma_F = \frac{1}{\sqrt{2d}} \sum_{i,j=1}^{2d} \mathrm{e}^{\mathrm{i}\pi ij/d} \left|i\right\rangle \left\langle j\right| , \tag{6.3.1}$$

where the coin space span vectors where labelled using i and j, and the summation indices are varied from 1 to $2d$ for convenience. Note that this does not coincide with the notation above where $i \in [0, d]$ and the basis states have been labelled as $|i^+\rangle$ and $|i^-\rangle$. However, for $i \in [0, d]$ the new labels for the basis states can be mapped to the old set according to

$$\begin{aligned} |i\rangle \quad &\leftrightarrow \quad |i^+\rangle \\ |i+d\rangle \quad &\leftrightarrow \quad |i^-\rangle \end{aligned} \tag{6.3.2}$$

and the old definition for the shift operator defined in (4.1.3) can be employed.

The most successful marking coin I could find for this algorithm is

$$\sigma_F' = \sum_{i=1}^{2d} \mathrm{e}^{\mathrm{i}\pi i/d} \left|i\right\rangle \left\langle i\right| . \tag{6.3.3}$$

Some results for the search will be shown and discussed in section 6.4.

6.3.2 Coin without backscattering

Harrison, Smilansky and Winn investigated graphs where backscattering is prohibited [62]. They introduced a coin

$$\sigma_{HSW} = \frac{1}{\sqrt{2d-1}} \left(H - \mathbb{1}_{2d} \right), \tag{6.3.4}$$

where H is an orthogonal $2d \times 2d$ matrix with entries ± 1 fulfilling $H + H^{\dagger} = 2\mathbb{1}_{2d}$. For the simulations shown in figure 6.3, the Matrix

$$H = \begin{pmatrix} 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ -1 & 1 & 1 & -1 \\ 1 & 1 & 1 & 1 \end{pmatrix} \tag{6.3.5}$$

was used.

According to the claim that backscattering at the vertices adjacent to the target vertex is important, these vertices have been marked as well as the target vertex. The best results where obtained for the marking coin $\sigma' = 2 |s\rangle \langle s| - \mathbb{1}_{2d}$.

6.4 Comparison of the results for different coins

Figure 6.3 shows the outcome of three simulations using the standard coin $\sigma = 2 |s\rangle \langle s| - \mathbb{1}_{2d}$, the Fourier coin and the coin without backscattering on a 31×31 lattice.

The algorithm based on the Fourier coin and the one based on the HSW-coin cannot compete with the search algorithm discussed in chapter 4. The localisations obtained using these coins have a maximum comparable to the probability of the uniform distribution. The algorithms achieve an enhancement compared to the uniform distribution of five or two times respectively. Additional to that poor result, the maxima are near the 50th time step, so they are not even faster. Although the maximum of the standard

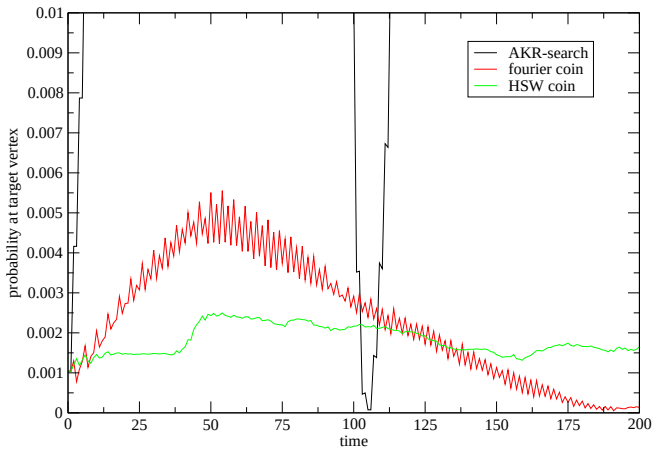

Figure 6.3: Probability on the marked vertex for different coins on a 31×31 lattice.

search algorithm is not visible, it is reached at about the same time. This can be seen from the minimum since the localisation takes half of the time of the first minimum which follows from a calculation analogue to the calculations in sections 4.4 and 3.4.

One surely has to analyse the spectrum of the algorithms simulated in this chapter to understand why they show the behaviour displayed in this chapters figures. Due to lack of time, this analysis has not been done in the scope of this thesis but nevertheless, some criteria for efficient search algorithms have been identified.

It seems crucial to have an avoided crossing in the spectra of the search algorithm and that the eigenvectors of the two eigenvalues that avoid the crossing are linear combinations of a state localised at the target vertices and a totally symmetric state that can be used as starting state. From section 3.4.1 it is known that the oscillation between this two states that leads to a localisation is the faster, the bigger the gap, that is $t = \frac{\pi}{\Delta}$, where Δ is the distance in the gap. Furthermore, the eigenvalues in the avoided crossings have to be separated from the others by a distance larger than Δ such that the main interaction is between them. Note that the last condition, the separation, applies only

to eigenvalues that are affected by the perturbation that changes the quantum random walk into a search algorithm. Both systems discussed in this thesis, the hypercube and the lattice search, had a highly degenerate eigenspace with eigenphase 0, that is, in the centre of the crossing, which where not affected by the perturbation parameter λ.

The last observation that shall be noted is that the coin of the hypercube and lattice search is $\sigma = 2|s\rangle\langle s| - \mathbb{1}_{2d}$. This is a very special choice as it coincides with the matrix D defined by $D_{ii} = -1 + \frac{2}{N}$ and $D_{ij} = \frac{2}{N}$ for $i \neq j$ used for Grover's algorithm 2.3.1. Thus the search algorithms using this coin have more similarities with Grover's algorithm than the search algorithms using different coins. Being close to Grover's algorithm might be a condition for an optimal search algorithm as it can be shown that Grover's algorithm scales like an optimal search algorithm [5, 23, 24].

CHAPTER 7

Conclusions

A new method to analyse search algorithms on graphs has been introduced and successfully applied to two search algorithms previously introduced in [25, 50]. For both systems, only the scaling of search time and localisation amplitude has been known before while this thesis gives the leading order terms of both quantities for the d-dimensional hypercube and for the 2 and 3 dimensional lattices. For a $d > 3$ dimensional lattice only the order has been calculated here but the leading order terms can be obtained using numerical integrations as has been discussed in section 4.3.

The new model for the analysis of the search algorithm containing one target vertex introduces a 2×2 model Hamiltonian that characterises an avoided crossing in the spectrum of a one parameter family of unitary operators. As a function of the parameter these operators change continuously from the unperturbed quantum walk to the search algorithm and back. Notably the crossing occurs exactly at the same position in parameter space as the search algorithm. It has also been shown that the localisation process of the later occurs due to oscillations between two eigenstates of the operator describing the search algorithm.

The search algorithm has then been generalised to find an arbitrary number of $m \ll N$

target vertices by generalising the model Hamiltonian to obtain a $(m+1) \times (m+1)$ Hamiltonian. This Hamiltonian has been used to calculate the search time for m target vertices. Compared to the $m = 1$ case the search time is reduced by a factor $\frac{1}{\sqrt{m}}$.

Apart from the new model, this thesis introduces new applications of search algorithms. If more than one vertex is marked, the algorithm can be used to send a signal from one vertex, the 'sender', to the other marked vertices which act as receivers. After a time which is twice the search time, a signal that was started at the sender is localised at the receivers with some amplitude remaining at the sender. Throughout this process, the signal on the unaffected vertices is small compared to the signals on sender and receivers. Remarkably, in the case of one receiver the signal is transferred in full. Additionally, the sender can, by tracking the local amplitude at its position, estimate how many receivers are 'listening' to the signal.

This sender-receiver scenario has been calculated analytically for the lattice search as well as for Grover's algorithm. The calculation for the lattice has been approved by some numerical simulations.

In a different setting the sender can send a continuous signal through a damped system. Again the signal is transferred to the receivers, an effect that can be used as search algorithm. This search demands no additional knowledge as required in the search algorithms starting in the uniform distributions as discussed in [23, 25, 50, 54]. This is because the measurement of the target vertices has no longer to be performed precisely at the localisation time which depends on the system size and the number of target vertices but at any time larger than that.

As described in section 5.6 the lattice search has some similarities to systems with crystalline structure and an experimental realisation using photonic crystals or solid states seem possible.

Some speculations about the effects causing the localisation at the target vertices are outlined in the last chapter and investigated using numerical simulations. It was assumed that increased backscattering leads to a better localisation at the target vertex and decreases the time of the search. The result of the simulation was quite unexpected and a detailed analysis would be necessary to understand the simulations. Furthermore the simulations for other test coins showed that the choice for the coin flip is crucial for the success of the search.

This thesis introduced a new model to investigate quantum search algorithms and found some interesting new applications. Yet, there are many open questions that remain open or are inspired by this thesis.

It is not a new but surely an interesting question for future research to gain more understanding of how to design successful search algorithms. And how can the general model of search algorithms be transferred to enable efficient searches on a wider class of graphs? Especially for the two dimensional lattice search a lot room for improvement might be left as the algorithm localises in $T \sim \sqrt{N \ln N}$ time and the optimal time for quantum search algorithms scales like $\sqrt{N}$ [23, 24].

So far, only periodic boundary conditions of the lattice have been considered. For finite lattices without periodic boundary conditions the positions on the lattice are no longer equivalent and it might matter if the target vertices are close to the boundary or near the centre of the system. How will this change the search algorithm and the sender-receiver model?

Even for the lattice with periodic boundary conditions it is unclear how well the sender can distinguish between different numbers of receivers. In the model discussed in section 5.3 no interaction between the localised state was assumed but results from numerical simulations as presented in figure 5.5 proved that the spatial configuration of

sender and receivers does have an effect and different configurations lead to variations in the amplitudes at sender and receiver.

Fluctuations of the bond lengths for quantum random walks have been studied [13] resulting in a critical disorder in the lengths at which the walk looses its quantum speedup. It is surely interesting to analyse how sensitive the search algorithm is to those fluctuation or to external noise.

Additional it is interesting to investigate how and if experimental applications of the search algorithm will localise. A graph is a purely mathematical construction and the bonds and vertices have zero width, so the question arises if a real lattice with a nonzero bond width will also localise at a target vertex.

A different experimental application was discussed in section 5.6. It was speculated that the search might be used to find local perturbations in periodic systems like solid states or laser induced lattices. Is it possible to send a signal from one perturbation to another in these systems?

I am confident that this thesis represents one step of gaining a better understanding of quantum search algorithms and that at least some of this questions will be answered by future research.

APPENDIX A

Eigenvectors and eigenvalues of the quantum walk on the hypercube

The derivation of eigenvectors and eigenvalues in this appendix follows the discussion by Moore and Russell presented in [56].

A.1 Definitions and notations

The quantum random walk is defined by the two operators C and S in (3.1.2) and (3.1.3) respectively. To analyse the quantum walk, the Fourier transformed position space vector

$$|v_{\vec{k}}\rangle = |u_{\vec{k}}\rangle \otimes \left(\frac{1}{\sqrt{N}} \sum_{\vec{x}} (-1)^{\vec{x}\vec{k}} |x\rangle \right) \tag{A.1.1}$$

is considered, where $\vec{k}$ is a vector containing d entries equal to 0 or 1. Let U be applied to this state

$$U\left|v_{\vec{k}}\right\rangle = S\left(\sigma\left|u_{\vec{k}}\right\rangle\right) \otimes \left(\frac{1}{\sqrt{N}}\sum_{\vec{x}}(-1)^{\vec{x}\vec{k}}\left|x\right\rangle\right) \tag{A.1.2}$$

$$= \frac{1}{\sqrt{N}}\sum_{i=1}^{d}\sum_{\vec{x}}(-1)^{\vec{x}\vec{k}}\left|ix\oplus e_i\right\rangle\left\langle i\mid\sigma\mid u_{\vec{k}}\right\rangle, \tag{A.1.3}$$

where the operation $\oplus$ is the sum modulo 2 and e_i the unit vector in the ith dimension. A reorganisation of the sum over x transforms this result to

$$U\left|v_{\vec{k}}\right\rangle = \frac{1}{\sqrt{N}}\sum_{i=1}^{d}\sum_{\vec{x}}(-1)^{\vec{x}\vec{k}+k_i}\left|ix\right\rangle\left\langle i\mid\sigma\mid u_{\vec{k}}\right\rangle \tag{A.1.4}$$

$$= \left(\sum_{i=1}^{d}(-1)^{k_i}\left|i\right\rangle\left\langle i\right|\sigma\right)\left|u_{\vec{k}}\right\rangle\otimes\left|v_{\vec{k}}\right\rangle \tag{A.1.5}$$

$$= \left(M_{\vec{k}}\left|u_{\vec{k}}\right\rangle\right)\otimes\left|v_{\vec{k}}\right\rangle, \tag{A.1.6}$$

where $M_{\vec{k}}$ is defined as the matrix operator acting on the coin space component only.

The factorisation according to (A.1.1) simplifies the eigenvalue equation of U to N eigenvalue equations, one for each of the $M_{\vec{k}}$'s, in the d-dimensional coin space, where

$$M_{\vec{k}} = \begin{pmatrix} (-1)^{k_1}\left(\frac{2}{d}-1\right) & (-1)^{k_1}\frac{2}{d} & \cdots & (-1)^{k_1}\frac{2}{d} \\ (-1)^{k_2}\frac{2}{d} & (-1)^{k_2}\left(\frac{2}{d}-1\right) & \ddots & \vdots \\ \vdots & \ddots & \ddots & (-1)^{k_{d-1}}\frac{2}{d} \\ (-1)^{k_d}\frac{2}{d} & \cdots & (-1)^{k_d}\frac{2}{d} & (-1)^{k_d}\left(\frac{2}{d}-1\right) \end{pmatrix}. \tag{A.1.7}$$

To simplify the matrix, the basis can be reordered by a permutation matrix $T_{\vec{k}}$ that shifts the rows (columns) with $k_i = 0$ and $k_i = 1$ to the top (left) and bottom (right) respectively. The result is a block matrix $M = T_{\vec{k}}M_{\vec{k}}T_{\vec{k}} = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$, where A is a $(d-k)\times(d-k)$

matrix and D a $k \times k$ matrix. The blocks are

$$A = \begin{pmatrix} \frac{2}{d} - 1 & \frac{2}{d} & \cdots & \frac{2}{d} \\ \frac{2}{d} & \frac{2}{d} - 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \frac{2}{d} \\ \frac{2}{d} & \cdots & \frac{2}{d} & \frac{2}{d} - 1 \end{pmatrix} \tag{A.1.8}$$

$$B = \begin{pmatrix} \frac{2}{d} & \cdots & \frac{2}{d} \\ \vdots & \ddots & \vdots \\ \frac{2}{d} & \cdots & \frac{2}{d} \end{pmatrix} \tag{A.1.9}$$

$$C = \begin{pmatrix} -\frac{2}{d} & \cdots & -\frac{2}{d} \\ \vdots & \ddots & \vdots \\ -\frac{2}{d} & \cdots & -\frac{2}{d} \end{pmatrix} \tag{A.1.10}$$

$$D = \begin{pmatrix} -\frac{2}{d} + 1 & -\frac{2}{d} & \cdots & -\frac{2}{d} \\ -\frac{2}{d} & -\frac{2}{d} + 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & -\frac{2}{d} \\ -\frac{2}{d} & \cdots & -\frac{2}{d} & -\frac{2}{d} + 1 \end{pmatrix}. \tag{A.1.11}$$

Let $|1_{d-k}\rangle$ and $|0_{d-k}\rangle$ be two vectors with $d-k$ entries equal to $\frac{1}{\sqrt{d-k}}$ and 0, respectively. The vectors $|1_k\rangle$ and $|0_k\rangle$ are defined analogously. These definitions will be used to compose vectors in the d-dimensional coin space by writing $|ab\rangle$ where $|a\rangle$ is a $d-k$ vector that fixes the first $d-k$ entries and $|b\rangle$ fixes the remaining k entries.

Now the matrices A, B, C and D can be denoted as

$$A = \frac{2}{d}(d-k)\,|1_{d-k}\rangle\langle 1_{d-k}| - \mathbb{1}_{d-k} \tag{A.1.12}$$

$$B = \frac{2}{d}\sqrt{k(d-k)}\,|1_{d-k}\rangle\langle 1_k| \tag{A.1.13}$$

$$C = -\frac{2}{d}\sqrt{k(d-k)}\,|1_k\rangle\langle 1_{d-k}| \tag{A.1.14}$$

$$D = -\frac{2}{d}k\,|1_k\rangle\langle 1_k| + \mathbb{1}_k. \tag{A.1.15}$$

A.2 Eigenvalues and eigenvectors

Any $(d-k)$-dimensional vector $|u_{d-k}\rangle$ orthogonal to $|1_{d-k}\rangle$ fulfils the equations $A|u_{d-k}\rangle = -|u_{d-k}\rangle$ and $C|u_{d-k}\rangle = |0_k\rangle$ and therefore the composed vector $|u_{d-k}0_k\rangle$ is a -1-eigenvector according to

$$M|u_{d-k}0_k\rangle = -|u_{d-k}0_k\rangle . \tag{A.2.1}$$

Analogously, any k-dimensional vector $|u_k\rangle$ orthogonal to $|1_k\rangle$ can be composed with $|0_{d-k}\rangle$ to form a 1-eigenvector of M that is

$$M|0_{d-k}u_k\rangle = |0_{d-k}u_k\rangle . \tag{A.2.2}$$

So far $d-2$ eigenvectors with eigenvalue $+1$ or -1 have been identified. The remaining 2 eigenvectors are orthogonal to the eigenvectors determined above and are thus linear combinations of $|1_{d-k}0_k\rangle$ and $|0_{d-k}1_k\rangle$. Let

$$\left|v_{\vec{k}}\right\rangle = a|1_{d-k}0_k\rangle + b|0_{d-k}1_k\rangle \tag{A.2.3}$$

then there are two distinct solutions for the coefficients a and b such that $\left|v_{\vec{k}}\right\rangle$ is an eigenvector of M. These solutions are $a = \mp\frac{\mathrm{i}}{\sqrt{2}}$ and $b = \frac{1}{\sqrt{2}}$ since

$$\begin{aligned} M\left|v_{\vec{k}}\right\rangle &= \left(a - \frac{2a}{d}k + \frac{2b}{d}\sqrt{k(d-k)}\right)|1_{d-k}0_k\rangle \\ &\quad + \left(-\frac{2a}{d}\sqrt{k(d-k)} - \frac{2bk}{d} + b\right)|0_{d-k}1_k\rangle \qquad \text{(A.2.4)} \\ &= \mp\frac{\mathrm{i}}{\sqrt{2}}\left(1 - \frac{2k}{d} \pm \frac{2\mathrm{i}}{d}\sqrt{k(d-k)}\right)|1_{d-k}0_k\rangle \\ &\quad + \frac{1}{\sqrt{2}}\left(1 - \frac{2k}{d} \pm \frac{2\mathrm{i}}{d}\sqrt{k(d-k)}\right)|0_{d-k}1_k\rangle . \qquad \text{(A.2.5)} \end{aligned}$$

Note that in the cases $k = 0$ or d one of these vector components contains zero entries. Choosing the coefficient for the vector without entries equal to zero $b = 0$ and $a = 0$ are regarded for $k = 0$ and $k = d$, respectively. In these cases, normalisation requires an additional factor $\sqrt{2}$.

The two eigenvalues of M have an absolute value 1 and are given as

$$v_{\vec{k}}^{\pm} = \mathrm{e}^{\pm \mathrm{i}\omega_k} = 1 - \frac{2k}{d} \pm \frac{2\mathrm{i}}{d}\sqrt{k\,(d-k)} \tag{A.2.6}$$

and the eigenvectors are

$$|v_{\vec{k}}^{\pm}\rangle = \sum_{i=1}^{d}\sum_{\vec{x}} (-1)^{\vec{k}\cdot\vec{x}}\, \frac{2^{-d/2}}{\sqrt{2}}\, \alpha_{k_i}^{\pm}\beta_k\, |i,\vec{x}\rangle\,, \tag{A.2.7}$$

where

$$\alpha_{k_i}^{\pm} = \begin{cases} 1/\sqrt{k} & \text{if } k_i = 1 \\ \mp \mathrm{i}/\sqrt{d-k}, & \text{if } k_i = 0 \end{cases} \tag{A.2.8}$$

$$\beta_k = \begin{cases} \sqrt{2}, & \text{if } k = 0 \text{ or } k = d \\ 1 & \text{otherwise} \end{cases}. \tag{A.2.9}$$

Since the set of eigenvectors and eigenvalues of U are identical to the eigenvectors and eigenvalues of $M_{\vec{k}} = T_{\vec{k}} M T_{\vec{k}}$ the eigenvalues of U are also eigenvalues of M as the permutation matrix $T_{\vec{k}}$ does not change the eigenvalues. The $(d-k-1)$ eigenvectors with eigenvalue $+1$ and the $(k-1)$ eigenvectors with eigenvalue -1 of U are supported in the vector components where $k_i = 0$ and $k_i = 1$ and orthogonal to the vectors $\left((1,1,\ldots,1)^{\mathsf{T}} - \vec{k}\right)$ and $\vec{k}$, respectively. The remaining two eigenvectors are, in the form denoted in (A.2.7)-(A.2.9), invariant under the permutation defined by $T_{\vec{k}}$.

APPENDIX B

Theory of avoided crossings

Avoided crossings occur in perturbed quantum mechanical two-level systems. Since this model is used to analyse the localisation process of the search algorithms in chapters 3 and 4, a more detailed introduction will be added.

B.1 The perturbed two level-system

In the basis of eigenstates, a two level system has the Hamiltonian

$$H_0 = \begin{pmatrix} E_1 & 0 \\ 0 & E_2 \end{pmatrix}, \tag{B.1.1}$$

where E_1 and E_2 are the two eigenenergies. The eigenenergies depend on a parameter λ and cross for some distinct value, say $\lambda = 0$ whereas $E_1 \neq E_2$ for $\lambda \neq 0$. Trivially, the eigenvectors of H are

$$|v_1\rangle = (1,0)^{\mathsf{T}} \tag{B.1.2}$$

$$|v_2\rangle = (0,1)^{\mathsf{T}}. \tag{B.1.3}$$

A sketch of the crossing is drawn in figure B.1.

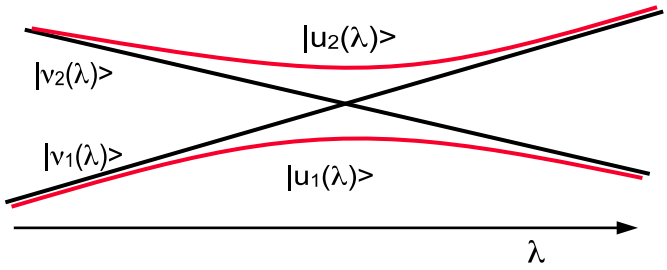

Figure B.1: Sketch of an avoided crossing.

The perturbation W is defined by a perturbation matrix $W = \begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{pmatrix}$, where the matrix elements w_{ij} are such that W is Hermitian, that is $w_{11}, w_{22} \in \mathbb{R}$ and $w_{21} = w_{12}^*$, where * denotes complex conjugation. Furthermore, the perturbation will be regarded as small compared to the typical differences of the entries of H_0. The diagonal elements of W can be neglected without loss of generality as they are only shifting the energy levels E_1 and E_2 and can be added easily into the final result by $E_i \to E_i + w_{ii}$.

This defines the Hamiltonian of the perturbed system as

$$H = H_0 + W = \begin{pmatrix} E_1 & w_{12} \\ w_{12}^* & E_2 \end{pmatrix} \tag{B.1.4}$$

and the eigenvalues of the perturbed Hamiltonian are

$$e_1 = \frac{E_1 + E_2}{2} + \frac{\Delta}{2} \tag{B.1.5}$$

$$e_2 = \frac{E_1 + E_2}{2} - \frac{\Delta}{2}, \tag{B.1.6}$$

where the difference in the energy levels is defined as $\Delta = \sqrt{(E_1 - E_2)^2 + 4\,|w_{12}|^2}$ and the corresponding non-normalised eigenvectors are

$$\left|u_1'\right\rangle = (E_1 - E_2 + \Delta, 2w_{12}^*)^{\mathsf{T}} \tag{B.1.7}$$

$$\left|u_2'\right\rangle = (E_1 - E_2 - \Delta, 2w_{12}^*)^{\mathsf{T}}. \tag{B.1.8}$$

Let $|u_1\rangle$ and $|u_2\rangle$ be the normalised eigenvectors, then in the limit $|w_{12}| \to 0$, the eigenvectors of H approach those of H_0 but how $|u_1\rangle$ and $|u_2\rangle$ approach $|v_1\rangle$ and $|v_2\rangle$

depends on the sign of $E_1 - E_2$ as

$$\Delta \approx |E_1 - E_2| \,. \tag{B.1.9}$$

Hence, for $E_1 - E_2 > 0$, $|u_1\rangle \to |v_1\rangle$ and $|u_2\rangle \to |v_2\rangle$, whereas on the other side of the crossing, that is for $E_1 - E_2 < 0$, the identification of eigenvectors is the other way around and $|u_1\rangle \to |v_2\rangle$ and $|u_2\rangle \to |v_1\rangle$. Therefore the eigenvectors and eigenvalues of H reorganise near the crossing and switch their limiting behaviour for $|\lambda| \to \infty$ approaching now the other eigenvector and eigenvalue of H_0 as can be seen in figure B.1.

B.2 Oscillations in the two level system

An interesting effect occurs when at $\lambda = 0$ the system is changed from H_0 to H. If the energy at the crossing is defined as $E_1 = E_2 = 0$, the Hamiltonian H takes the form

$$H_{\text{crossing}} = \begin{pmatrix} 0 & \omega_{12} \\ \omega_{12}^* & 0 \end{pmatrix} \tag{B.2.1}$$

and its eigenvectors and eigenvalues are

$$\begin{array}{cc} \text{eigenvector} & \text{eigenvalue} \\ \left|u_1^{\text{crossing}}\right\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} \mathrm{e}^{\mathrm{i}\omega} \\ 1 \end{pmatrix} & |\omega_{12}| \\ \left|u_2^{\text{crossing}}\right\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} -\mathrm{e}^{\mathrm{i}\omega} \\ 1 \end{pmatrix} & -|\omega_{12}| \end{array} , \tag{B.2.2}$$

where ω is the phase of ω_{12}.

If the system is initialised in one eigenstate of H_0, say in $|v_1\rangle$ when the perturbation is turned on, the state will evolve in time as $|v(t)\rangle = \mathrm{e}^{-\mathrm{i}H_{\text{crossing}}t} |v_1\rangle$, where $\hbar = 1$ was

chosen, and therefore

$$|v(t)\rangle = \frac{e^{-i\omega}}{\sqrt{2}} e^{-iH_{\text{crossing}}t} \left(\left| u_1^{\text{crossing}} \right\rangle - \left| u_2^{\text{crossing}} \right\rangle \right) \tag{B.2.3}$$

$$= \frac{e^{-i\omega}}{\sqrt{2}} \left(e^{-i|\omega_{12}|t} \left| u_1^{\text{crossing}} \right\rangle - e^{i|\omega_{12}|t} \left| u_2^{\text{crossing}} \right\rangle \right) \tag{B.2.4}$$

$$= \cos\left(|\omega_{12}|\, t\right) |v_1\rangle + e^{-i\omega - \frac{\pi}{2}} \sin\left(|\omega_{12}|\, t\right) |v_2\rangle . \tag{B.2.5}$$

It can be seen that the system oscillates between the two states $|v_1\rangle$ and $|v_2\rangle$ with frequency $|\omega_{12}|$.

A more detailed discussion can be found in textbooks on quantum mechanics like [63].

APPENDIX C

Eigenvectors and eigenvalues of the quantum walk on the d-dimensional lattice

The analysis in this appendix is based on the derivation of eigenvectors and eigenvalues in [25] by Ambainis, Kempe and Rivosh.

Using the definitions from section 4.1.2, the eigenvalue equation simplifies from $U\left|\phi_{\vec{k}}\right\rangle = \phi_{\vec{k}}\left|\phi_{\vec{k}}\right\rangle$ to $\left(\sigma_{\vec{k}}\left|u\right\rangle\right) \otimes \left|X_{\vec{k}}\right\rangle = \phi_{\vec{k}}\left|u_{\vec{k}}\right\rangle \otimes \left|X_{\vec{k}}\right\rangle$, where $\phi_{\vec{k}}$ is the eigenvalue and $\sigma_{\vec{k}} = M\sigma$ is a $2d \times 2d$ matrix defined using the local coin flip σ.

Let the basis vectors of the coin space $|i^{\pm}\rangle$ for $i = 1 \ldots d$ be mapped on the canonical basis vectors $|j\rangle$, $j = 1 \ldots 2d$ such that

$$\left|l^{+}\right\rangle \rightarrow |2l-1\rangle \tag{C.0.1}$$

$$\left|l^{-}\right\rangle \rightarrow |2l\rangle \tag{C.0.2}$$

for $l = 1 \ldots d$. In this basis, M is block diagonal matrix consisting of d blocks $M_i =$

$\begin{pmatrix} 0 & \omega^{k_i} \\ \omega^{-k_i} & 0 \end{pmatrix}$, that is,

$$M = \begin{pmatrix} M_1 & 0 & 0 & \\ 0 & M_2 & 0 & \dots \\ 0 & 0 & M_3 & \\ & \vdots & & \ddots \end{pmatrix}. \tag{C.0.3}$$

Hence $\left|\phi_{\vec{k}}\right\rangle$ is an $\phi_{\vec{k}}$-eigenvector of U, if and only if $|u\rangle$ is an eigenvector of $\sigma_{\vec{k}}$ with eigenvalue $\phi_{\vec{k}}$.

The set of eigenvectors, especially for arbitrary d, is hard to calculate although some of them are easily verified. The analysis in appendix C.1 gives the complete set of eigenvalues and discusses important properties of the eigenvectors.

C.1 Eigenvectors and eigenvalues

C.1.1 ± 1-eigenvectors of U.

Each of the blocks M_i has the two eigenvalues ± 1 with corresponding eigenvectors $\frac{1}{\sqrt{2}}\left(\omega^{\frac{k_i}{2}}, \pm\omega^{-\frac{k_i}{2}}\right)^{\mathsf{T}}$. Therefore the matrix M has eigenvalues ± 1 with multiplicity d each. Both eigenspaces can be reorganised such that $d-1$ eigenvectors are orthogonal to $|s\rangle$ leaving two non-orthogonal normalised eigenvectors, say $\left|w_{\vec{k}}^{\pm 1}\right\rangle$.

Since $\sigma_{\vec{k}} = M\left(2\left|s\right\rangle\left\langle s\right| - \mathbb{1}_{2^d}\right)$, the ± 1-eigenvectors of M orthogonal to $|s\rangle$ are ∓ 1-eigenvectors of $\sigma_{\vec{k}}$. Let the $d-1$ eigenvectors of both eigenspaces be denoted as $\left|u_{j\vec{k}}^{\pm 1}\right\rangle$, where the index $j = 1, \dots, d-1$ numbers the eigenvectors of each eigenspace.

In the case of $\vec{k} = \vec{0}$ all -1-eigenvectors of M are orthogonal to $|s\rangle$ for the eigenvectors of the d blocks are $(1,-1)^{\mathsf{T}}$. Again, the 1-eigenvectors of M can be reorganised such

that $d-1$ of them are orthogonal to $|s\rangle$. As $|s\rangle$ is a 1-eigenvector of M, the eigenvalues of M are ± 1 with multiplicity d each.

The $d-1$ eigenvectors of M with eigenvalue 1 that are orthogonal to $|s\rangle$ are -1-eigenvectors of $\sigma_{\vec{0}}$ while $|s\rangle$ is a 1-eigenvector of M and a 1-eigenvector of $\sigma_{\vec{0}}$. The d eigenvectors of M with eigenvalue -1 are orthogonal to $|s\rangle$ and therefore 1-eigenvectors of $\sigma_{\vec{0}}$. In this case the 1 eigenspace has dimension $d+1$ and there are $d-1$ dimensions in the -1 eigenspace.

Note that the 1-eigenvector of U defined as $|\phi_0\rangle := |s\rangle \otimes \left|X_{\vec{0}}\right\rangle$ is a state uniformly distributed in coin and position space.

C.1.2 Remaining eigenvectors and eigenvalues for $\vec{k} \neq \vec{0}$ of U

Let the two remaining eigenvectors of M for $\vec{k} \neq \vec{0}$, that is, the linear combinations of the eigenvectors of M that are not orthogonal to $|s\rangle$, be $\left|w_{\vec{k}}^{+1}\right\rangle$ and $\left|w_{\vec{k}}^{-1}\right\rangle$. Since $|s\rangle$ lies in the plane spanned by these vectors, two coefficients α_1 and α_2 with $|\alpha_1|^2 + |\alpha_2|^2 = 1$ can be found such that $|s\rangle = \alpha_1 \left|w_{\vec{k}}^{+1}\right\rangle + \alpha_2 \left|w_{\vec{k}}^{-1}\right\rangle$. Since $\left|w_{\vec{k}}^{+1}\right\rangle$ and $\left|w_{\vec{k}}^{-1}\right\rangle$ are both eigenvectors of the unitary matrix M for different eigenvalues and therefore orthogonal, the vector $\left|s^{\perp}\right\rangle = \alpha_2^* \left|w_{\vec{k}}^{+1}\right\rangle - \alpha_1^* \left|w_{\vec{k}}^{-1}\right\rangle$ is orthogonal to $|s\rangle$. Here * denotes complex conjugation.

In return the vectors $\left|w_{\vec{k}}^{\pm 1}\right\rangle$ can be expanded according to

$$\left|w_{\vec{k}}^{\pm 1}\right\rangle = \left(|s\rangle\langle s| + \left|s^{\perp}\right\rangle\left\langle s^{\perp}\right|\right)\left|w_{\vec{k}}^{\pm 1}\right\rangle \tag{C.1.1}$$

and

$$\left|w_{\vec{k}}^{+1}\right\rangle = \alpha_1^* |s\rangle + \alpha_2 \left|s^{\perp}\right\rangle \tag{C.1.2}$$

$$\left|w_{\vec{k}}^{-1}\right\rangle = \alpha_2^* |s\rangle - \alpha_1 \left|s^{\perp}\right\rangle \tag{C.1.3}$$

is obtained. Note that the vectors are still undefined as $\left|s^{\perp}\right\rangle$ cannot be evaluated easily for general d and $\vec{k}$.

To obtain the eigenvalues, let $|u\rangle = \beta_1 |s\rangle + \beta_2 \left|s^{\perp}\right\rangle$ be one of the two eigenvectors of $\sigma_{\vec{k}}$ not orthogonal to $|s\rangle$ with $|\beta_1|^2 + |\beta_2|^2 = 1$. Now $\left|u^{\perp}\right\rangle = \beta_2^* |s\rangle - \beta_1^* \left|s^{\perp}\right\rangle$ is a vector in the plane orthogonal to $|u\rangle$ and

$$|s\rangle = \beta_1^* |u\rangle + \beta_2 \left|u^{\perp}\right\rangle \tag{C.1.4}$$

$$\left|s^{\perp}\right\rangle = \beta_2^* |u\rangle - \beta_1 \left|u^{\perp}\right\rangle. \tag{C.1.5}$$

Using these definitions the eigenvalues can be determined. First $\sigma |u\rangle$ can be evaluated by expanding $|u\rangle$ in the eigenbases of σ using $|s\rangle$ and $\left|s^{\perp}\right\rangle$. Then M can be applied to an expansion of $\sigma |u\rangle$ in the eigenbasis of M. One obtains

$$\sigma_{\vec{k}} |u\rangle = (\alpha_1\beta_1 - \alpha_2^*\beta_2) \left|w_{\vec{k}}^{+1}\right\rangle + (-\alpha_2\beta_1 - \alpha_1^*\beta_2) \left|w_{\vec{k}}^{-1}\right\rangle. \tag{C.1.6}$$

This vector is then transformed back into the $|u\rangle$, $\left|u^{\perp}\right\rangle$ basis which yields

$$\begin{aligned}\sigma_{\vec{k}} |u\rangle = &\left(|\alpha_1|^2 - |\alpha_2|^2 + 4\mathrm{i}\,\mathrm{Im}\ (\alpha_1\alpha_2\beta_1\beta_2^*)\right) |u\rangle \\ &+ \left(-2\alpha_1^*\alpha_2^*\beta_2^2 - 2\alpha_1\alpha_2\beta_1^2\right) \left|u^{\perp}\right\rangle.\end{aligned} \tag{C.1.7}$$

Since $|u\rangle$ was defined as eigenvector of $\sigma_{\vec{k}}$, the second term has to be zero, that is, $\alpha_1^*\alpha_2^*\beta_2^2 + \alpha_1\alpha_2\beta_1^2 = 0$.

The real part of the eigenvalue $|\alpha_1|^2 - |\alpha_2|^2$ can be calculated by noting that $\langle s| \sigma_{\vec{k}} |s\rangle = |\alpha_1|^2 - |\alpha_2|^2$ and $\langle s| \sigma_{\vec{k}} |s\rangle = \frac{1}{2d} \sum_{i=1}^{d} \left(\omega^{k_i} + \omega^{-k_i}\right) = \frac{1}{d} \sum_{i=1}^{d} \cos \frac{2\pi k_i}{n}$. Therefore the two eigenvectors of type $|u\rangle$ have eigenvalues with identical real parts. Since the determinant of $\sigma_{\vec{k}}$ is real, they have to be complex conjugated.

Now the two eigenvalues of $\left|u_{\vec{k}}^{\pm}\right\rangle$ can be identified as $\mathrm{e}^{\pm\mathrm{i}\theta_{\vec{k}}}$, where $\mathrm{Re}\,\mathrm{e}^{\pm\mathrm{i}\theta_{\vec{k}}} = \cos\theta_{\vec{k}} = \frac{1}{d} \sum_{i=1}^{d} \cos \frac{2\pi k_i}{n}$ and the phase of the eigenvector is defined such that $\left\langle s \mid u_{\vec{k}}^{\pm} \right\rangle$ is real and greater zero.

C.2 Calculation of the scalar product of the coin space contributions with $|s\rangle$

As discussed in previous sections M is a block diagonal matrix consisting of d blocks

$$M_i = \begin{pmatrix} 0 & \omega^{k_i} \\ \omega^{-k_i} & 0 \end{pmatrix},$$

$$M = \begin{pmatrix} M_1 & 0 & 0 & \\ 0 & M_2 & 0 & \dots \\ 0 & 0 & M_3 & \\ & \vdots & & \ddots \end{pmatrix}, \tag{C.2.1}$$

where $\omega = \mathrm{e}^{2\pi \mathrm{i}/n}$ [25].

A block diagonal matrix T with $TT = \mathbb{1}$ defined as

$$T = \begin{pmatrix} 0 & 1 & 0 & 0 & \\ 1 & 0 & 0 & 0 & \dots \\ 0 & 0 & 0 & 1 & \\ 0 & 0 & 1 & 0 & \\ & \vdots & & & \ddots \end{pmatrix}, \tag{C.2.2}$$

can be used to perform a complex conjugation on M since the transformation TMT results in M^* and, due to $T\sigma T = \sigma$, the transformation matrix has the same effect on $\sigma_{\vec{k}}$ since $T\sigma_{\vec{k}}T = \sigma^*_{\vec{k}}$.

Comparison of the eigenvalue equation in coin space.

$$\sigma_{\vec{k}} \left| u^+_{\vec{k}} \right\rangle = \mathrm{e}^{\mathrm{i}\theta_{\vec{k}}} \left| u^+_{\vec{k}} \right\rangle \tag{C.2.3}$$

and its complex conjugate

$$\sigma_{\vec{k}}^{*} \left| u_{\vec{k}}^{+} \right\rangle^{*} = \mathrm{e}^{-\mathrm{i}\theta_{\vec{k}}} \left| u_{\vec{k}}^{+} \right\rangle^{*} \tag{C.2.4}$$

$$T\sigma_{\vec{k}}T \left| u_{\vec{k}}^{+} \right\rangle^{*} = \mathrm{e}^{-\mathrm{i}\theta_{\vec{k}}} \left| u_{\vec{k}}^{+} \right\rangle^{*} \tag{C.2.5}$$

results in the relation $T \left| u_{\vec{k}}^{+} \right\rangle^{*} = \left| u_{\vec{k}}^{-} \right\rangle$ between these two eigenvectors of $\sigma_{\vec{k}}$.

The phase of $\left| u_{\vec{k}}^{+} \right\rangle$ is defined by choosing $\left\langle s \mid u_{\vec{k}}^{+} \right\rangle$ to be real and greater zero. Since $|s\rangle$ is a 1-eigenvector of T,

$$\left\langle s \mid u_{\vec{k}}^{+} \right\rangle = \left\langle s \mid u_{\vec{k}}^{-} \right\rangle \qquad \in \mathbb{R} \tag{C.2.6}$$

is concluded.

The eigenvectors $|ev\rangle$ of $\sigma_{\vec{k}}$ can easily be organised to be orthonormal and the identity operator in coin space can be denoted as a sum of projectors on all eigenvectors

$$\mathbb{1}_{2d} = \sum_{|ev\rangle} |ev\rangle \langle ev| . \tag{C.2.7}$$

By multiplication with $|s\rangle$ from both sides, and using that all ± 1-eigenvectors are orthogonal to $|s\rangle$ [25]

$$1 = \left| \left\langle s \mid u_{\vec{k}}^{+} \right\rangle \right|^2 + \left| \left\langle s \mid u_{\vec{k}}^{-} \right\rangle \right|^2 \tag{C.2.8}$$

is obtained.

Using (C.2.6), the promised result $\left\langle s \mid \ u_{\vec{k}}^{\pm} \right\rangle = \frac{1}{\sqrt{2}}$ is derived.

Bibliography

[1] B. Hein and G. Tanner. Quantum search algorithms on the hypercube. *J. Phys. A*, **42**:085303, 2009.

[2] B. Hein and G. Tanner. Wave communications across regular lattices. *Phys. Rev. Lett.*, **103**:260501, 2009.

[3] B. Hein and G. Tanner. Quantum search algorithms on a regular lattice. Accepted for Publication in Phys. Rev. A, 2010.

[4] N. L. Biggs, E. K. Lloyd, and R.J. Wilson. *Graph Theory 1736 - 1936*. Clarendon Press, Oxford, 1976.

[5] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, Cambridge, 2000.

[6] R. S. Braich, N. Chelyapov, C. Johnson, P. W. K. Rothemund, and L. Adleman. Solution of a 20 variable 3-SAT problem on a DNA computer. *Science*, **296** :499, 2002.

[7] Y. Aharonov, L. Davidovich, and N. Zagury. Quantum random walks. *Phys. Rev. A*, **48**(2):1687, 1993.

[8] A. Ambainis, E. Bach, A. Nayak, A. Vishwanath, and J. Watrous. One-dimensional quantum walks. In *Proceedings of the 33rd ACM Symposium on Theory of Computing*, page 37, 2001.

[9] J. Kempe. Quantum random walks - an introductory overview. *Contemporary Physics*, **44**:307, 2003.

[10] D. Aharonov, A. Ambainis, J. Kempe, and U. Vazirani. Quantum walks on graphs. In *Proceedings of the 33rd ACM Symposium on Theory of Computing*, page 50, 2001.

[11] A. M. Childs, E. Farhi, and S. Gutmann. An example of the difference between quantum and classical random walks. *Quantum Information Processing*, 1(1/2):35, 2002.

[12] J. Kempe. Discrete quantum walks hit exponentially faster. In *Lecture Notes in Computer Science*, volume **2764** , page 781, Berlin/Heidelberg, 2003. Springer.

[13] J. P. Keating, N. Linden, J. C. F. Matthews, and A. Winter. Localization and its consequences for quantum walk algorithms and quantum communication. *Phys. Rev. A*, **76**:012315, 2007.

[14] V. Kendon. Decoherence in quantum walks - a review. *Mathematical Structures in Comp. Sci*, **17** (6):1169, 2007. Preprint at arXiv:quant-phys/0606016.

[15] V. Kendon. A random walk approach to quantum algorithms. *Phil. Trans. R. Soc. A*, **364** :3407, 2006.

[16] M. Santha. Quantum walk based search algorithms. *Proceedings of the 5th conference on theory and applications of models of computation*, **4978** :31, 2008. Preprint at arXiv:quant-phys/0808.0059v1.

[17] P. Xue, B. C. Sanders, and D. Leibfried. Quantum walk on a line for a trapped ion. *Phys. Rev. Lett.*, 103:183602, 2009.

[18] H. Schmitz, R. Matjeschk, C. Schneider, J. Glueckert, M. Enderlein, T. Huber, and T. Schaetz. Quantum walk of a trapped ion in phase space. *Phys. Rev. Lett.*, 103: 090504, 2009.

[19] F. Zähringer, G. Kirchmair, R. Gerritsma, E Solano, R. Blatt, and C. F. Ross. Realization of a quantum walk with one and two trapped ions. *Phys. Rev. Lett.*, 104:100503, 2010.

[20] A. Schreiber, K. N. Cassemiro, V. Potoček, A. Gábris, P. J. Mosley, E. Andersson, I. Jex, and C. Silberhorn. Photons walking on the line: A quantum walk with adjustable coin operations. *Phys. Rev. Lett.*, 104:050502, 2010.

[21] M. A. Broome, A. Fedrizzi, B. P. Lanyon, I. Kassal, A. Aspuru-Guzik, and A.G. White. Discrete single-photon quantum walk with turnable decoherence. *Phys. Rev. Lett.*, 104:153602, 2010.

[22] G. Tanner. From quantum graphs to quantum random walks. In *Non-Linear Dynamics and Fundamental Interactions*, volume 213, page 69, Springer, Dordrecht, 2006. Preprint at arXiv:quant-phys/0504224.

[23] L. K. Grover. A fast quantum mechanical algorithm for database search. In *Proc. 28th STOC*, pages 212–219, ACM Press, Philadelphia, Pennsylvania, 1996.

[24] C. H. Bennett, E. Bernstein, G. Brassard, and U. Vazirani. Strengths and weaknesses of quantum computing. *SIAM J. Comput*, **26**(5):1510, 1997.

[25] A. Ambainis, J. Kempe, and A. Rivosh. Coins make quantum walks faster. In *Proceedings of the sixteenth annual ACM-SIAM symposium on discrete algorithms*, pages 1099–1108, Philadelphia, 2005. Society for industrial and applied mathematics.

[26] A. M. Childs and J. Goldstone. Spatial search by quantum walk. *Phys. Rev. A*, **70**: 022314, 2004.

[27] A. M. Childs and J. Goldstone. Spatial search and the Dirac equation. *Phys. Rev. A*, **70**:042312, 2004.

[28] V. Potoček, A. Gábris, T. Kiss, and I. Jex. Optimized quantum random-walk search algorithms on the hypercube. *Phys. Rev. A*, **79**:012325, 2009.

[29] A. M. Turing. On computable numbers, with an application to the Entscheidungsproblem. In *Proceedings of the London Mathematical Socociety 2*, volume **42**, page 230, 1936.

[30] R. P. Feynman. Simulating physics with computers. *Int. J. Theor. Phys.*, **21**(6/7):467, 1982.

[31] D. Deutsch. Quantum theory, the church-turing principle and the universal quantum computer. *Proc. R. Soc. Lond. A*, **400**:97, 1985.

[32] D. Deutsch and R. Jozsa. Rapid solution of problems by quantum computation. *Proc. R. Soc. Lond. A.*, **439** :553, 1992.

[33] P. W. Shor. Polynoial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.*, **26** (5):1484, 1997.

[34] A. Politi, J. C. F. Matthews, and J. L. O'Brien. Shor's quantum factoring algorithm on a photonic chip. *Science*, 325(5945):1221, 2009.

[35] F. Barra and P. Gaspard. Classical dynamics on graphs. *Phys. Rev. E*, 63(6):066215, 2001.

[36] S. Gnutzmann and U. Smilansky. Quantum graphs: Applications to quantum chaos and universal spectral statistics. *Advances in Physics*, **55**:527, 2006.

[37] T. Kottos and U. Smilansky. Periodic orbit theory and spectral statistics for quantum graphs. *Annals of Physics*, **274**:76, 1999.

[38] S. Severini and G. Tanner. Regular quantum graphs. *J. Phys. A*, **37**:6675, 2004.

[39] L. Pauling. The diamagnetic anisotropy of aromatic molecules. *J. Chem. Phys.*, **4**: 673, 1936.

[40] T. Kottos and U. Smilansky. Quantum chaos on graphs. *Phys. Rev. Lett.*, **79**(3):4794, 1997.

[41] T. Kottos and H. Schanz. Quantum graphs: a model for quantum chaos. *Physica E: Low-dimensional Systems and Nanostructures*, 9(3):523, 2001.

[42] P. Schapotschnikow and S. Gnutzmann. Spectra of graphs and semi-conductin polymers. *Proceedings of Symposia in Pure Mathematics*, **77** :691, 2008.

[43] K. Ruedenberg and C. W. Scherr. Free-electron network model for conjugated systems. I. theory. *J. Chem. Phys*, **21**:1565, 1953.

[44] F. Barra and P. Gaspard. Transport and dynamics on open quantum graphs. *Phys. Rev. E*, **65** (21):16205, 2001.

[45] H. Schanz and U. Smilansky. Periodic-orbit theory of Anderson localization on graphs. *Phys. Rev. Lett.*, **84** (7):1427, 2000.

[46] T. Kottos and U. Smilansky. Chaotic scattering on graphs. *Phys. Rev. Lett.*, 85(5):968, 2000.

[47] M. Puhlmann, H. Schanz, T. Kottos, and T. Geisel. Quantum decay of an open chaotic system: A semiclassical approach. *Europhys. Lett.*, **69** :313, 2005.

[48] S. Gnutzmann, J. P. Keating, and F. Piotet. Quantum ergodicity on graphs. *Phys. Rev. Lett.*, **101** :264102, 2008.

[49] O. Hul, S. Bauch, P. Pakoński, N. Savytskyy, K. Życzkowski, and L. Sirko. Experimental simulation of quantum graphs by microwave networks. *Phys. Rev. E*, 69(5): 056205, 2004.

[50] N. Shenvi, J. Kempe, and K. B. Whaley. Quantum random-walk search algorithm. *Phys. Rev. A*, **67**:052307, 2003.

[51] A. M. Childs. On the relationship between continuous- and discrete-time quantum walk. *Commun. Math. Phys.*, **294**:581, 2010.

[52] E. Farhi and S. Gutmann. Quantum computation and decision trees. *Phys. Rev. A*, **58**(2):915, 1998.

[53] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders. Efficient quantum algorithms for simulating sparse hamiltonians. *Communications in Mathematical Physics*, **270**(2): 359, 2007.

[54] L. K. Grover. Quantum mechanics helps in searching for a needle in a haystack. *Phys. Rev. Lett.*, **79**:325, 1997.

[55] N. Bhattacharya, H. B. van Linden van den Heuvell, and R. J. C. Spreeuw. Implementation of quantum search algorithm using classical fourier optics. *Phys. Rev. Lett.*, **88**:137901, 2002.

[56] C. Moore and A. Russell. Quantum walks on the hypercube. In *Proceedings of RANDOM*, pages 164–178, Springer, Cambridge, 2002. Preprint at arXiv:quant-ph/0103137v1.

[57] G. Merziger, G. Mühlbach, D. Wille, and T. Wirth. *Formeln + Hilfen zur Höheren Mathematik*. Binomi Verlag, Springe, 1996.

[58] T.M. Apostol. *Mathematical Analysis*. Addison-Wesley, Reading, 2nd edition, 1974.

[59] L. K. Grover. A framework for fast quantum mechanical algorithms. In *STOC '98: Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pages 53–62, New York, 1998. ACM.

[60] A. Mizel. Critically damped quantum search. *Phys. Rev. Lett.*, **102**:150501, 2009.

[61] J. D. Joannopoulos, S. G. Johnson, J. N. Winn, and R.D. Meade. *Photonic Crystals*. Princeton University Press, Princeton, Oxford, 2008.

[62] J. M. Harrison, U. Smilansky, and B. Winn. Quantum graphs where back-scattering is prohibited. *J. Phys. A*, **40** :14181, 2007.

[63] C. Cohen-Tannoudji. *Quantum Mechanics*, volume one. John Wiley & Sons, New York, 1977.