Quantum walks: grenzen op de mengtijd en versnelling van klassieke walks

Quantum Walks: Speed Limits on Mixing and Fast-Forwarding Classical Walks

Simon Apers

Promotor: prof. dr. ir. A. Sarlette
Proefschrift ingediend tot het behalen van de graad van
Doctor in de ingenieurswetenschappen: toegepaste natuurkunde

UNIVERSITEIT
GENT

Vakgroep Elektronica en Informatiesystemen
Voorzitter: prof. dr. ir. K. De Bosschere
Faculteit Ingenieurswetenschappen en Architectuur
Academiejaar 2018 - 2019

ISBN 978-94-6355-200-4
NUR 993, 925
Wettelijk depot: D/2019/10.500/8

Supervisor
Alain Sarlette, Ghent University and INRIA Paris-Rocquencourt

Reading Committee
Andris Ambainis, University of Latvia
Stijn De Baerdemacker, Ghent University
Stacey Jeffery, CWI Amsterdam

Examination Board
Tijl De Bie, Ghent University
Gert de Cooman, Ghent University (chair)
Jean-Charles Delvenne, Université catholique de Louvain

(image: jewel designed by the physics enthusiast Louisa-Maria Ponseele
for the marriage of my wife and I)

(pen drawings gratefully taken from the work of Jonas Apers)

Contents

Foreword

The theme that underlies this thesis is the mutualist relation between classical walks and quantum walks on graphs, the latter being a promising algorithmic component of future quantum computers. We study both sides of this coin. On the one side we prove how classical walks can simulate quantum walks. This allows the use of folklore bounds on the behavior of classical walks to study speed limits on quantum walks, a question which had long evaded progress. On the other side, we prove how quantum walks can be used to speed up the behavior of a large class of classical walks. This leads to a new quantum algorithm which we call quantum walk fast-forwarding. We show that this algorithm allows to naturally speed up classical walk algorithms for search and property testing on graphs.

The work is theoretical and mathematical, and supposed to be relevant for quantum computers that should some day exist. I feel proud of this work, and am indebted to many people. Foremost I must thank my supervisor Alain Sarlette, you have been the best supervisor I could wish for, if only for learning me that no-go proofs are traffic diversions rather than dead ends. This work would not have been possible without you. I collaborated on different parts of this thesis with Francesco Ticozzi and Peter Høyer, to both of which I owe my gratitude. I am thankful for discussing with a great number of people, the following of which I must thank in particular: David Feder, Balász Gerencsér, Ronald de Wolf, Jérémie Roland, Michiel Burgelman, Tim Depraetere, Florian Adriaens, Arthur Van Camp, Danial Dervovic, Pieter Claeys, Julien Hendrickx, Dirk Aeyels and Lode Wylleman. I also wish to sincerely thank my parents, being the amazing, loving and supportive people they are, as well as the entire rest of my family and family-in-law, friends and colleagues. My last words, finally, are reserved for my loved and beautiful wife Elisa: this work is dedicated to you and our soon-to-be-born wonder.

Ghent, January 2019 Simon Apers

Synopsis

This thesis studies quantum walks on graphs, the quantum counterpart of random walks and Markov chains on graphs. It is hard to overstate the importance of the latter to fields such as computer science, physics, mathematics, economics and biology. They have been successful both as a way of describing and studying existing phenomena, and as an algorithmic tool to solve computational problems. With the uprise of quantum computing and the so-called "second quantum revolution", quantum walks are announced to serve a similar role, both for describing existing phenomena and for solving computational problems. With the subject gaining momentum less than two decades ago, many interesting questions and research directions remain open or unexplored. The work presented in this thesis studies the mutualist interface and relationship between quantum and classical walks. In very rough strokes, we ask the question of whether classical walks can simulate quantum walks, and conversely, to what extent quantum walks can simulate classical walks. This addresses both sides of a coin, one of which was open, one of which largely unexplored.

In the first part of this thesis we study the simulation of quantum walks using classical walks. A particular interest of the latter is its mixing behavior on graphs, a valued feature that allows to efficiently sample from large state spaces, and is relevant again both from a physical and a computational perspective. The analysis of the mixing behavior of classical walks is a field on its own, with a broad scale of techniques both for designing appropriate walks and for estimating and accelerating their mixing performance. Towards this latter aspect, speeding up the mixing behavior of classical walks, quantum walks have been shown to hold much potential. A paradigmatic example is the cycle graph, on which a quantum walk was shown to provide a quadratic speedup in mixing as compared to a classical walk. This early observation initiated much subsequent research and interest on the topic, leading to both a range of results and a range of open questions on quantum walk mixing. One of these open questions, which will be central to this thesis part, is the exact speedup that quantum walks can achieve over classical walks. Our approach towards solving this question is motivated by another observation on the cycle graph: if we allow the random walk to have an additional memory slot, which is implicitly assumed for the quantum walk, then we can similarly improve the mixing time quadratically, be it in a purely classical manner. The resulting walk is called a lifted Markov chain. Driven by this example, we raise the question of whether

such lifted Markov chains can always achieve the same speedup as quantum walks can. By building on ideas from hidden variable theories in quantum mechanics, we will show that indeed for a broad class of quantum walks there exists a lifted Markov chain that effectively simulates the quantum walk dynamics, and therefore mixing performance. As a critical derivative of this study we are able to prove *speed limits* on the dynamics of a quantum walk. How long does it take a quantum walk to reach a part of the graph? Also, can it do so faster than classical Markov chains? The question of speed limits is well studied for the latter, and has led to the concept of *conductance bounds*, yielding an approach that is both elegant and intuitive. The conductance of a graph quantifies how well-knit or connected it is, and conductance bounds relate this property to the speed at which Markov chains can propagate through the graph. Since we can prove such conductance bounds for lifted Markov chain, we find that they also hold for quantum walks.

In the second part of this thesis we study to what extent quantum walks can simulate classical walks. Crucially, we will be interested in creating a quantum superposition that encodes the classical walk probability distribution, rather than sampling from the classical distribution itself. We will refer to this task as quantum simulation of Markov chains. The creation of quantum superpositions is an important primitive in the field of quantum computing. It underlies quantum algorithms for speeding up search problems and machine learning tasks, as well as for testing properties of the original probability distributions. The problem of quantum simulation was initially tackled in the early work of John Watrous [1], largely initiating the study of quantum walks. He constructed a quantum walk that allows to quantum simulate a random walk over any given number of steps, using an equal number of quantum walks steps, and with success probability inversely proportional to the 2-norm of the random walk distribution. As the main technical contribution of this thesis part we construct a quantum walk algorithm, which we call *quantum walk fast-forwarding* (QFF), that quadratically accelerates this scheme. Specifically, this algorithm builds on quantum walks to quantum simulate the broad class of reversible Markov chains, using quadratically less steps, and with the same success probability as Watrous' scheme. The reversibility condition, while common, is critical here, showing most notably that the lifted Markov chain simulators from the first thesis part cannot be fast-forwarded in this way. Our algorithm builds on the seminal work by Ambainis [2] and Szegedy [3], following up on Watrous' scheme, where they construct quantum walks to speed up specific random walk algorithms for search problems. The direct speedup of Markov chain dynamics, allowed by QFF, leads to a number of new quantum algorithms. Most importantly, we show that QFF allows to straightforwardly accelerate existing random walk algorithms for graph property testing, an important and topical problem in the field of sublinear algorithms. Specifically, we find a quantum speedup with respect to a range of recent algorithms for testing graph expansion and clusterability. Related to this, we discuss how QFF allows to classify nodes into clusters, a relevant problem in the fields of data science and machine learning. Apart from these results, we discuss the use of QFF in the context of graph search problems, showing how QFF allows to escape large sets quadratically faster, and show how QFF can be relevant for quantum state generation.

Structure of the Thesis

In the preliminaries, consisting of Chapters 1 and 2, the most important concepts are introduced. Chapter 1 presents the concept of graphs, random walks and mixing time on graphs, and the random walk extension called lifted Markov chains. Chapter 2 introduces quantum walks, the quantum counterpart of classical walks.

The first thesis part, consisting of Chapters 3 to 6, addresses the problem of bounding the mixing time and simulating quantum walks with lifted Markov chains. Chapter 3 introduces the concept of conductance bounds on Markov chains, Chapter 4 demonstrates how lifted Markov chains allow to simulate quantum walks, and Chapter 5 uses this simulation result to prove conductance bounds on the mixing time of quantum walks. We discuss these results, and present some open question, in Chapter 6.

The second thesis part, consisting of Chapters 7 to 10, presents our new algorithm called quantum walk fast-forwarding, and discusses its applications. In Chapter 7 we discuss the existing approaches to associating a quantum walk to a given classical Markov chain, and in Chapter 8 we present the algorithm for quantum walk fast-forwarding. In Chapter 9 we apply quantum fast-forwarding to the problem of graph property testing, and finally in Chapter 10 we discuss some more applications and present an outlook of remaining questions.

In Appendix A of the back matter, we provide an additional result which has interest of its own, but does not require the techniques presented in the thesis. We provide a quantum walk algorithm which allows to create superpositions over graphs in $\tilde{O}(1/\sqrt{\mathrm{HT}})$ quantum walk steps, with HT the hitting time of the starting node.

Samenvatting (Dutch Synopsis)

Deze thesis bestudeert quantum walks op grafen, de quantum tegenhanger van klassieke walks of Markov ketens op grafen. Het is moeilijk de relevantie van deze laatste te overschatten in gebieden als computerwetenschappen, fysica, wiskunde, economie en biologie. Klassieke walks worden succesvol aangewend zowel voor het beschrijven en bestuderen van bestaande fenomenen, als voor het oplossen van algoritmische problemen. Met de opkomst van de quantumcomputer, en de zogenaamde "tweede quantum revolutie", wordt quantum walks een soortgelijke rol toegeschreven, zowel voor het beschrijven van bestaande fenomenen als voor het oplossen van algoritmische problemen. De studie van quantum walks kwam slechts minder dan twee decennia geleden op, zodat veel interessante vragen en onderzoeksrichtingen vooralsnog open of onverkend zijn. In deze thesis, specifiek, beschrijven we de symbiotische relatie tussen quantum walks en Markov ketens of klassieke walks. In grove lijnen stellen we de vraag of klassieke walks toelaten om quantum walks te simuleren, en omgekeerd, in welke mate quantum walks toelaten om klassieke walks na te bootsen. Zo beschrijven we twee zijden van eenzelfde medaille, één zijde die deels onbegrepen was, en één zijde die deels onverkend was.

In het eerste deel van deze thesis bestuderen we het simuleren van quantum walks met klassieke walks. Een belangrijke eigenschap van deze laatste is hun menggedrag op grafen, een kostbare eigenschap die toelaat om efficiënt grote toestandsruimten te verkennen en bemonsteren, wat opnieuw relevant is van zowel een fysisch als een computationeel standpunt. De analyse van het menggedrag van klassieke walks is een gebied op zichzelf, met een breed spectrum aan technieken zowel voor het ontwerpen van geschikte walks, als voor het schatten en versnellen van de mengprestatie. Dit laatste, het versnellen van het menggedrag van klassieke walks, is een belangrijk aspect waar vermoed wordt dat quantum walks een voordeel kunnen bieden. Een belangrijk voorbeeld hiervan is de cirkel-graaf. Het is welbekend dat quantum walks op deze graaf de mengtijd kwadratisch verbeteren ten opzichte van klassieke walks. Dit was een vroege observatie die aanleiding gaf tot veel opvolgend onderzoek en interesse in het menggedrag van quantum walks, leidend naar zowel een groot aantal resultaten als een groot aantal open vragen. Een van deze open vragen, centraal in dit deel van de thesis, is de precieze versnelling in menggedrag die quantum walks over klassieke walks kunnen

hebben. Onze aanpak om dit probleem op te lossen is gemotiveerd door een andere observatie op de cirkel-graaf: als we een klassieke walk een extra geheugenslot geven, wat impliciet verondersteld wordt voor de quantum walk, dan kunnen we eveneens de mengtijd kwadratisch verbeteren, zij het op een puur klassieke manier. De resulterende klassieke walk, met een extra geheugenslots, wordt een gelifte walk genoemd. Gedreven door dit voorbeeld stellen we de vraag of deze gelifte walks altijd dezelfde versnelling in menggedrag als quantum walks kunnen verwezenlijken. Gebruik makende van ideeën uit verborgen variabelen theorieën, tonen we aan dat voor een brede klasse van quantum walks er inderdaad gelifte walks bestaan die hun dynamica simuleren, en bijgevolg hetzelfde menggedrag zullen hebben. Een opvallend gevolg van dit resultaat is dat dit toelaat om limieten op het menggedrag van quantum walks te bewijzen: Hoe veel tijd heeft een quantum walk nodig om bepaalde delen van een graaf te bereiken? Kan het dit sneller doen dan klassieke walks? Dergelijke vragen zijn uitgebreid bestudeerd voor klassieke walks, leidend tot het concept van *conductantiegrenzen*, een elegante en intuïtieve karakterisatie van de gezochte limieten. De conductantie van een graaf kwantificeert hoe goed een graaf verbonden is, en conductantiegrenzen relateren deze kwantiteit aan de snelheid waarmee een klassieke walk doorheen een graaf kan bewegen. Door het bewijzen van deze conductantiegrenzen voor gelifte walks, die toelaten om quantum walks te simuleren, bewijzen we dat conductantiegrenzen ook kunnen gebruikt worden om het menggedrag van quantum walks te begrenzen.

In het tweede deel van deze thesis bestuderen we in welke mate quantum walks toelaten om klassieke walks te bestuderen. Van cruciaal belang is dat we geïnteresseerd zijn in het creëeren van quantum superposities die de klassieke walk distributie beschrijft, eerder dan het creëren van de klassieke distributie. Deze taak noemen we het *quantum-simuleren* van een klassieke walk. Het creëren van dergelijke quantum superposities is een belangrijke primitieve in het veld van quantumcomputers. Het ligt aan de basis van quantum algoritmes voor taken zoals het versnellen van zoekproblemen, machinaal leren, en het vergelijken van de originele probabiliteitsdistributies. Het quantum-simuleren van klassieke walks werd voor het eerst beschouwd in het vroege werk van John Watrous [1], wat mede aan de basis lag van de opkomst van quantum walks. Hij beschreef een quantum walk die quantum-simulatie toelaat van een klassieke walk over een bepaald aantal stappen, gebruik makende van een gelijk aantal quantum walk stappen, en met een succesprobabiliteit evenredig met de 2-norm van de klassieke walk distributie. Als de belangrijkste technische contributie van dit thesisgedeelte beschrijven we een quantum walk algoritme, dat we *quantum walk fast-forwarding* (QFF) noemen, dat dit scheme kwadratisch versneld. Dit algoritme bouwt op quantum walks om de brede klasse van reversibele klassieke walks te quantum-simuleren, gebruik makende van kwadratisch minder stappen, en met dezelfde succesprobabiliteit als het algoritme van Watrous. De beperking op de reversibiliteit van de klassieke walks is gangbaar maar cruciaal, en toont bijvoorbeeld aan dat het algoritme niet toelaat om de gelifte walk constructies uit het eerste thesisgedeelte te versnellen. Het algoritme vertrekt van het belangrijke werk van Ambainis [4] en Szegedy [3], voortbouwend op het werk van Watrous, waarin ze quantum walks beschrijven die toelaten om bepaalde klassieke walk algoritmes voor zoekproblemen te versnellen.

De directe versnelling van de klassieke walk dynamica, bekomen via ons QFF algoritme, leidt naar een aantal nieuwe quantum algoritmes. Allereerst tonen we aan dat QFF toelaat om op directe wijze bestaande algoritmes voor het testen van graafeigenschappen te versnellen, een belangrijk en actueel probleem in het gebied van sublineaire algoritmes. Specifiek tonen we een quantumversnelling aan over de bestaande algoritmes voor het testen van de conductantie en clusterstructuur van de graaf. Hieraan gerelateerd beschrijven we hoe QFF toelaat om nodes te klassificeren in clusters, een relevant probleem binnen de datawetenschappen en het machinaal leren. Naast deze resultaten beschrijven we het gebruik van QFF voor zoekproblemen op grafen, waarbij we aantonen hoe QFF toelaat om kwadratisch sneller te ontsnappen uit grote verzamelingen in een graaf, en we bespreken de relevantie van QFF voor het creëren van quantum toestanden.

Structuur van de thesis

In de inleiding, bestaande uit Hoofdstukken 1 en 2, worden de belangrijke concepten geïntroduceerd. Hoofdstuk 1 bespreekt grafen, klassieke walks en mengtijden op grafen, en de extensie van klassieke walks genaamd gelifte Markov ketens. Hoofdstuk 2 introduceert quantum walks, de quantum tegenhanger van klassieke walks.

Het eerste gedeelte van de thesis, bestaande uit Hoofdstukken 3 tot 6, behandelt het probleem van het begrenzen van de mengtijd en het simuleren van quantum walks met gelifte Markov ketens. Hoofdstuk 3 introduceert het concept van conductantie grenzen op Markov ketens, Hoofdstuk 4 demonstreert hoe gelifte Markov ketens toestaan om quantum walks te simuleren, en Hoofdstuk 5 gebruikt dit resultaat om snelheidsgrenzen te bewijzen op de mengtijd van quantum walks. We bespreken deze resultaten en enkele open vragen in Hoofdstuk 6.

Het tweede gedeelte van de thesis, bestaande uit Hoofdstukken 7 tot 10, introduceert een nieuw algoritme om quantum walks vooruit te spoelen, en bespreekt toepassingen hiervan. Hoofdstuk 7 bespreekt de bestaande aanpakken om een quantum walk te associëren aan een gegeven Markov keten, en in Hoofdstuk 8 wordt hiervan gebruik gemaakt om het nieuwe algoritme te introduceren. Hoofdstuk 9 past dit algoritme toe op het testen van graafeigenschappen, en in Hoofdstuk 10 worden verdere applicaties en enkele open vragen besproken.

In Appendix A wordt een extra resultaat besproken, dat echter niet direct gebruik maakt van de technieken gepresenteerd in de thesis, maar desalniettemin relevant is. Er wordt een quantum walk algoritme beschreven om een superpositie over de nodes van een graaf te creëren in $\tilde{O}(1/\sqrt{\mathrm{HT}})$ quantum walk stappen, met HT de raaktijd van de beginnode.

Glossary

Asymptotic Notation. We will use asymptotic notation to capture the asymptotic scaling of certain quantities. For f and g functions on the natural numbers these are defined as below.

- $f(n) \in o(g(n))$, or "f is asymptotically dominated by g", if for every $k > 0$ there exists n_0 such that for all $n \geq n_0$ it holds that $f(n) < kg(n)$.
- $f(n) \in O(g(n))$, or "f is upper bounded asymptotically by g", if there exist constants c and n_0 such that for all $n \geq n_0$ it holds that $f(n) \leq cg(n)$.
- $f(n) \in \Theta(g(n))$, or "f is of the same order as g", if $f(n) \in O(g(n))$ and $g(n) \in O(f(n))$.
- $f(n) \in \Omega(g(n))$, or "f is lower bounded asymptotically by g", if $g(n) \in O(f(n))$.
- $f(n) \in \omega(g(n))$, or "f asymptotically dominates g", if $g(n) \in o(f(n))$.
- The tilde-notation $\tilde{o}, \tilde{O}, \tilde{\Theta}, \tilde{\Omega}, \tilde{\omega}$ denotes the same relation as above, up to polynomials in $\log(n)$. E.g., $f(n) \in \tilde{o}(g(n))$ if and only if $f(n) \in o(g(n)\,\mathrm{polylog}(n))$.

Norms. For $v \in \mathbb{C}^N$ we use the following norms:

$\|v\| = \sqrt{\sum \|v(j)\|^2}$	2-norm
$\|v\|_1 = \sum \|v(j)\|$	1-norm
$\|v\|_{\mathrm{TV}} = \frac{1}{2}\|v\|_1$	total variation distance
$\|v\|_\infty = \max \|v(j)\|$	infinity norm

Symbols and Acronyms.

$\otimes$	tensor product
e_i	i-th basis vector
sup	support
T_t	t-th Chebyshev polynomial
	Graphs
G	graph
$G[\mathcal{S}]$	subgraph induced by a subset $\mathcal{S} \subseteq \mathcal{V}$
$\mathcal{V}$	node set
N	number of nodes $N = \lvert\mathcal{V}\rvert$
$\mathcal{E}$	edge set $\subseteq \mathcal{V} \times \mathcal{V}$
$E(\mathcal{S}, \mathcal{T})$	edge set from $\mathcal{S}$ to $\mathcal{T}$
$\mathcal{S}^c$	complement of a subset $\mathcal{S} \subseteq \mathcal{V}$: $\mathcal{S}^c = \mathcal{V} \backslash \mathcal{S}$
$\partial\mathcal{S}$	nodes in $\mathcal{S}$ that have an edge to $\mathcal{S}^c$
$\mathbb{P}(\cdot)$	probability of the event "$\cdot$"
$d(j)$	degree of node j
d_{tot}	total degree of a graph
$\hat{\mathcal{V}}, \hat{\mathcal{E}}, \hat{P}, \ldots$	lifted versions of $\mathcal{V}$, $\mathcal{E}$, P, ...
$\Phi(\mathcal{S})$	conductance of the subset $\mathcal{S} \subseteq \mathcal{V}$
$\Phi(P)$	conductance of the Markov chain P
$\Phi_{G,\pi}$	graph conductance
P_N	N-node path
$\mathbb{Z}_N$	N-node cycle
K_N	N-node complete graph
$K_N - K_N$	dumbbell graph
$T_{2,k}$	complete binary tree of depth k
	Markov chains
MC	Markov chain
P	Markov chain
$P(i, j)$	transition probability from j to i
Q	ergodic flow
$P \sim G$	P is local with respect to graph G
D	discriminant matrix
π	stationary probability distribution
$\tau(\epsilon)$	ϵ-mixing time
τ	mixing time or $(1/4)$-mixing time
δ	spectral gap
$P_{\mathcal{V}}$	Markov chain induced on $\mathcal{V}$
P_s	interpolated Markov chain
π_s	stationary distribution of P_s
HT_j	hitting time from π to j
Γ	stochastic map

$\mathcal{S}^g$	internal nodes of $\mathcal{S}$
	Quantum Walks
$\|\psi\rangle$	ket (quantum state)
$\langle\psi\|$	bra (linear functional)
$\|w\rangle, w \in \mathbb{C}^N$	quantum state described by $w/\|w\|$
$\mathcal{H}$	Hilbert space
QW	quantum walk
tr	trace operator
ρ	density operator
Π	projector
R_ψ	reflector around $\|\psi\rangle$
QFF	quantum fast-forwarding

Preliminaries

This thesis revolves around both classical and quantum dynamics on graphs. In the coming preliminaries part we will introduce elementary concepts such as random walks, Markov chains and quantum walks that describe these dynamics. All of these concepts arise very naturally in a broad range of settings, from physics and computer science to engineering and the social sciences.

In Chapter 1 we define graphs as the abstract structures used to describe distributed network structures and logical state spaces. We introduce random walks and Markov chains, both as a description of stochastic dynamics on these graphs and as an algorithmic means to explore a graph structure. We also introduce one of the central themes of this thesis: mixing on graphs. Under quite general conditions, the dynamics on a graph settle down or converge to some limit behavior, a process which we call mixing. The rate at which this happens is a crucial parameter of the corresponding system or algorithm

In Chapter 2 we introduce the generalizing theory of quantum mechanics, describing the evolution and measurement of quantum states over graphs. We show how quantum walks can describe general quantum dynamics over graphs, and their relation to classical Markov chains. Finally we discuss the mixing of quantum walks and introduce quantum mixing schemes, allowing the comparison of quantum and classical approaches for mixing.

Chapter 1
Graphs and Walks

To a man with a hammer, anything looks like a nail. To a researcher working on graphs and random walks, a similar destiny befalls. Indeed, the broad applicability of these concepts has long and firmly established their central role and ubiquity in fields such as physics, biology, social sciences, and computer science. The coming chapter serves as a primer for graphs and random walks, and more general Markov chains and lifted Markov chains.

1.1 Graphs and Their Vector Space

The stage of this thesis is built from graphs. These are abstract mathematical entities that describe a notion of discrete space. This space can correspond to a physically distributed network or maze, such as a city map. Equally often it corresponds to a logical state space, describing for example the states of a combination lock.

To define a graph G, we start from a finite set of nodes $\mathcal{V}$ representing the state space, and we specify a set of edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. We write $G = (\mathcal{V}, \mathcal{E})$. Throughout the thesis we will restrict our attention to *undirected graphs*, for which $(i, j) \in \mathcal{E}$ if and only if $(j, i) \in \mathcal{E}$. One can think of the nodes as the positions on a map or the numbers on a lock, and the edges as streets on the map or turns of the lock. If $(i, j) \in \mathcal{E}$ then we call the transition from node i to node j allowed, and we call j a *neighbor* of i. We denote this by $j \sim i$. The *degree* $d(i)$ of a node i equals the number of neighbors of i. The *boundary* $\partial\mathcal{S}$ of a subset of the nodes $\mathcal{S} \subset \mathcal{V}$ is defined as all the nodes in $\mathcal{S}$ that have an edge going to $\mathcal{S}^c$: $\partial\mathcal{S} = \{j \in \mathcal{S} : \exists i \in \mathcal{S}^c \text{ s.t. } (j, i) \in \mathcal{E}\}$.

A graph G is *connected* if for all $i, j \in \mathcal{V}$ there exists a path from node i to node j. Here a *path* from i to j is an ordered set of nodes $[i_0, i_1, \ldots, i_r]$, with $(i_l, i_{l+1}) \in \mathcal{E}$ for all $0 \leq l < r$, such that $i_0 = i$ and $i_r = j$. The length of this path is r. The *distance* from i to j equals the minimum length of a path from i to j, and the *diameter* of the graph G is given by the maximum distance between any two nodes. G is called *undirected* if $(i, j) \in \mathcal{E}$ implies that $(j, i) \in \mathcal{E}$. In this thesis we will only consider connected and undirected graphs.

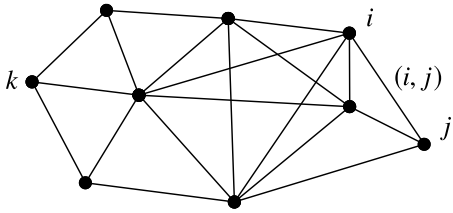


Fig. 1.1 Connected, undirected graph with diameter 3, attained between for instance nodes k and j.

We will associate an inner product space $\mathbb{R}^{\mathcal{V}}$ to G, isomorphic to $\mathbb{R}^{|\mathcal{V}|}$, by identifying with each node i a basis vector e_i, so that

$$\mathbb{R}^{\mathcal{V}} = \operatorname{span}_{\mathbb{R}}\{e_i \mid i \in \mathcal{V}\}.$$

For $v = \sum v(i)e_i$ and $w = \sum w(i)e_i$, the standard inner product is defined as

$$\langle v, w\rangle = v^{\dagger}w = \sum_{i\in\mathcal{V}} v(i)w(i),$$

where $v^{\dagger}$ denotes the dual vector of w. This inner product naturally induces the norm $\|v\| = \sqrt{\langle v, v\rangle}$.

1.2 Random Walks and Markov Chains

As one of the main characters acting on these graphs, we can think about an agent or walker that moves from node to node. Similarly we can think about a configuration that is changing, like when randomly trying the combinations of a lock. This results in a *stochastic process* over the state space $\mathcal{V}$, which is described by a chain of random variables

$$\{X_t\}_{t\in\mathbb{N}} = \{X_0, X_1, X_2, \dots\},$$

where the subscript represents a time index. We can describe the probability distribution of the random variable $X_t \in \mathcal{V}$ by a probability vector $v_t \in \mathbb{R}^{\mathcal{V}}$ defined by

$$v_t(j) = \mathbb{P}(X_t = j),$$

where $\mathbb{P}(Y)$ represents the probability of event Y. A *random walk* is a stochastic process described by the following rule:

If current state is i, pick a uniformly random neighbor $j \sim i$ and move there.

This generates a chain $\{X_t\}$ where

$$\begin{aligned}&\mathbb{P}(X_{t+1} = j \mid (X_t, \dots, X_0) = (i_t, \dots, i_0))\\ &\quad = \mathbb{P}(X_{t+1} = j \mid X_t = i_t) = \begin{cases} 1/d(i_t) & \text{if } j \sim i_t\\ 0 & \text{otherwise,}\end{cases}\end{aligned}$$

where $\mathbb{P}(Y|Z)$ denotes the probability of event Y conditioned on event Z. The state of a random walk X_t at time t is distributed according to the probability vector v_t. Given v_t, we can calculate v_{t+1} by using the *transition matrix* P of the random walk. This matrix is defined by the random walk transition rule, setting

$$P(j,i) = \mathbb{P}(X_{t+1} = j \mid X_t = i),$$

which is independent of t. A transition matrix is necessarily stochastic, i.e., it is elementwise nonnegative and every column sums to 1. We can now easily calculate v_{t+1} from v_t by noting that

$$v_{t+1} = Pv_t,$$

for all $t \geq 0$. As a direct consequence it holds that $v_t = P^t v_0$, and so the probability of being in node j after t steps, starting from node i, is given by

$$\mathbb{P}(X_t = j \mid X_0 = i) = P^t(j,i).$$

Example 1 (Random Walk on $\mathbb{Z}_N$). As an example, consider a random walk on the N-cycle graph $G = (\mathcal{V}, \mathcal{E})$, so that

$$\mathcal{V} = \{0, 1, \ldots, N-1\} \text{ and } \mathcal{E} = \{(i, i \pm 1), i \in \mathbb{Z}_N\},$$

where all additions in this example are modulo N. If $X_t = i$, so the random walk at time t is at position i, then it hops left or right with equal probability:

$$\mathbb{P}(X_{t+1} = i \pm 1 \mid X_t = i) = \frac{1}{2} = P(i \pm 1, i),$$

and P is zero elsewhere. This defines the transition matrix, illustrated on the left in Figure 1.2. Alternatively, let $P^{\uparrow}$ and $P^{\downarrow}$ correspond to the clockwise, respectively, anti-clockwise shift operators, so that $P^{\uparrow}e_i = e_{i+1}$ and $P^{\downarrow}e_i = e_{i-1}$. Then $P = \frac{P^{\uparrow}+P^{\downarrow}}{2}$. This random walk is a model for classical diffusion dynamics. If $N \gg t \gg 1$ then the distribution v_t of X_t will approximate a binomial distribution, centered around the origin with standard deviation in $\Theta(\sqrt{t})$. On the right in Figure 1.2 we plot the probability distribution $v_t = P^t e_0$ of the random walk after $t = 100$ steps, starting from the origin. △

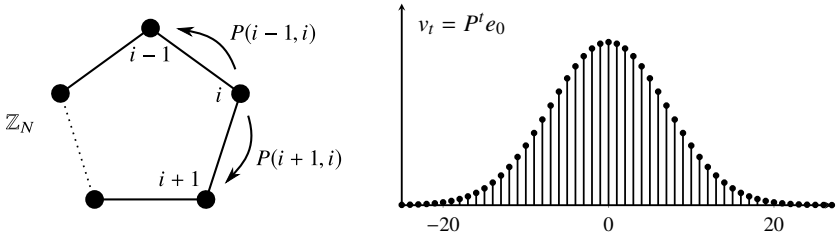


Fig. 1.2 (left) Random walk on the cycle $\mathbb{Z}_N$. (right) Plot showing the distribution $v_t = P^t e_0$ of X_t, the random walk starting at the origin, for $t = 100$. The distribution approaches a Gaussian with standard deviation $\Theta(\sqrt{t})$.

Example 2 (Random Walk on $\mathbb{Z}_2^N$). As another example, we consider a random walk on the hypercube $\mathbb{Z}_2^N$. In this case we can describe

$$\mathcal{V} = \{+1, -1\}^N \quad \text{and} \quad \mathcal{E} = \{(v, v') : v, v' \in \mathcal{V}, \|v - v'\|_1 = 2\},$$

so that an edge exists between v and v' if and only if they differ in exactly one of the N coordinates. As a consequence, any node v has exactly N neighbors, each of which is described by flipping a single coordinate. A random walk on $\mathbb{Z}_2^N$ will then uniformly at random flip a single coordinate.

A realistic situation where such a graph arises is the one-dimensional Ising model, consisting of N spins or magnets on a line, each of which is either oriented up $(+1)$ or down (-1). A single configuration then corresponds to an element in $\mathcal{V} = \{+1, -1\}^N$. A random walk on this configuration graph describes a model where at each time step a single uniformly random spin is flipped. △

By the definition of a random walk, we also have that

$$\begin{aligned}\mathbb{P}(X_{t+1} = j \mid \{X_0, \ldots, X_t\} = \{i_0, \ldots, i_t\}) \\ = \mathbb{P}(X_{t+1} = j \mid X_t = i_t) = P(j, i_t),\end{aligned}$$

for some stochastic matrix P. That is, the transition at time t only depends on the current state X_t. A *Markov chain* is defined as a more general stochastic process that obeys this condition, yet the precise transition probabilities $P(j, i)$ can be more complicated then for a random walk. As a consequence, a Markov chain $\{X_t\}_{t \in \mathbb{N}}$ is fully characterized by a transition matrix P, and an initial state $X_0 = i_0$ or a distribution v_0 of X_0. Often we will omit a description of the initial state, simply referring to the transition matrix P as a Markov chain.

A general Markov chain associated to a graph G should not be restricted to the simple random walk transition rule, hopping to a uniformly random neighbor at any time. It is common that transitions to certain neighbors are more attractive or more likely than others. The transition probabilities in P can in principle be chosen freely, up to the fact that P has to be a stochastic matrix, and that certain transition are forbidden:

$$P(j, i) = 0 \quad \text{if} \quad (i, j) \notin \mathcal{E}.$$

If P is stochastic and it obeys this condition then we write $P \sim G$. The following properties will be relevant:

- A Markov chain is *irreducible* if

  $$\forall i, j \in \mathcal{V}, \exists t \geq 0 : \quad P^t(j, i) > 0,$$

  so that there is a nonzero probability of reaching node j after t steps, starting from node i.
- A Markov chain is *aperiodic* if

  $$\forall i \in \mathcal{V} : \quad \gcd\{m > 0 \mid P^m(i, i) > 0\} = 1,$$

  where gcd denotes the greatest common divisor.
- A Markov chain is *ergodic* if it is irreducible and aperiodic.

As an example, the random walk on the cycle $\mathbb{Z}_N$ is always irreducible, yet it is aperiodic if and only if N is odd. A distribution π is a *stationary distribution* of P if $P\pi = \pi$. As a consequence, if the initial state is distributed according to π, so $v_0 = \pi$, then X_t is also distributed according to π:

$$v_t = P^t \pi = \pi.$$

Given a graph G, it follows by inspection that the stationary distribution π of the random walk is defined by

$$\pi(j) = \frac{d(j)}{d_{\text{tot}}}, \quad \text{where} \quad d_{\text{tot}} = \sum_{j \in \mathcal{V}} d(j). \tag{1.1}$$

The following theorem is called the *fundamental theorem of Markov chains*, a proof of which can be found in for instance [5].

Theorem 1 (Fundamental Theorem). *If a Markov chain P is irreducible then it has a unique stationary distribution π, and $\pi > 0$ elementwise. If P is ergodic then for any distribution v it holds that* $\lim_{t\to\infty} P^t v = \pi$.

The following corollary will prove useful.

Corollary 1. *(i) The eigenvalues of a Markov chain P lie in or on the unit circle. (ii) An ergodic Markov chain P has a unique eigenvalue λ with $|\lambda| = 1$.*

Proof. (i) With any eigenvalue λ corresponds a left eigenvector f, such that $fP = \lambda f$. Since P is stochastic, fP represents a stochastic combination of the elements of f, and thus $\max |(fP)(j)| \leq \max |f(j)|$. Since $fP = \lambda f$, this implies that $\max |(fP)(j)| = |\lambda| \max |f(j)| \leq \max |f(j)|$, and hence $|\lambda| \leq 1$.
(ii) Follows from the fundamental theorem, with the corresponding eigenvector being the unique stationary distribution. □

Markov chains arising in practice are often *reversible*. This means that the "time-reversed" dynamics in the stationary regime are equal to the original dynamics. For an irreducible Markov chain P, starting from its stationary distribution $v_0 = \pi$, this means that any path $\{i_0, i_1, \ldots, i_t\}$ has the same probability as its time-reversed path $\{i_t, i_{t-1}, \ldots, i_0\}$:

$$\begin{aligned} &\mathbb{P}(\{X_0, X_1, \ldots, X_t\} = \{i_0, i_1, \ldots, i_t\} \mid v_0 = \pi) \\ &= \mathbb{P}(\{X_0, X_1, \ldots, X_t\} = \{i_t, i_{t-1}, \ldots, i_0\} \mid v_0 = \pi). \end{aligned} \tag{1.2}$$

By direct inspection, an irreducible Markov chain P is reversible if and only if a condition called *detailed balance* holds, stating that the probability flow from any node i to any node j in the stationary regime equals the probability flow from node j to node i. We call this stationary flow from i to j the *ergodic flows* $Q(i, j)$, defined as $Q(i, j) = P(j, i)\pi(i)$. The detailed balance condition hence becomes

$$Q(i, j) = Q(j, i).$$

If P is symmetric, then π is the uniform distribution, and P is always reversible.

A nice consequence of reversibility is that P has real eigenvalues. To see this, we define the symmetric *discriminant matrix*

$$D = \sqrt{P \circ P^T}, \tag{1.3}$$

where "$\sqrt{}$" and "$\circ$" denote the elementwise square root and product, respectively. If P is irreducible and reversible, then

$$D(i,j) = \sqrt{P(i,j)P(j,i)} = \sqrt{\pi(i)^{-1}}P(i,j)\sqrt{\pi(j)}.$$

Therefore $D = \operatorname{diag}(\sqrt{\pi})^{-1} P \operatorname{diag}(\sqrt{\pi})$, so that the matrices P and D are similar and thus share the same eigenvalues. Since D is symmetric, these eigenvalues are necessarily real, and by the above corollary they lie in the interval $[-1, 1]$.

Example 3 (Metropolis Algorithm). An interesting demonstration and motivation of the above concepts is the Metropolis algorithm, originally described in 1953 in [6]. Here we are given a graph $G = (\mathcal{V}, \mathcal{E})$ and we wish to sample an element of $\mathcal{V}$ according to some probability distribution π. The Metropolis algorithm allows to do so in situations where we cannot explicitly calculate π, yet we have access to the *relative probabilities* $\pi(v)/\pi(v')$ for $(v, v') \in \mathcal{E}$.

A typical situation where this arises is in statistical physics, where we wish to calculate the expected value of an observable over some large statistical ensemble such as a Gibbs state. As a concrete illustration we retake the one-dimensional Ising model, introduced in Example 2, where $\mathcal{V} = \{+1, -1\}^N$ represents the set of configurations of a system of N spins. The Ising model associates to any configuration $v \in \mathcal{V}$ an energy $E(v) = -\sum_{i=1}^{N-1} v_i v_{i+1}$, where v_i denotes the spin orientation of the i-th spin. This energy is minimized when all spins are equally oriented $+1$ or -1, in analogy to a line of magnets which are all oriented to the same side. The Gibbs distribution π over this ensemble is defined by

$$\pi(v) = \frac{e^{-E(v)}}{\sum_v e^{-E(v)}},$$

associating a higher weight to states with a lower energy. Now assume that we wish to calculate the expected energy $\mathbb{E}_\pi(E) = \sum \pi(v)E(v)$ over this ensemble. With respect to N, which we typically want to choose large, this sum has exponentially many terms, making it computationally infeasible to exactly evaluate it. A classic approach is then to use a Monte Carlo approach for approximating the expectation, randomly sampling elements from π and using these samples to construct an estimate of $\mathbb{E}_\pi(E)$. Standard methods for sampling from π, such as inverse transform sampling or rejection sampling, require that we can explicitly calculate $\pi(v)$ for any v. However, from the form of $\pi(v)$ we see that this again requires to evaluate an exponentially large sum, being $\sum_v e^{-E(v)}$ in this case (this is called the *partition function*). The enterprise is saved by noting that one thing that we can do, is to evaluate the relative probabilities $\pi(v)/\pi(v') = e^{-E(v)}/e^{-E(v')}$, which only require to calculate the sum expressions for $E(v)$ and $E(v')$, which is linear in N. The Metropolis algorithm shows that it is effectively possible to sample from π using only these relative probabilities.

The algorithm builds on a Markov chain that implements the following rule on a general graph $G = (\mathcal{V}, \mathcal{E})$ (which in the case of the Ising model would typically be the hypercube $\mathbb{Z}_2^N$):

If the current state is i, pick a random neighbor $j \sim i$.
With probability $\min\left(1, \frac{\pi(j)d(i)}{\pi(i)d(j)}\right)$ *move to j, otherwise stay at i.*

This transition rule corresponds to a Markov chain P, reminiscent of a random walk, defined by

$$P(j,i) = \begin{cases} \frac{1}{d(i)} \min\left(1, \frac{\pi(j)d(i)}{\pi(i)d(j)}\right) & \text{if } j \sim i, \\ 1 - \sum_{j \sim i} \frac{1}{d(i)} \min\left(1, \frac{\pi(j)d(i)}{\pi(i)d(j)}\right) & \text{if } j = i, \\ 0 & \text{otherwise.} \end{cases}$$

The Markov chain moves to a random neighbor ("accepting" the move) with a probability $\min\left(1, \frac{\pi(j)d(i)}{\pi(i)d(j)}\right)$, otherwise staying at the original node ("rejecting" the move). Hence the step is always accepted if $\pi(j)d(i) \geq \pi(i)d(j)$, yet if $\pi(j)d(i) < \pi(i)d(j)$ it is rejected with probability $\pi(j)d(i)/\pi(i)d(j)$. On a regular graph, we see that this Markov chain decreases the probability of moving to a neighbor with lower weight, and the chain will have a higher probability of being in states with a higher weight.

By direct inspection we see that $P\pi = \pi$. If in addition $\pi > 0$ and the original random walk is ergodic, then P will be ergodic so that

$$\lim_{t \to \infty} P^t v = \pi.$$

We see that, starting from an arbitrary state or distribution, the Markov chain converges to our goal distribution π, eventually returning a sample distributed according to π. This algorithm is a special case of an approach called *Markov Chain Monte Carlo* (MCMC), where Markov chains are used as a way to sample elements.

Another interesting application of the Metropolis algorithm is when we wish to sample a uniformly random element from the nodes of a graph. Running a random walk until it reaches its stationary distribution (1.1) is a straightforward strategy, yet in general this stationary distribution will not be uniform. As a remedy we can use the Metropolis algorithm, setting π equal to the uniform distribution, to adapt the random walk into a Markov chain whose limiting distribution is uniform. △

1.3 Mixing Time

From the fundamental theorem of Markov chains we know that an ergodic Markov chain P will converge to some stationary distribution π. We call this *mixing* of the Markov chain. Mixing is central to MCMC algorithms, where we wish to sample from the limit distribution π. The *mixing time* quantifies the time it takes the Markov chain to get close to π, which is a critical aspect of MCMC algorithms.

Formally, consider an ergodic Markov chain P over a graph $G = (\mathcal{V}, \mathcal{E})$ with stationary distribution π. Then the *ϵ-mixing time* $\tau(\epsilon)$ of P is defined as the smallest time step such that the Markov chain remains ϵ-close to π, starting from the worst initial state:

$$\tau(\epsilon) = \min\{T \mid \max_{v} \|P^t v - \pi\|_{\text{TV}} \leq \epsilon, \forall t \geq T\}.$$

Here the maximization runs over all distributions over $\mathcal{V}$ and $\|\cdot\|_{\text{TV}}$ denotes the total variation (TV) distance, defined as half of the 1-norm $\|\cdot\|_{\text{TV}} = \frac{1}{2}\|\cdot\|_1$. It is

natural to use the TV distance as a measure of convergence between probability distribution, since this norm has an operational interpretation: if p and q are probability distributions, then

$$\|p - q\|_{\mathrm{TV}} = \max_{\mathcal{S} \subset \mathcal{V}} |p(\mathcal{S}) - q(\mathcal{S})|.$$

In words, $\|p - q\|_{\mathrm{TV}}$ equals the maximum difference between the probability associated to a subset or event $\mathcal{S}$ by p and the probability associated to $\mathcal{S}$ by q. We note that the maximization in the definition of $\tau(\epsilon)$ can in fact be restricted to initial states of the form $v = e_j$. This can easily be seen from the following inequality:

$$\begin{aligned}\|P^t v - \pi\|_{\mathrm{TV}} &= \Big\| \sum_j v(j) \left(P^t e_j - \pi\right) \Big\|_{\mathrm{TV}} \\ &\leq \sum_j v(j) \|P^t e_j - \pi\|_{\mathrm{TV}} \leq \max_j \|P^t e_j - \pi\|_{\mathrm{TV}},\end{aligned}$$

where the first inequality follows from applying the triangle inequality on the TV distance. This implies that $\max_v \|P^t v - \pi\|_{\mathrm{TV}} = \max_j \|P^t e_j - \pi\|_{\mathrm{TV}}$.

We are generally interested in the mixing time for asymptotically small ϵ, yet it is not a priori clear that we can find good estimates on $\tau(\epsilon)$ for all ϵ. The following lemma shows that for this purpose it suffices to have an upper bound on $\tau(\epsilon_0)$ for some fixed $\epsilon_0 < 1/2$. It is often called the *amplification lemma* because it shows how iterating $P^{\tau(\epsilon_0)}$ amplifies the closeness to π.

Lemma 1 (Amplification lemma). *Consider an ergodic Markov chain P that mixes to π. For any $\epsilon_0 < 1/2$ it holds that*

$$\tau(\epsilon) \leq \tau(\epsilon_0) \cdot \left\lceil \frac{\log \frac{1}{\epsilon}}{\log \frac{1}{2\epsilon_0}} \right\rceil, \quad \forall \epsilon > 0.$$

Proof. Let $T = \tau(\epsilon_0)$. First we will assume that $t = kT$, $k \in \mathbb{N}$. We can bound

$$\max_v \|P^{kT} v - \pi\|_{\mathrm{TV}} \leq \max_{v,w} \|P^{kT} v - P^{kT} w\|_{\mathrm{TV}}.$$

This is an easy consequence of the fact that w is maximized over a set containing π, and $P^{kT}\pi = \pi$. We can now use a property called *submultiplicativity* of the total variation distance under application of a stochastic matrix A, see Lemma 2. Setting $A = P^T$ this implies that

$$\max_{v,w} \|P^{kT} v - P^{kT} w\|_{\mathrm{TV}} \leq \big(\max_{v,w} \|P^T v - P^T w\|_{\mathrm{TV}} \big)^k.$$

From the triangle inequality we know that

$$\max_{v,w} \|P^T v - P^T w\|_{\mathrm{TV}} \leq 2 \max_v \|P^T v - \pi\|_{\mathrm{TV}} \leq 2\epsilon_0,$$

since we defined $T = \tau(\epsilon_0)$. As a consequence we find that

$$\max_v \|P^{kT} v - \pi\|_{\mathrm{TV}} \leq \max_{v,w} \|P^{kT} v - P^{kT} w\|_{\mathrm{TV}} \leq (2\epsilon_0)^k \ \leq \epsilon,$$

if $k \geq \log(1/\epsilon)/\log(1/(2\epsilon_0))$. This proves the statement for $t = kT$.

Now if $t = t' + kT$ for some $t' \in (0, T)$, we use the fact that the TV distance is nonincreasing under multiplication with a stochastic matrix: $\|Pw\|_{\mathrm{TV}} \leq \|w\|_{\mathrm{TV}}$ for any vector w. To see this, let $w = w_{>0} + w_{<0}$, where $w_{>0}$ and $w_{<0}$ hold the positive and negative part respectively of w. Then $\|w\|_{\mathrm{TV}} = \|w_{>0}\|_{\mathrm{TV}} + \|w_{<0}\|_{\mathrm{TV}}$, and so

$$\begin{aligned}\|Pw\|_{\mathrm{TV}} &\leq \|Pw_{>0}\|_{\mathrm{TV}} + \|Pw_{<0}\|_{\mathrm{TV}} \\ &= \|w_{>0}\|_{\mathrm{TV}} + \|w_{<0}\|_{\mathrm{TV}} = \|w\|_{\mathrm{TV}},\end{aligned}$$

where we used the fact that $\|Pw_{>0}\|_{\mathrm{TV}} = \|w_{>0}\|_{\mathrm{TV}}$, and similarly for $w_{<0}$, which is a consequence of the stochasticity of P.

From this property of the TV distance we see that for all v it holds that

$$\begin{aligned}\|P^t v - \pi\|_{\mathrm{TV}} &= \|P^{t'} P^{kT} v - \pi\|_{\mathrm{TV}} \\ &= \|P^{t'}(P^{kT} v - \pi)\|_{\mathrm{TV}} \leq \|P^{kT} v - \pi\|_{\mathrm{TV}},\end{aligned}$$

where we also used the fact that $P^{t'}\pi = \pi$. So finally we find that, for arbitrary $t \geq 0$,

$$\max_v \|P^t[v] - \pi\|_{\mathrm{TV}} \leq \max_v \|P^{\lfloor t/T \rfloor \cdot T}[v] - \pi\|_{\mathrm{TV}} \leq \epsilon,$$

if $\lfloor t/T \rfloor \geq \log(1/\epsilon)/\log(1/(2\epsilon_0))$, or equivalently

$$\begin{aligned}t &\geq T \cdot \lceil \log(1/\epsilon)/\log(1/(2\epsilon_0)) \rceil \\ &= \tau(\epsilon_0) \cdot \lceil \log(1/\epsilon)/\log(1/(2\epsilon_0)) \rceil.\end{aligned}$$

□

Lemma 2. *Let A be a stochastic matrix and define $\overline{d}_A(t) = \max_{v,w} \|A^t v - A^t w\|_{\mathrm{TV}}$, where the maximization is over probability vectors v and w. Then $\overline{d}_A(t)$ is submultiplicative:*

$$\overline{d}_A(t+s) \leq \overline{d}_A(t)\overline{d}_A(s).$$

In particular, $\overline{d}_A(t) \leq \overline{d}_A(1)^t$.

Proof. See [5, Lemma 4.12]. □

Setting $\epsilon_0 = 1/4$, the amplification lemma implies that

$$\tau(\epsilon) \leq \tau(1/4)\lceil \log(1/\epsilon) \rceil. \tag{1.4}$$

Asymptotically it also holds that

$$\tau(\epsilon) \in \Omega\left(\frac{\tau(1/4)}{\log(1/\min \pi(j))} \log(1/\epsilon)\right),$$

which can be calculated from the below Proposition 1. As a consequence, the bound (1.4) is asymptotically tight up to a log factor[1]. This proves that $\tau(1/4)$ is a good quantifier for the ϵ-mixing time $\tau(\epsilon)$. In this thesis, and in line with the existing mixing literature, we rely on this close correspondence to denote by $\tau(1/4)$ the canonical *mixing time* τ of the graph:

$$\tau \equiv \tau(1/4).$$

[1] So far we have not been able to come up with an example where this extra log factor is necessary.

We mentioned that reversible Markov chains have a well-behaved spectrum. Indeed, for these Markov chains we can use the spectrum to bound $\tau(\epsilon)$. Thereto define the *spectral gap* δ of a reversible, ergodic Markov chain P as the minimum distance between the unique eigenvalue 1 and any other eigenvalue:

$$\delta = \min_{\lambda \neq 1}(1 - |\lambda|),$$

where the minimization is over all eigenvalues of P different from 1. The following proposition is standard, and a proof can be found in [5].

Proposition 1. *If P is a reversible, ergodic Markov chain with stationary distribution π, then*

$$\left(\frac{1}{\delta} - 1\right) \ln\left(\frac{1}{2\epsilon}\right) \leq \tau(\epsilon) \leq \frac{1}{\delta} \ln\left(\frac{1}{\epsilon \min_j \pi(j)}\right).$$

We can use this proposition to estimate the random walk mixing time on $\mathbb{Z}_N$.

Example 4 (Mixing time on $\mathbb{Z}_N$). We previously introduced the random walk on the N-cycle graph, having a characteristic spreading of $\Theta(\sqrt{t})$ in t steps. As a consequence, it will take N^2 steps to spread out over the cycle, and one expects a mixing time in $\Theta(N^2)$. To prove this, note that the transition matrix $P = \frac{P^{\uparrow}+P^{\downarrow}}{2}$ is translationally invariant ($P^{\uparrow}PP^{\downarrow} = P$), and is hence diagonal in the Fourier basis

$$\{w_k\}_{0 \leq k < N}, \quad \text{where } w_k(j) = e^{i\frac{2\pi k}{N}j}.$$

Indeed, it is straightforward to verify that $Pw_k = \cos\left(\frac{2\pi k}{N}\right) w_k = \lambda_k w_k$.

If N is even, then the random walk is not ergodic since it is periodic, and it will not mix. Indeed, in this case P has an eigenvalue $\lambda_{N/2} = -1$ and so the spectral gap $\delta = 0$. If N is odd, then $\lambda_{\lceil N/2 \rceil} = \cos\left(\pi + \frac{\pi}{N}\right)$ is the eigenvalue with the largest modulus different from 1. The spectral gap is therefore given by

$$\delta = 1 - \left|\cos\left(\pi + \frac{\pi}{N}\right)\right| = \frac{\pi^2}{2N^2} + O\left(\frac{1}{N^4}\right).$$

By Proposition 1 this predicts a lower bound on the mixing time $\tau \in \Omega(N^2)$. The upper bound gives $\tau \in O(N^2 \log N)$. More advanced methods (see again [5]) allow to lose this extra log-factor, finally yielding the mixing time $\tau \in \Theta(N^2)$, as we anticipated. △

1.4 Lifted Markov Chains

The random walk on $\mathbb{Z}_N$ has a mixing time in $\Theta(N^2)$, which is unsatisfactory since it is quadratically larger than the diameter $\lceil N/2 \rceil$. Gerencsér [7] showed that in fact any Markov chain on $\mathbb{Z}_N$ that mixes to the uniform distribution has a mixing time in $\Omega(N^2)$. Motivated by this, Diaconis, Holmes and Neal [8] proposed to lift the state space $\mathcal{V} = \mathbb{Z}_N$, adding auxiliary states to the walker, with the goal of accelerating its mixing time. The resulting lifted Markov chains were introduced more formally by Chen, Lovász and Pak in [9].

Consider an ergodic Markov chain P over some graph $G = (\mathcal{V}, \mathcal{E})$ with stationary distribution π. A graph $\hat{G} = (\hat{\mathcal{V}}, \hat{\mathcal{E}})$, together with a many-to-one function $f : \hat{\mathcal{V}} \to \mathcal{V}$, is called a *lifting* of G if and only if

$$(f(i), f(j)) \notin \mathcal{E} \Rightarrow (i, j) \notin \hat{\mathcal{E}}.$$

So the lifting associates a set of "child nodes" $f^{-1}(i) \subset \hat{\mathcal{V}}$ to a "parent node" $i \in \mathcal{V}$, and there is an edge between child nodes only if there is an edge between the parent nodes. We define an *initialization map* as a one-to-one function $F : \mathcal{V} \to \hat{\mathcal{V}}$ for which $(f \circ F)(i) = i$, i.e., it associates a unique child node with every parent node. A general *lifted Markov chain* (LMC) on a lifted graph $\hat{G}$ is then described by a Markov chain $\hat{P}$ over $\hat{G}$, together with an initialization map F. We are generally interested in the *marginal distribution* of $\hat{P}^t \circ F$ on the original node set $\mathcal{V}$, where now "$\circ$" denotes composition. Starting from some distribution v over $\mathcal{V}$, this marginal is naturally defined by

$$(f \circ \hat{P}^t \circ F)[v]. \tag{1.5}$$

Here we intuitively extended f and F to maps between probability distributions: $f[v]$ for $v \in \mathbb{R}^{\hat{\mathcal{V}}}$ is defined by $f[v](j) = v(f^{-1}(j))$, and $F[v]$ for $v \in \mathbb{R}^{\mathcal{V}}$ is defined by $F[v](j) = v(i)$ if $j = F(i)$ for some $i \in \mathcal{V}$ and zero elsewhere. We call the combined scheme $(f \circ \hat{P}^t \circ F)$ a *classical mixing scheme*, mapping an"input" probability distribution over $\mathcal{V}$ to an "output" distribution over $\mathcal{V}$, and we say that a scheme mixes to π if and only if for all distributions v over $\mathcal{V}$ it holds that

$$\lim_{t \to \infty} (f \circ \hat{P}^t \circ F)[v] = \pi.$$

The above condition thus demands that the lifted Markov chain is such that, after initialization F, its marginal converges to π. Note that an LMC can mix to a unique stationary distribution even when the lifted transition matrix $\hat{P}$ is not ergodic, contrary to the case for simple Markov chains. The ϵ-mixing time $\tau(\epsilon)$ of an LMC is defined similarly to simple Markov chains:

$$\tau(\epsilon) = \min\{T \mid \max_{v} \|(f \circ \hat{P}^t \circ F)[v] - \pi\|_{\mathrm{TV}} \leq \epsilon, \forall t \geq T\}.$$

We now discuss the original LMC construction on the cycle, described in [8].

Example 5 (LMC on $\mathbb{Z}_N$). In [8], an LMC was proposed with the aim of speeding up the unsatisfactory random walk mixing time on the cycle. They propose the following transition rule, for some $1 \geq \alpha \geq 0$ (again all operations are modulo N):

> *If current state is i, and former state $i-1$, "go forward" to $i+1$ with probability $1-\alpha$ and otherwise "go back" to $i-1$.*
> *If current state is i, and former state $i+1$, "go forward" to $i-1$ with probability $1-\alpha$ and otherwise "go back" to $i+1$.*

For $\alpha = 0$ this is the trivial non-backtracking random walk on the cycle, which moves fast around the cycle, yet it is N-periodic and hence never mixes to its stationary distribution. For $1 \gg \alpha > 0$, the Markov chain keeps moving in the same direction with a high probability, moving fast around the cycle, yet it stays

ergodic. This transition rule can be implemented using a simple LMC. Thereto define the lifted graph $\hat{G} = (\hat{\mathcal{V}}, \hat{\mathcal{E}})$ by

$$\hat{\mathcal{V}} = C \times \mathcal{V} = \{\uparrow, \downarrow\} \times \{0, 1, \ldots, N-1\},$$

with "$\times$" the Cartesian product, and the marginalization function by $f(\uparrow, i) = f(\downarrow, i) = i$ for all $i \in \mathcal{V}$. Furthermore, $\hat{\mathcal{E}}$ contains all edges between $C \times \{i\}$ and $C \times \{i \pm 1\}$ for all $i \in \mathcal{V}$. So apart from a position, the walker now also has a coin with sides $\uparrow$ and $\downarrow$. The following LMC $\hat{P}$ on $\hat{G}$ implements their strategy:

$$\hat{P} = S \cdot (C \otimes I), \quad C = \begin{bmatrix} 1-\alpha & \alpha \\ \alpha & 1-\alpha \end{bmatrix}, \tag{1.6}$$

where "$\otimes$" represents the tensor product and $S = e_\uparrow e_\uparrow^\dagger \otimes P^\uparrow + e_\downarrow e_\downarrow^\dagger \otimes P^\downarrow$ is the conditional shift operator. Intuitively, this LMC implements the following two steps:

1. *coin toss* $C \otimes I$: keep the walkers' coin $\uparrow$ or $\downarrow$ with probability $1 - \alpha$, and otherwise flips it,
2. *shift* S: if the coin state is $\uparrow$, move to the right, if it is $\downarrow$, move to the left.

The initialization map F can be chosen arbitrarily, e.g., $F(i) = (\uparrow, i)$. It is easy to check that $\hat{P}$ is ergodic, and mixes to the uniform distribution over $\hat{\mathcal{V}}$. After marginalization this corresponds to the uniform distribution π over $\mathcal{V}$, so that indeed the LMC mixes to π. In [8] they continue to show that if $\alpha = 1/N$, so the LMC keeps moving in the same direction for an expected number of steps N, then $\hat{P}$ mixes in $\Theta(N)$ steps. As a consequence, the LMC quadratically accelerates the mixing time of any simple Markov chain.

In more recent work by Diaconis and Miclo [10] this construction is generalized to accelerate random walks on Cayley graphs of Abelian groups. They conjecture that this leads to a similar quadratic speedup. △

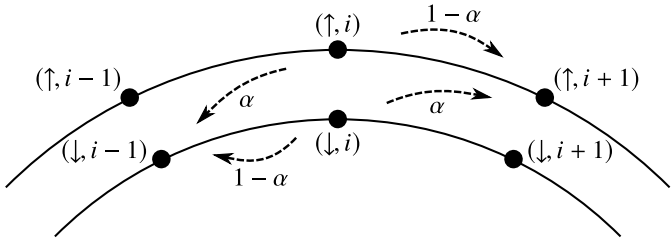


Fig. 1.3 Lifted Markov chain on $\mathbb{Z}_N$ as proposed in [8], with $\alpha = 1/N$. The LMC has a large probability $1 - \alpha$ of continuing in the same direction, e.g., $(\uparrow, i) \to (\uparrow, i+1) \to (\uparrow, i+2) \to \ldots$, at every step switching only with a small probability α. This scheme remedies the diffusive behavior of the random walk on $\mathbb{Z}_N$, mixing to the uniform distribution in $O(N)$ steps.

Chen, Lovász and Pak [9] furthermore showed that the LMC speedup on the cycle is not an isolated case, proving the below theorem. The statement involves the quantity $\Phi(P)$, called the conductance of the Markov chain P, which we will introduce later. For the cycle $\mathbb{Z}_N$, $\Phi(P) \in O(1/N)$.

Theorem 2 ([9]). *For any graph G and Markov chain $P \sim G$ that mixes to some distribution π, there exists an LMC $\hat{P} \sim G$ that mixes to π with a mixing time*

$$\tau \in O\left(\frac{1}{\Phi(P)} \log \frac{1}{\min_j \pi(j)}\right).$$

They actually provide an explicit construction for this LMC. It is however practically infeasible as it requires to solve complicated multicommodity flow problems over the graph, and the local coin size is $\Omega(|\mathcal{V}|)$.

We finally mention that we deviate in our definitions from the original definitions in [9] and [8], in which LMCs are defined without the initialization and marginalization maps F and f. Dropping F, they take into account all possible initial states $v \in \mathbb{R}^{\hat{\mathcal{V}}}$, rather than the initialized subset $F[v] \subset \mathbb{R}^{\hat{\mathcal{V}}}$ as we do. Dropping f, they require that an LMC converges on the entire lifted state space, whereas we only demand that the distribution of interest, being the marginal distribution, converges. A pragmatic motivation for our deviation is that F and f allow to describe LMCs as a mapping between distributions on the original node set $\mathcal{V}$, as we do in (1.5). This will allow us to straightforwardly compare different mixing schemes over $\mathcal{V}$ (e.g., casting them as stochastic processes, Section 4.1), all of which are maps over $\mathcal{V}$. Our results however carry over to the original framework, and can even strengthen certain results. We discuss this very thoroughly in [11]. Here is some overview:

- *Lower bounds on LMC mixing time.* We prove lower bounds on the mixing time of LMCs with initialization and marginalization maps. These maps form relaxations of LMC mixing without these maps: we only maximize the mixing time over a subset of "good" initializations of $\hat{\mathcal{V}}$, and we only require that the marginal converges, which by the triangle inequality converges at least as fast as the original distribution over $\hat{\mathcal{V}}$. Therefore our bounds carry over to the original setting, and in fact strengthen existing bounds to our relaxed setting.
- *Constructions of LMCs with a certain mixing time.* Initialization and marginalization maps relax the mixing time definition, so that our constructions do not directly carry over to similar LMC constructions in the original setting. However, as we discuss in [11], we can easily adapt LMC constructions with initialization and marginalization map to LMC constructions without these maps with the same mixing time.

Chapter 2
Quantum Mechanics and Quantum Walks

In this chapter we introduce our second main character: quantum walks. By looking at nature, scientists found that the classical laws underlying random walks did not suffice to describe a number of physical phenomena. A more complicated story did: the theory of quantum mechanics. We will show how principles such as superposition and measurement profoundly change the rules. Not only did quantum mechanics lead to a better understanding of existing phenomena, its peculiar features have driven, and in fact are driving, much of our current technologies.

In this thesis we will mainly concern ourselves with a particular derivative of the theory, called quantum walks – the quantum mechanical analogue of random walks. We already mentioned that random walks are ubiquitous for modeling natural processes and are a versatile algorithmic tool to explore and extract information about networks. Quantum walks are being attributed similar promises. They describe how the position of a "walking" quantum particle evolves on a graph. The original continuous time model was introduced by Aharonov, Davidovich and Zagury in 1993 [12], where it served as a physical model for a spin particle on a line under the influence of a magnetic field. They showed how quantum effects can cause the particle to behave radically different from a classical random walk, in some cases spreading out much faster.

The follow-up papers, focusing on the ones that lie closest to our cause, actually took inspiration from this to investigate quantum walks from an algorithmic point of view. Farhi and Gutmann [13] and Watrous [1] used quantum walks to show that quantum computers can simulate random walk algorithms for decision tree and graph connectivity problems, speeding up the random walk algorithm in certain cases. Ambainis et al [14] and Aharonov et al [15] defined general quantum walks on graphs, and they showed how quantum walks can speed up the diffusive mixing behavior of random walks. Following these results, research on quantum walks really took off, studying them as a paradigm for quantum computing [16, 17] and using them to speed up algorithmic search tasks such as the collision problem in [2], triangle finding in [18], and the problem of finding marked elements on graph [3, 19–22]. See [23] for a nice survey. Again from a physical point of view, they were subsequently associated to improved transport phenomena in biological systems [24, 25], thermodynamic theories, breakdown models and topological states of matter [26–29], and they have been simulated in various experiments [30–34].

Despite these impressive advances, many aspects of quantum walks remain poorly understood, leaving open a lot of important questions. For example, through a long series of papers [3, 21, 22, 35–37], a general quadratic speedup by quantum walks was established for the problem of finding a marked node on a graph. Albeit closer to the original observations on the line [14], the complementary problem of mixing has resisted such general speedup by quantum walk. There is evidence for a quadratic speedup on specific, highly symmetric graphs such as the cycle [15], hypercube [38], and torus [39]. A quadratic speedup was also proven under certain restrictive conditions: the existence of a slowly-varying sequence of Markov chains [40–43], or if the mixing time equals the hitting time of the graph [44]. It is conjectured in [45] that a general quadratic speedup is possible.

2.1 Quantum Mechanics

The quantum mechanical framework needed for this thesis is defined by three concepts: quantum states, their evolution, and their measurement. Quantum states and their evolution describe the state and dynamics of a quantum system, such as a molecular system or the inner workings of a quantum computer. We can gain information about this system by performing a measurement on it.

Quantum States: At any time, the state of a random walk X_t corresponds to a node of the graph: $X_t \in \mathcal{V}$. In quantum mechanics, this condition is relaxed and a walk can be in a superposition of nodes. Such superposition defines a *quantum state*, and can be described by a complex-valued vector

$$X_t = \sum_{j \in \mathcal{V}} \psi(j) e_j,$$

with $\{e_j\}_{j \in \mathcal{V}}$ a set of orthonormal basis vectors indexed by the graph nodes, and $\psi \in \mathbb{C}^{|\mathcal{V}|}$ a complex vector of 2-norm equal to 1, i.e., $\|\psi\| = \sum_{j \in \mathcal{V}} |\psi(j)|^2 = 1$. Throughout the thesis we will use the *bra-ket notation* for quantum states, where we write a *ket* $|\psi\rangle$ as a shorthand for the quantum state X_t with components ψ. Letting the ket $|j\rangle$ denote the basis vector e_j, we can expand the quantum state $|\psi\rangle$ as

$$|\psi\rangle = \sum_{j \in \mathcal{V}} \psi(j) |j\rangle.$$

A *bra* $\langle\psi| = \sum \overline{\psi}(j) \langle j|$ denotes the dual or Hermitian conjugate vector $\psi^\dagger$, with $\overline{\psi}$ the complex conjugate of ψ and $\langle j|$ denoting the Hermitian conjugate of the ket $|j\rangle$. To associate a vector space to these states, we define the inner product $\big\langle |\chi\rangle, |\psi\rangle \big\rangle$ between the quantum state $|\chi\rangle$ and $|\psi\rangle$ as the standard inner product between the vectors χ and ψ, i.e.,

$$\big\langle |\chi\rangle, |\psi\rangle \big\rangle = \langle \chi, \psi \rangle = \sum_{j \in \mathcal{V}} \overline{\chi}(j) \psi(j) \equiv \langle \chi | \psi \rangle.$$

The condition that quantum states have norm 1 then becomes $\langle\psi|\psi\rangle = 1$. We can now define the vector space $\mathcal{H}_\mathcal{V}$ as the vector space spanned by the set of orthonormal states $\{|j\rangle\}_{j \in \mathcal{V}}$, equipped with the above inner product. Throughout

we assume that this vector space is a *Hilbert space*, i.e., it is complete in some precise mathematical sense which we do not detail here.

Randomness in a classical system forces us to describe the state of a random walk $X_t \in \mathcal{V}$ by a probability distribution v over the states in $\mathcal{V}$. Similarly, randomness in a quantum system can force us to describe the state of a quantum walk $X_t \in \mathcal{H}_\mathcal{V}$ by a probability distribution v over quantum states in $\mathcal{H}_\mathcal{V}$. This results in an ensemble $\{v(i), |\psi_i\rangle\}$. If the ensemble is trivial ($\{|\psi\rangle\}$), and so the quantum state is simply $|\psi\rangle$, we call it a *pure state*. If the ensemble is nontrivial, we call it a *mixed state*. It turns out that any ensemble can be conveniently represented by the following operator

$$\rho = \sum_{i=1}^{|\mathcal{V}|} v(i)|\psi_i\rangle\langle\psi_i|,$$

which we call a *density operator*. The set of all density operators that describe such quantum ensembles corresponds to the set of all Hermitian linear operators ρ over $\mathcal{H}_\mathcal{V}$ that have trace 1,

$$\mathrm{tr}(\rho) = 1 = \sum \langle j|\rho|j\rangle,$$

and are positive semidefinite,

$$\langle\chi|\rho|\chi\rangle \geq 0, \quad \forall|\chi\rangle \in \mathcal{H}_\mathcal{V}.$$

A density operator represents a pure state if and only if it has rank 1. A classical probability distribution v over $\mathcal{V}$ corresponds to the ensemble $\{v(i), |i\rangle\}_{i\in\mathcal{V}}$, resulting in a density matrix that is diagonal in the node basis $\{|i\rangle\}_{i\in\mathcal{V}}$:

$$\rho_v = \sum_{i\in\mathcal{V}} v(i)|i\rangle\langle i|.$$

Evolution: The evolution of a pure quantum state $|\psi\rangle \mapsto |\psi'\rangle$ is described by a *unitary operator*, that is, $|\psi'\rangle = U|\psi\rangle$ where U is a linear operator such that

$$UU^\dagger = U^\dagger U = I.$$

Notice that $\langle\psi'|\psi'\rangle = \langle\psi|U^\dagger U|\psi\rangle = \langle\psi|\psi\rangle = 1$, so that indeed the unitary operator maps quantum states to quantum states. A crucial feature of unitary evolution is that it is invertible: $U^\dagger$ is again a unitary operator and $U^\dagger|\psi'\rangle = |\psi\rangle$. For further reference, we remind that a unitary operator U can always be decomposed as

$$U = \sum_k e^{i\theta_k} \Pi_k,$$

where $\{\Pi_k\}$ is a set of orthogonal projectors ($\Pi_k^2 = \Pi_k$), and $\{0 \leq \theta_k < 2\pi\}$ is a set of *eigenphases*.

Similarly to the case for quantum states, randomness can be involved in the evolution of a quantum system (and, as we will see below, measurements). Without going into details, we mention that the generalized evolution, now mapping density matrices to density matrices, can always be represented by a *Kraus map* Ψ:

$$\rho \mapsto \rho' = \Psi[\rho] = \sum_k B_k \rho B_k^\dagger,$$

where the *Kraus operators* $\{B_k\}$ form a set of linear operators such that $\sum_k B_k^\dagger B_k = I$. Indeed, ρ' is again a density operator:

$$\mathrm{tr}(\rho') = \mathrm{tr}\left(\sum_k B_k \rho B_k^\dagger\right) = \mathrm{tr}\left(\sum_k B_k^\dagger B_k \rho\right) = \mathrm{tr}(\rho) = 1,$$

using the fact that $\mathrm{tr}(AB) = \mathrm{tr}(BA)$, and for all $|\chi\rangle$ it holds that

$$\langle\chi|\rho'|\chi\rangle = \sum_k \langle\chi|B_k\rho B_k^\dagger|\chi\rangle = \sum_k (\langle\chi_k|B_k)\rho(B_k^\dagger|\chi_k\rangle) \geq 0,$$

using the fact that ρ is positive semidefinite[1]. Notice that the above unitary evolution is a special case, mapping an ensemble $\{v(i), |\psi_i\rangle\} \mapsto \{v(i), U|\psi_i\rangle\}$, and so $\rho' = \sum v(i)U|\psi_i\rangle\langle\psi_i|U^\dagger = U\rho U^\dagger$.

An easy example of a Kraus map arises when we randomly apply one of two possible unitary operators on a quantum state, applying U_1 with probability p and U_2 with probability $1-p$. The corresponding Kraus map is then

$$\rho \mapsto \rho' = \Psi[\rho] = pU_1\rho U_1^\dagger + (1-p)U_2\rho U_2^\dagger.$$

This Kraus map can be described by the Kraus operators $\{B_1 = \sqrt{p}U_1, B_2 = \sqrt{1-p}U_2\}$, for which indeed $B_1^\dagger B_1 + B_2^\dagger B_2 = I$.

Measurement: Taking a pragmatic viewpoint, quantum states and evolution are the "under the lid" description of a quantum system. We can probe this system from the exterior by performing measurements on it. For this thesis we will restrict ourselves to *projective measurements*, fully described by a "complete" set of options or outcomes. Mathematically this corresponds to a set $\{\Pi_k\}$ of orthogonal projectors,

$$\Pi_k^2 = \Pi_k, \qquad \Pi_k\Pi_{k'} = \delta_{k,k'}\Pi_k,$$

that partition the Hilbert space $\mathcal{H}_\mathcal{V}$:

$$\sum_k \Pi_k = I,$$

or equivalently $\bigcup_k \mathrm{Im}(\Pi_k) = \mathcal{H}_\mathcal{V}$ with "Im" denoting the image of an operator (which equals the support in the case of a projector). Now assume that the quantum system is in a pure state $|\psi\rangle$. Performing the measurement $\{\Pi_k\}$ returns the outcome "l", corresponding to projector Π_l, with a probability given by

$$\mathbb{P}(\text{outcome } l) = \|\Pi_l|\psi\rangle\|^2.$$

Crucially, the state of the quantum system after the measurement is changed, reflecting this outcome: following an outcome "l", the state becomes

$$|\psi'\rangle = \frac{\Pi_l|\psi\rangle}{\|\Pi_l|\psi\rangle\|},$$

which equals the (renormalized) projection of the initial state onto the image of Π_l. This process is called the "collapse" of the quantum state, randomly projecting

[1] In principle, yet of minor importance to this thesis, a stronger condition called *complete positivity* of the map should be checked. See for instance [46].

the quantum state into one of the measurement outcomes. Note that this collapse implies that if now we repeat the same measurement $\{\Pi_k\}$ we again find the outcome "l", now with probability 1 though, since $\mathbb{P}(\text{outcome } l) = \|\Pi_l|\psi'\rangle\|^2 = 1$.

By inference, if the quantum state is in a mixed state ρ, corresponding to an ensemble $\{v(i), |\psi_i\rangle\}$, this becomes

$$\mathbb{P}(\text{outcome } l) = \sum_i v(i)\|\Pi_l|\psi_i\rangle\|^2.$$

We can rewrite this expression by noting that

$$\begin{aligned}\sum_i v(i)\|\Pi_l|\psi_i\rangle\|^2 &= \sum_i v(i) \sum_{j\in\mathcal{V}} \langle j|\Pi_l|\psi_i\rangle\langle\psi_i|\Pi_l|j\rangle \\ &= \sum_i v(i)\text{tr}(\Pi_l|\psi_i\rangle\langle\psi_i|\Pi_l).\end{aligned}$$

Now using the fact that

$$\text{tr}(\Pi_l|\psi_i\rangle\langle\psi_i|\Pi_l) = \text{tr}(\Pi_l^2|\psi_i\rangle\langle\psi_i|) = \text{tr}(\Pi_l|\psi_i\rangle\langle\psi_i|),$$

this becomes $\mathbb{P}(\text{outcome } l) = \text{tr}(\Pi_l\rho)$.

As an example consider the pure state $|\psi\rangle = \sqrt{|\mathcal{V}|^{-1}}\sum_{j\in\mathcal{V}}|j\rangle$, that is, the uniform superposition over all the nodes of a graph. A measurement that we could perform is defined by a partitioning of the node set $\mathcal{V} = \bigcup_k \mathcal{S}_k$, leading to a set of projectors $\{\Pi_k = \sum_{j\in\mathcal{S}_k}|j\rangle\langle j|\}$. This measurement will yield outcome "l" with probability

$$\mathbb{P}(\text{outcome } l) = \|\Pi_l|\psi\rangle\|^2 = \sum_{j\in\mathcal{S}_l}|\langle j|\psi\rangle|^2,$$

collapsing the quantum state into

$$|\psi'\rangle = \frac{\Pi_l|\psi\rangle}{\|\Pi_l|\psi\rangle\|} = \frac{1}{\sqrt{|\mathcal{S}_l|}}\sum_{j\in\mathcal{S}_l}|j\rangle.$$

Another example derives from the set of Fourier modes

$$\{|\phi_k\rangle\}_{0\le k<|\mathcal{V}|}, \qquad \text{where } |\phi_k\rangle = \frac{1}{\sqrt{|\mathcal{V}|}}\sum_{j\in\mathcal{V}} e^{i\frac{2\pi k}{|\mathcal{V}|}j}|j\rangle,$$

which leads to a set of projectors $\{\Pi_k = |\phi_k\rangle\langle\phi_k|\}$. By noting that the uniform superposition $|\psi\rangle = |\phi_0\rangle$, we see that performing this measurement on $|\psi\rangle$ will yield the outcome "$l = 0$" with probability 1, since $\mathbb{P}(\text{outcome } 0) = \||\phi_0\rangle\langle\phi_0|\psi'\rangle\|^2 = 1$. As a result, the quantum state will not change: $|\psi'\rangle = \frac{\Pi_0|\psi\rangle}{\|\Pi_0|\psi\rangle\|} = |\psi\rangle$.

The set of Fourier modes forms an orthonormal basis for $\mathcal{H}_\mathcal{V}$. In fact we can associate a measurement $\{\Pi_k = |\varphi_k\rangle\langle\varphi_k|\}$ to any orthonormal basis $\{|\varphi_k\rangle\}$. Performing this measurement is commonly referred to as *measuring in the* $\{|\varphi_k\rangle\}$ *basis*. In this thesis we will be mainly concerned with *measurements in the standard basis*, where the standard basis denotes the set of node states $\{|j\rangle\}_{j\in\mathcal{V}}$ of the graph under investigation. Whenever we discuss a measurement without specifying the set of measurement operators, we will mean a measurement in the standard basis.

2.2 Unitary Quantum Walks

The simplest form of a *quantum walk* (QW) on a graph $G = (\mathcal{V}, \mathcal{E})$ is a *unitary QW*, defined by a unitary transition operator U over $\mathcal{H}_\mathcal{V}$. Similarly to a Markov chain transition operator over G, we require that U respect the locality of G:

$$\langle j|U|i\rangle = U(j,i) = 0 \text{ if } (i,j) \notin \mathcal{E}.$$

We denote this condition by $U \sim G$. If this condition does not hold, so there exists $(i,j) \notin \mathcal{E}$ for which $\langle j|U|i\rangle \neq 0$, then applying a QW step on the state $|i\rangle$, and performing a measurement on the new state $U|i\rangle$, returns the state j with a nonzero probability $\|\Pi_j U|i\rangle\|^2 = |\langle j|U|i\rangle|^2 > 0$. Therefore a forbidden transition takes place with a nonzero probability. If the locality condition does hold, then no such transitions can take place.

As an example we wish to describe a quantum walk on the cycle $\mathbb{Z}_N$. It was however noted early on, in the 1996 work by Meyer [47], that any unitary quantum walk on the cycle is trivial. Indeed the only unitary matrix that is translationally invariant and obeys the locality of the cycle is $U = e^{i\phi}P^\uparrow$ or $U = e^{i\phi}P^\downarrow$, with $\phi \in [0, 2\pi)$ some phase. This corresponds to trivial dynamics, where the quantum state is simply multiplied by a phase and translated along the cycle. In the example below we will show that lifting the graph, like we did for the LMC (example 5), remedies this issue. It is for this reason that quantum walks are generally defined over a lifted graph $\hat{G}$, similarly to LMCs. Strictly speaking one could call these "lifted quantum walks", but due to the lack of non-trivial non-lifted quantum walks on general graphs they are simply referred to as quantum walks.

Example 6 (Unitary QW on $\mathbb{Z}_N$). Similar to the LMC example on $\mathbb{Z}_N$, we define the lifted graph $\hat{G} = (\hat{\mathcal{V}}, \hat{\mathcal{E}})$ by

$$\hat{\mathcal{V}} = C \times \mathcal{V} = \{\uparrow, \downarrow\} \times \{0, 1, 2, \ldots, N-1\}.$$

A unitary QW $\hat{U}$ on $\hat{G}$ can be described as

$$\hat{U} = S \cdot (C \otimes I), \quad C \in U(2),$$

where $S = |\uparrow\rangle\langle\uparrow| \otimes P^\uparrow + |\downarrow\rangle\langle\downarrow| \otimes P^\downarrow$ is again the shift operator as in Example 5, and $U(2)$ denotes the group of unitary 2 by 2 matrices. An example is

$$C = \begin{bmatrix} \sqrt{1-\alpha} & \sqrt{\alpha} \\ \sqrt{\alpha} & -\sqrt{1-\alpha} \end{bmatrix}.$$

Notice the very similar structure to the LMC $\hat{P}$ defined in (1.6). Both $\hat{U}$ and $\hat{P}$ consist of a unitary resp. stochastic coin toss $C \otimes I$ and the conditional shift S. The latter is a permutation matrix, and hence both unitary and stochastic.

Setting $\alpha = 1/2$ in the above matrix yields the Hadamard matrix, and the corresponding QW is called the *Hadamard QW*. In Figure 2.2 we show a plot of the probability distribution describing the measurement outcome of $\hat{U}^t|\uparrow, 0\rangle$ for $t = 100$, the quantum state after 100 steps of the Hadamard QW on the initial state $|\uparrow, 0\rangle$. This QW is extensively discussed in the early papers [14, 15]. Its behavior is strikingly different from a random walk. The random walk dynamics are diffusive,

its distribution having a standard deviation in $\Theta(\sqrt{t})$ after t steps. On the other hand, the probability distribution underlying the QW dynamics, when projected back to the original graph G, are ballistic, spreading out quadratically faster. This will turn out to be one of the quintessential features of quantum walks that motivate this thesis. △

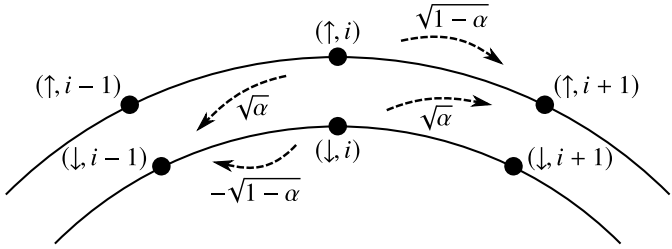


Fig. 2.1 Figure of a quantum walk on the cycle $\mathbb{Z}_N$. For $\alpha = 1/2$ this is the Hadamard quantum walk discussed in for instance [14, 15]. Similarly to the lifted Markov chain, shown in Figure 1.3, the quantum walk spreads out quadratically faster than the random walk on $\mathbb{Z}_N$.

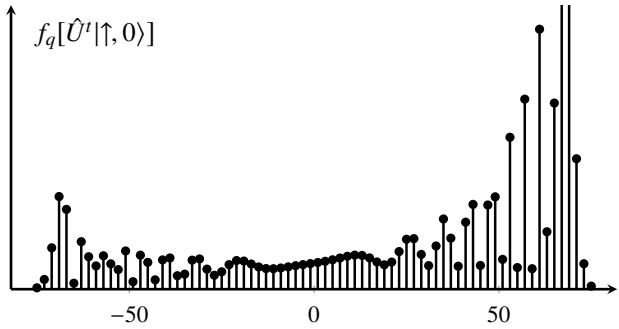


Fig. 2.2 Plot showing the probability distribution following a measurement of the Hadamard quantum walk $\hat{U}^t|\uparrow, 0\rangle$ for $t = 100$ (see next section for definition of f_q). Due to quantum interference effects the distribution is much less smooth than the random walk distribution, shown in Figure 1.2. The distribution has a standard deviation that scales with t, quadratically improving the standard deviation of the random walk.

2.3 Nonunitary Quantum Walks and Mixing

The fundamental theorem of Markov chains states that an ergodic Markov chain will converge or mix to a unique limit distribution: $\lim_{t\to\infty} P^t v = \pi$. This lies at the basis of all Markov chain sampling algorithms, which have the aim to sample from the distribution π. A unitary QW does not converge: the limit $\lim_{t\to\infty} U^t|\psi\rangle$ generally does not exist. Even stronger, a unitary QW is *quasi-periodic*: for any

$\epsilon > 0$ there exist infinitely many τ for which $\|U^\tau|\psi\rangle - |\psi\rangle\| \leq \epsilon$, for all $|\psi\rangle$. To see this, recall that $U = \sum_k e^{i\theta_k} \Pi_k$ for some set $\{\Pi_k\}$ of orthogonal projectors. For any finite set $\{e^{i\theta_k}\}$ there exist infinitely many τ such that for each k it holds that $|e^{i\theta_k \tau} - 1| \leq \epsilon$. For any such τ, we indeed find that

$$\begin{aligned}\|U^\tau|\psi\rangle - |\psi\rangle\| &= \sqrt{\sum_k |e^{i\theta_k\tau} - 1|^2 \|\Pi_k|\psi\rangle\|^2} \\ &\leq \max_k |1 - e^{i\theta_k\tau}| \leq \epsilon,\end{aligned}$$

using the fact that $\sum_k \|\Pi_k|\psi\rangle\|^2 = \||\psi\rangle\|^2 = 1$.

As a consequence, unitary QWs do not seem suited for mixing purposes. It is for this reason that we introduce the most general form of a quantum walk: a Kraus map Ψ, defined by a set of Kraus operators $\{B_k\}$. In the density operator formalism, this gives rise to a *quantum Markov chain* with transitions

$$\rho_t \mapsto \rho_{t+1} = \Psi[\rho_t] = \sum_k B_k \rho_t B_k^\dagger.$$

Similarly to the unitary transition operator, we demand that these operators respect the locality of G:

$$\forall k : \ \langle j|B_k|i\rangle = B_k(j,i) = 0 \ \text{ if } \ (i,j) \notin \mathcal{E}.$$

Recall that we generally define QWs on a lifted graph $\hat{G} = (\hat{\mathcal{V}}, \hat{\mathcal{E}})$[2], whereas we are actually interested in sampling from a distribution over the original graph $G = (\mathcal{V}, \mathcal{E})$. It is for this reason that we will often associate an initialization and marginalization map to a QW, similarly to what we did for LMCs. This results in what we call a *quantum mixing scheme*, which finally maps an input probability distribution[3] over $\mathcal{V}$ to an output probability distribution over $\mathcal{V}$, again similar to LMCs. The initialization map is straightforward, being a one-to-one map $F : \mathcal{V} \mapsto \hat{\mathcal{V}}$. We extend it to a linear map from probability distributions over $\mathcal{V}$ to quantum states over $\hat{\mathcal{V}}$ by setting

$$v \mapsto F[v] = \rho_v = \sum_{j \in \mathcal{V}} v(j) |F(j)\rangle\langle F(j)|.$$

Towards defining the *marginalization or output map*, recall that we are finally interested in a classical distribution over $\mathcal{V}$. We associate a marginalization or output to a quantum state over the lifted node set $\hat{\mathcal{V}}$ as follows:

1. perform a measurement $\{\Pi_k = |k\rangle\langle k|\}_{k \in \hat{\mathcal{V}}}$ of the quantum state in the lifted node basis,
2. output the "parent node" in $\mathcal{V}$ that corresponds to the outcome node of the measurement.

[2] We note that if we continue to impose translational invariance of the dynamics, then lifting the state space is still required (even for this generalized form) if we want to have non-trivial quantum effects.

[3] Later on we will compare classical mixing schemes with quantum mixing schemes. To be able to compare these, we will assume that the input of a mixing scheme is always a classical distribution.

The output of this scheme can be described by a probability distribution, so that the scheme in fact maps a quantum state over $\hat{\mathcal{V}}$ to a probability distribution over $\mathcal{V}$. We denote this map by f_q, and for a pure quantum state $|\psi\rangle$ over $\hat{\mathcal{V}}$ it is defined by

$$f_q[|\psi\rangle] = f(v_{|\psi\rangle}), \quad \text{where} \quad v_{|\psi\rangle}(j) = |\langle j|\psi\rangle|^2, \tag{2.1}$$

with f the original marginalization map associated to $\hat{G}$, mapping nodes in $\hat{\mathcal{V}}$ to their parent nodes in $\mathcal{V}$. If the quantum state is described by a density operator ρ, then the map becomes

$$f_q[\rho] = f(v_\rho), \quad \text{where} \quad v_\rho(j) = \mathrm{tr}(|j\rangle\langle j|\rho) = \langle j|\rho|j\rangle.$$

Combining a QW Ψ with an initialization map F and a marginalization map f_q yields a *quantum mixing scheme*, mapping an "input" distribution v_0 over $\mathcal{V}$ to an "output" distribution v_t over $\mathcal{V}$:

$$v_t = (f_q \circ \Psi^t \circ F)[v_0].$$

Similar to the classical mixing scheme, we say that the quantum scheme mixes to a distribution π if for any v_0 it holds that

$$\lim_{t\to\infty} (f_q \circ \Psi^t \circ F)[v_0] = \pi.$$

We define its ϵ-mixing time $\tau(\epsilon)$ as

$$\tau(\epsilon) = \min\{T \mid \max_v \|(f_q \circ \Psi^t \circ F)[v] - \pi\|_{\mathrm{TV}} \leq \epsilon, \forall t \geq T\}.$$

Example 7 (Classical Markov Chains). Quantum walks are a generalization of classical Markov chains, so it should be possible to cast a general classical Markov chains as a quantum walk. Indeed, consider a lifted Markov chain $\hat{P}$ with initialization map F and marginalization map f (for a simple Markov chain, F and f are simply the identity). We can associate a quantum walk Ψ to this chain through the Kraus operators

$$\left\{ B_{ji} = \sqrt{\hat{P}(j,i)}|j\rangle\langle i| \right\}.$$

These give rise to the map

$$\Psi[\rho] = \sum_{j,i\in\hat{\mathcal{V}}} \hat{P}(j,i)|j\rangle\langle i|\rho|i\rangle\langle j|. \tag{2.2}$$

A classical probability distribution v over the node set $\hat{\mathcal{V}}$, encoded as a diagonal quantum state ρ_v, then gets mapped correctly to $\hat{P}v$, encoded as the diagonal $\rho_{\hat{P}v}$:

$$\begin{aligned}
\rho_v = \sum_{i\in\hat{\mathcal{V}}} v(i)|i\rangle\langle i| \mapsto \Psi[\rho_v] &= \sum_{j,i\in\hat{\mathcal{V}}} \hat{P}(j,i)|j\rangle\langle i|\rho_v|i\rangle\langle j| \\
&= \sum_{j,i\in\hat{\mathcal{V}}} \hat{P}(j,i)v(j)|j\rangle\langle j| \\
&= \sum_{j\in\hat{\mathcal{V}}} (\hat{P}v)(j)|j\rangle\langle j| = \rho_{\hat{P}v}.
\end{aligned}$$

It is a direct consequence that

$$f_q \circ \Psi^t \circ F \equiv f \circ \hat{P}^t \circ F,$$

so that the corresponding mixing schemes are also equal. △

As a more interesting example, we introduce a non-unitary QW on the cycle.

Example 8 (Non-unitary QW on $\mathbb{Z}_N$). The unitary QW $\hat{U}$ on the lifted cycle graph $\hat{G}$, introduced in the previous section, does not mix. We can however adapt it easily into a closely associated Kraus map that does mix. Thereto consider the following transition rule, for some $0 \leq p \leq 1$:

Apply $\hat{U}$. With probability p perform a measurement in the lifted node basis.

This results in the Kraus map

$$\rho \mapsto \rho' = \Psi[\rho] = (1-p)\hat{U}\rho\hat{U}^\dagger + p\mathbb{M}[\hat{U}\rho\hat{U}^\dagger].$$

Here the operator $\mathbb{M}$ represents the measurement, mapping the quantum state ρ to the ensemble

$$\{v(j) = \langle j|\rho|j\rangle, |j\rangle\}_{j\in\hat{\mathcal{V}}},$$

or equivalently

$$\rho \mapsto \rho' = \mathbb{M}[\rho] = \sum_{j\in\hat{\mathcal{V}}} \langle j|\rho|j\rangle |j\rangle\langle j|.$$

We can rewrite $\langle j|\rho|j\rangle|j\rangle\langle j| = |j\rangle\langle j|\rho|j\rangle\langle j|$, so that the measurement operator $\mathbb{M}$ can be represented by a Kraus map with operators $\{B_j = |j\rangle\langle j|\}_{j\in\hat{\mathcal{V}}}$.

Notice that if $p = 1$, this Kraus map actually corresponds to the Kraus map (2.2) for $\hat{P}$ the LMC (1.6) on $\mathbb{Z}_N$ that we introduced earlier. As a consequence, if we implement the QW $\hat{U}$ and we measure after every step, we actually retrieve the LMC.

The Kraus map Ψ for $p \in (0,1)$ is discussed in amongst others [39, 48]. There it is shown that indeed this map converges: for any density operator ρ it holds that

$$\lim_{t\to\infty} \Psi^t[\rho] = \frac{1}{|\hat{\mathcal{V}}|} I = \frac{1}{|\hat{\mathcal{V}}|} \sum_{j\in\hat{\mathcal{V}}} |j\rangle\langle j|,$$

corresponding to the uniformly distributed ensemble

$$\left\{v(j) = |\hat{\mathcal{V}}|^{-1}, |j\rangle\right\}_{j\in\hat{\mathcal{V}}}.$$

This implies that the corresponding mixing scheme, defined above, converges to the uniform distribution:

$$\lim_{t\to\infty} (f_q \circ \Psi^t \circ F)[v_0] = \pi.$$

In the previous section we mentioned the remarkable fact that the unitary quantum walk $\hat{U}$ on $\mathbb{Z}_N$ spreads out quadratically faster than the random walk. In [39, 48] it is shown that this ballistic behavior allows the closely related quantum walk Ψ to mix quadratically faster than the random walk. Indeed, they show that the mixing time of the scheme based on Ψ has a mixing time $\tau \in \Theta(N)$. △

Apart from the cycle, a very similar quantum walk construction has been shown to quadratically speed up mixing on the lattice [49], and a log-factor speedup was shown on the hypercube [38, 50]. This quadratic speedup of quantum walks with respect to random walks lies at the basis of much of the following work.

We mentioned that general unitary quantum walks show quasi-periodic behavior that prevents them from mixing, and how this can be remedied by considering nonunitary quantum walks as in the above example. A tangent approach to this problem, as considered in e.g. [15, 39, 49], is to consider the *Cesaro average* $\bar{v}_t$ of the quantum walk mixing scheme, defined as:

$$\bar{v}_t = \frac{1}{t} \sum_{l=0}^{t-1} v_l.$$

At time step t the output distribution is thus defined as the uniform time average of the original output distributions. As a trivial example, we can consider the unitary quantum walk which moves deterministically to the right on the cycle $\mathbb{Z}_N$. Starting from a single node, the corresponding output distribution v_t will always be localized on a single node and will never mix. Its time average $\bar{v}_t$ can however easily be shown to converge to the uniform distribution in $\Theta(N)$ steps, so that the *Cesaro mixing time* of the walk is defined to be $\Theta(N)$. As we discuss in Section 5.3, our results also carry over to this setting of unitary quantum walks with a finite Cesaro mixing time.

1st Part: Speed Limits

In this part of the thesis we prove *speed limits* that characterize the mixing performance of quantum walks, and more general stochastic processes. This is a central theme in research on the algorithmic use of quantum walks for speeding up Markov chain algorithms [15, 51]. The mixing performance also plays a key role in the study of physical processes, such as the thermodynamic relaxation of quantum systems [52] or the scrambling of quantum information around black holes [53].

We prove speed limits that are determined by a property of the underlying graph called the conductance. This conductance forms an intuitive geometric measure of the occurrence of bottlenecks in the graph, restricting the probability flow from certain sections to other sections. Similarly to an electrical circuit, a graph has a low conductance if this flow is too restricted. Speed limits derived from the graph conductance stem from early results in differential geometry [54, 55], which took inspiration from the classic subject of isoperimetric problems. Their utility for Markov chains was noted rather quickly [56, 57], and they were turned into one of the standard tools to estimate their mixing time. As we demonstrate in Section 3.2, the proof of such conductance bounds for Markov chains relies on typical aspects of classical Markov chains such as contractivity and nonnegativity.

Quantum walks, as we discussed in 2.2, generally lack such well-behavedness due to for instance interference effects (compare Figure 1.2 with Figure 2.2). This has allowed them to evade general speed limits for a long time. Existing efforts can be found in [15] and [51], yet both are restricted to certain special families of quantum walks and their methods seem not to generalize. We succeed in proving speed limits for general quantum walks, building on an insight of independent interest: in Chapter 4 we show that quantum walks can be simulated by classical Markov chains with memory. Thereto we generalize results from hidden variables theory [58], showing the existence of certain "stochastic" bridges which allow to classically track the underlying probability distribution of a quantum walk. We combine these bridges into a single lifted Markov chain, reminding a classical version of the "clock Hamiltonians" by Feynman [59] and Kitaev [60] used to prove the universality of adiabatic quantum computing. These classical lifted Markov chains bring us back to familiar territory, allowing us to prove a conductance bound on these LMC simulators in Chapter 5, and by inference also on the original quantum walks.

These results provide several insights. First of all, quantum walks have long been conjectured to show a speedup in mixing behavior [24, 25, 39]. From a physics perspective, this could lead to mixing time as a "quantum signaler", where an improved mixing behavior is diagnostic for quantum effects. By our results, we see that this is not the case: any speedup in mixing attained by a quantum walk can also be attained by a classical Markov chain with memory. Second, from a quantum computing perspective it is valuable to identify tasks for which quantum walks can achieve an advantage when compared with classical schemes. From our results it follows that a quantum advantage for mixing should focus on design and resource constraints, rather than simply on the speed of mixing. Indeed, one of the key features of a random walk is its simple design and efficient implementation. If a similarly efficient QW design would allow to accelerate the random walk mixing time, then this would still provide an advantage over our more complicated lifted Markov chain construction, even though they have the same mixing time.

Chapter 3
Conductance Bound for Markov Chains

Mixing describes the general convergence of the dynamics of stochastic processes, examples of which are the equilibration of thermodynamical systems and the spreading of rumors or diseases. Mixing schemes are also important subroutines of many algorithms such as the Metropolis algorithm for Markov chain Monte Carlo, see Example 3 in Chapter 1, the approximation algorithms for the permanent of a 0-1 matrix [61] and the volume of a convex body [62]. In Chapter 1 we have introduced the standard concept of mixing time as a quantifier for the time it takes a Markov chain to mix over a graph, and we have discussed how it can be estimated by the spectral gap of the Markov chain transition operator. In this chapter we elaborate the existing techniques on using the intuitive concept of conductance bounds to estimate these quantities. We introduce techniques and lemmas that are crucial to our further results.

The existence of conductance bounds solidifies the following intuition: if a graph is "badly connected" in some sense, then a Markov chain will spread and mix slower than otherwise. A natural candidate to quantify this connectedness is the graph diameter D, measuring the maximum distance between any two nodes. Indeed, we can prove the following classic bound [5].

Proposition 2 (Diameter bound). *If a graph G has diameter D, then any Markov chain $P \sim G$ has a mixing time $\tau \geq D/2$.*

Proof. From the triangle inequality we know that

$$\|P^t v - P^t w\|_{\mathrm{TV}} \leq \|P^t v - \pi\|_{\mathrm{TV}} + \|P^t w - \pi\|_{\mathrm{TV}}.$$

By definition of the diameter, there exist $x, y \in \mathcal{V}$ such that $\mathrm{dist}(x, y) = D$. It is clear that then $P^t e_x$ and $P^t e_y$, for $t = \lfloor (D-1)/2 \rfloor$, are supported on disjoint subsets of the nodes. By the operational definition of the total variation distance, this shows that

$$\|P^t e_x - P^t e_y\|_{\mathrm{TV}} = \max_{\mathcal{S} \subset \mathcal{V}} |(P^t e_x)(\mathcal{S}) - (P^t e_y)(\mathcal{S})| = 1,$$

where the last equality follows from setting $\mathcal{S}$ equal to the support of $P^t e_x$ or $P^t e_y$. Applying the triangle inequality on the left side then shows that $\max\left\{\|P^t e_x - \pi\|_{\mathrm{TV}}, \|P^t e_y - \pi\|_{\mathrm{TV}}\right\} \geq \frac{1}{2}$. This shows that the mixing time $\tau > \lfloor (D-1)/2 \rfloor$, or equivalently $\tau \geq D/2$. □

This gives a tight bound for the cycle $\mathbb{Z}_N$. In general however the diameter does not appropriately capture the connectedness of the graph, causing this diameter bound to be much too loose. Consider for instance the *dumbbell graph* $K_N - K_N$, depicted in Figure 3.1, consisting of two complete graphs over N nodes connected by a single edge. The diameter of this graph is three, yet it can be shown that the random walk mixing time on this graph is $\Omega(N^2)$ [63]. This is easily seen by noting that starting from the left cluster, it will take $\Theta(N)$ expected steps before reaching the central node, after which the central edge will be traversed only with probability $1/N$. This example, be it in a differential geometry setting, was originally ascribed to Eugenio Calabi. It served as the motivating example for the seminal 1969 paper by Jeff Cheeger [54], where he first introduced the use of conductance bounds, the main subject of this thesis part.

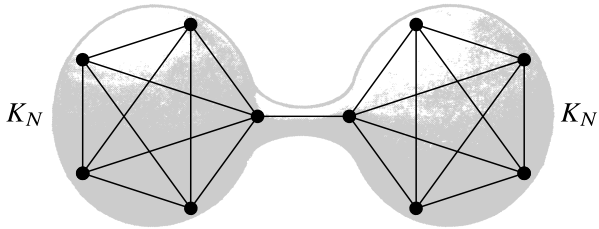


Fig. 3.1 Dumbbell graph $K_N - K_N$, consisting of two complete graphs K_N connected by a single edge. The graphic underlying the graph is taken from the original paper by Cheeger [54].

It is clear that the graph $K_N - K_N$ is poorly connected: the left half is connected to the right half by a single edge, which severely congests the flow of probability. The graph diameter does not capture the occurrence of such bottlenecks, rendering the trivial diameter bound poorly useful. A different quantity, called the *conductance* of the graph, remedies this. In the following section we will define this intuitive quantity, and show how it can be used to bound the mixing time.

3.1 Markov Chain Conductance

Consider a Markov chain P over a graph $G = (\mathcal{V}, \mathcal{E})$ with stationary distribution π, and a subset $\mathcal{S} \subset \mathcal{V}$. The conductance $\Phi(\mathcal{S})$ of P with respect to the cut $(\mathcal{S}, \mathcal{S}^c)$ is defined rather technically by

$$\Phi(\mathcal{S}) = \frac{Q(\mathcal{S}, \mathcal{S}^c)}{\pi(\mathcal{S})} = (P\pi_{\mathcal{S}})(\mathcal{S}^c), \tag{3.1}$$

with $\pi(\mathcal{S}) = \sum_{j \in \mathcal{S}} \pi(j)$ and $Q(\mathcal{S}, \mathcal{S}^c)$ the ergodic flow from $\mathcal{S}$ to its complement $\mathcal{S}^c$, defined as

$$Q(\mathcal{S}, \mathcal{S}^c) = \sum_{i \in \mathcal{S}, j \in \mathcal{S}^c} Q(i, j), \quad \text{with} \quad Q(i, j) = P(j, i)\pi(i).$$

Intuitively, $\Phi(\mathcal{S})$ equals the probability that flows from $\mathcal{S}$ to its complement $\mathcal{S}^c$ in a single step of the Markov chain, starting from $\pi_{\mathcal{S}}$. The quantity $\pi(\mathcal{S})$ in the

denominator is often seen as a measure of the size, volume or relative importance of the subset $\mathcal{S}$, as determined by the stationary distribution π. For a random walk, $\pi(\mathcal{S}) = d(\mathcal{S})/d_{\text{tot}}$, the total degree inside $\mathcal{S}$ relative to the total degree of the graph. The quantity $Q(\mathcal{S}, \mathcal{S}^c)$ is seen as a measure of the boundary of $\mathcal{S}$, determined by both π and P. The conductance $\Phi(\mathcal{S})$ with respect to the cut $(\mathcal{S}, \mathcal{S}^c)$ then measures the size of the boundary of $\mathcal{S}$ over the size of $\mathcal{S}$. The conductance $\Phi(P)$ of the Markov chain P is defined as the worst conductance over all subsets $\mathcal{S}$ that are not too big:

$$\Phi(P) = \min_{\pi(\mathcal{S}) \leq 1/2} \Phi(\mathcal{S}). \tag{3.2}$$

This results in the *isoperimetric inequality* $\pi(\mathcal{S}) \leq \Phi(P)^{-1} Q(\mathcal{S}, \mathcal{S}^c)$ for all $\mathcal{S}$ for which $\pi(\mathcal{S}) \leq 1/2$. Such isoperimetric inequalities, bounding the maximum volume that can be enclosed by a boundary of fixed size, have a geometric origin. The classic example of such an inequality states that the area of any region in the plane, whose boundary has a length 1, is at most $1/(4\pi)$. This maximum is attained by the circle.

If P denotes a simple random walk, the expression $\Phi(P)$ simplifies. Thereto recall that the stationary distribution π of a random walk is described by $\pi(j) = d(j)/d_{\text{tot}}$, see (1.1), and so the ergodic flow $Q(i, j) = 1/d_{\text{tot}}$. The conductance of a set $\mathcal{S}$ then becomes

$$\Phi(\mathcal{S}) = \sum_{i \in \mathcal{S}, j \in \mathcal{S}^c} \frac{1/d_{\text{tot}}}{d(\mathcal{S})/d_{\text{tot}}} = \frac{|E(\mathcal{S}, \mathcal{S}^c)|}{d(\mathcal{S})} = \frac{|E(\mathcal{S}, \mathcal{S}^c)|}{|E(\mathcal{S})|},$$

where $E(\mathcal{S}, \mathcal{S}^c)$ denotes the set of directed edges from $\mathcal{S}$ to $\mathcal{S}^c$, $E(\mathcal{S})$ the set of directed edges starting in $\mathcal{S}$, and $d(\mathcal{S}) = \sum_{j \in \mathcal{S}} d(j)$. The conductance $\Phi(P)$ for a random walk P thus becomes

$$\Phi(P) = \min_{d(\mathcal{S}) \leq d_{\text{tot}}/2} \frac{|E(\mathcal{S}, \mathcal{S}^c)|}{d(\mathcal{S})}. \tag{3.3}$$

For the dumbbell graph $K_N - K_N$, which we discussed earlier, it is easily seen that the worst cut is given by cutting the central edge, so that $|E(\mathcal{S}, \mathcal{S}^c)| = 1$ and $d(\mathcal{S}) = N^2 - N + 1$. The random walk conductance thus becomes $\Phi(P) = \frac{1}{N^2 - N + 1} \in \Theta(1/N^2)$. We see that, contrary to the diameter, the conductance effectively captures the bottleneck of the dumbbell graph.

An interesting and intuitive estimate of the conductance follows from bounding

$$\Phi(\mathcal{S}) = \frac{\sum_{j \in \mathcal{S}^c, i \in \mathcal{S}} P(j, i)\pi(i)}{\pi(\mathcal{S})} \leq \frac{\sum_{i \in \partial \mathcal{S}} \pi(i)}{\pi(\mathcal{S})} = \frac{\pi(\partial \mathcal{S})}{\pi(\mathcal{S})},$$

where $\partial \mathcal{S}$ denotes the inner boundary of $\mathcal{S}$, that is, the set of nodes in $\mathcal{S}$ which have an edge going to $\mathcal{S}^c$. We see that $\Phi(\mathcal{S})$ forms a measure for the size of the boundary of $\mathcal{S}$ with respect to the size of $\mathcal{S}$. It follows that

$$\Phi(P) = \min_{\pi(\mathcal{S}) \leq 1/2} \Phi(\mathcal{S}) \leq \min_{\pi(\mathcal{S}) \leq 1/2} \frac{\pi(\partial \mathcal{S})}{\pi(\mathcal{S})}. \tag{3.4}$$

In the following section we elaborate the proof techniques for the intuition that the mixing time should behave inversely proportional to the conductance.

3.2 Conductance Bound

Conductance bounds were originally proven in a differential geometry setting. Cheeger [54] and Buser [55] showed that the smallest positive eigenvalue of the Laplacian on a Riemannian manifold can be bounded by the isoperimetric constant of that manifold. These results were translated to a discrete geometric and graph setting by the seminal work of Fiedler [64], Dodziuk [65] and Alon [57]. The smallest positive eigenvalue of the graph Laplacian is nowadays called the Fiedler value, in honor of this work. Alon [57] used these bounds to prove critical results on an elusive family of graphs called "expander graphs", to which we will come back later. In Markov chain analysis, the work by Aldous [56], Lawler and Sokal [66], Mihail [67] and Jerrum and Sinclair [68] led to the final form presented here.

The crucial instrument for the Markov chain conductance bound will be the following lemma, which shows how the conductance restricts the amount of probability flow through a cut. The lemma and the conductance bound following from it are known results and can be found in for instance [5, 69], yet the proofs are very instructive and we will need them later.

Lemma 3. *If P is an irreducible Markov chain on a graph $G = (\mathcal{V}, \mathcal{E})$ with stationary distribution π, then*

$$(P^t \pi_{\mathcal{S}})(\mathcal{S}^c) \leq t\Phi(\mathcal{S}).$$

Proof. We will prove this statement by showing that the "net probability flow" per time step from $\mathcal{S}$ to $\mathcal{S}^c$ is bounded by $\Phi(\mathcal{S})$. More precisely,

$$(P^{l+1}\pi_{\mathcal{S}})(\mathcal{S}^c) - (P^l \pi_{\mathcal{S}})(\mathcal{S}^c) \leq \Phi(\mathcal{S}) = (P\pi_{\mathcal{S}})(\mathcal{S}^c), \quad \forall l \geq 0. \tag{3.5}$$

This suffices to prove the lemma, since we can rewrite

$$(P^t \pi_{\mathcal{S}})(\mathcal{S}^c) = \sum_{l=1}^{t} \Big((P^l \pi_{\mathcal{S}})(\mathcal{S}^c) - (P^{l-1}\pi_{\mathcal{S}})(\mathcal{S}^c) \Big),$$

where we used that $(P^0 \pi_{\mathcal{S}})(\mathcal{S}^c) = \pi_{\mathcal{S}}(\mathcal{S}^c) = 0$. To prove (3.5), we use that by the operational interpretation of the total variation distance (1.4) it holds that

$$|(P^{l+1}\pi_{\mathcal{S}})(\mathcal{S}^c) - (P^l \pi_{\mathcal{S}})(\mathcal{S}^c)| \leq \left\| P^{l+1}\pi_{\mathcal{S}} - P^l \pi_{\mathcal{S}} \right\|_{\mathrm{TV}}.$$

Now we can evoke the fact that the total variation distance is nonincreasing under a stochastic matrix: $\|Pw\|_{\mathrm{TV}} \leq \|w\|_{\mathrm{TV}}$ for any vector w (see proof of Lemma 1). This implies that

$$\left\| P^{l+1}\pi_{\mathcal{S}} - P^l \pi_{\mathcal{S}} \right\|_{\mathrm{TV}} \leq \|P\pi_{\mathcal{S}} - \pi_{\mathcal{S}}\|_{\mathrm{TV}}.$$

We finish the proof by showing that $\|P\pi_{\mathcal{S}} - \pi_{\mathcal{S}}\|_{\mathrm{TV}} = (P\pi_{\mathcal{S}})(\mathcal{S}^c)$, which equals $\Phi(\mathcal{S})$. We can use the fact that for any distributions p and q

$$\|p - q\|_{\mathrm{TV}} = \max_{\mathcal{W} \subseteq \mathcal{V}} |p(\mathcal{W}) - q(\mathcal{W})| = \sum_{j : p(j) > q(j)} \big(p(j) - q(j) \big).$$

For all $j \in \mathcal{S}^c$ it trivially holds that $(P\pi_{\mathcal{S}})(j) - \pi_{\mathcal{S}}(j) = (P\pi_{\mathcal{S}})(j) \geq 0$, and for all $j \in \mathcal{S}$ we see that

$$(P\pi_{\mathcal{S}})(j) = \frac{\sum_{i\in\mathcal{S}} P(j,i)\pi(i)}{\pi(\mathcal{S})} \leq \frac{\sum_{i\in\mathcal{V}} P(j,i)\pi(i)}{\pi(\mathcal{S})} = \frac{\pi(j)}{\pi(\mathcal{S})} = \pi_{\mathcal{S}}(j),$$

using the fact that $P\pi = \pi$. This implies that we can rewrite

$$\begin{aligned}\|P\pi_{\mathcal{S}} - \pi_{\mathcal{S}}\|_{\mathrm{TV}} &= \sum_{j:(P\pi_{\mathcal{S}})(j)>\pi_{\mathcal{S}}(j)} (P\pi_{\mathcal{S}})(j) - \pi_{\mathcal{S}}(j) \\ &= \sum_{j\in\mathcal{S}^c} (P\pi_{\mathcal{S}})(j) - \pi_{\mathcal{S}}(j) \\ &= (P\pi_{\mathcal{S}})(\mathcal{S}^c) = \Phi(S),\end{aligned}$$

which finalizes the proof. □

From this lemma it is intuitively clear that the mixing time should be inversely proportional to the conductance $\Phi(P)$. Indeed we can now easily prove the following proposition, which we call the conductance bound for Markov chains. Recall that τ denotes the $(1/4)$-mixing time, i.e., the smallest t such that $\|P^{t'}v - \pi\|_{\mathrm{TV}} \leq 1/4$ for all v and $t' \geq t$.

Proposition 3 (Conductance bound). *If P is an irreducible Markov chain on a graph $G = (\mathcal{V}, \mathcal{E})$ with stationary distribution π, then its mixing time*

$$\tau \geq \frac{1}{4\Phi(P)}.$$

Proof. We will show that there exists an $\mathcal{S} \subset \mathcal{V}$ such that $\|P^t\pi_{\mathcal{S}} - \pi\|_{\mathrm{TV}} \geq \frac{1}{2} - t\Phi(P) > \frac{1}{4}$ if $t < 1/(4\Phi(P))$, which is equivalent to $\tau \geq 1/(4\Phi(P))$. Thereto, let $\mathcal{S} \subset \mathcal{V}$ be such that $\pi(\mathcal{S}) \leq 1/2$, and define the marginalizing or "coarse-graining" function $f_{\mathcal{S}} : \mathbb{R}^{\mathcal{V}} \to \mathbb{R}^2$ by

$$f_{\mathcal{S}}[w] = [w(\mathcal{S}), w(\mathcal{S}^c)].$$

Now note that the TV-distance is contractive under coarse-graining:

$$\begin{aligned}\|f_{\mathcal{S}}[w]\|_{\mathrm{TV}} &= \frac{1}{2}(|w(\mathcal{S})| + |w(\mathcal{S}^c)|) \\ &\leq \frac{1}{2}\Big(\sum_{j\in\mathcal{S}} |w(j)| + \sum_{j\in\mathcal{S}^c} |w(j)|\Big) = \|w\|_{\mathrm{TV}}.\end{aligned}$$

Therefore we can bound

$$\begin{aligned}\|P^t\pi_{\mathcal{S}} - \pi\|_{\mathrm{TV}} &\geq \|f_{\mathcal{S}}[P^t\pi_{\mathcal{S}}] - f_{\mathcal{S}}[\pi]\|_{\mathrm{TV}} \\ &\geq \|f_{\mathcal{S}}[\pi] - f_{\mathcal{S}}[\pi_{\mathcal{S}}]\|_{\mathrm{TV}} - \|f_{\mathcal{S}}[\pi_{\mathcal{S}}] - f_{\mathcal{S}}[P^t\pi_{\mathcal{S}}]\|_{\mathrm{TV}},\end{aligned}$$

where we also used the triangle inequality. The first term

$$\begin{aligned}\|f_{\mathcal{S}}[\pi] - f_{\mathcal{S}}[\pi_{\mathcal{S}}]\|_{\mathrm{TV}} &= \frac{1}{2}(|\pi(\mathcal{S}) - \pi_{\mathcal{S}}(\mathcal{S})| + |\pi(\mathcal{S}^c) - \pi_{\mathcal{S}}(\mathcal{S}^c)|) \\ &= 1 - \pi(\mathcal{S}) \geq \frac{1}{2},\end{aligned}$$

by our assumption that $\pi(\mathcal{S}) \leq 1/2$. The second term we can bound

$$\begin{aligned}\|f_{\mathcal{S}}[\pi_{\mathcal{S}}] - f_{\mathcal{S}}[P^t\pi_{\mathcal{S}}]\|_{\mathrm{TV}} \\ &= \frac{1}{2}(|\pi_{\mathcal{S}}(\mathcal{S}) - (P^t\pi_{\mathcal{S}})(\mathcal{S})| + |\pi_{\mathcal{S}}(\mathcal{S}^c) - (P^t\pi_{\mathcal{S}})(\mathcal{S}^c)|) \\ &= (P^t\pi_{\mathcal{S}})(\mathcal{S}^c) \leq t\Phi(\mathcal{S}),\end{aligned}$$

by the former lemma. Combining these inequalities we find that $\|P^t\pi_{\mathcal{S}} - \pi\|_{\mathrm{TV}} \geq \frac{1}{2} - t\Phi(\mathcal{S})$. Maximizing over all sets $\mathcal{S}$ for which $\pi(\mathcal{S}) \leq 1/2$ indeed gives $\|P^t\pi_{\mathcal{S}} - \pi\|_{\mathrm{TV}} \geq \frac{1}{2} - t\Phi(P)$. This finalizes the proof. □

For the dumbbell graph $K_N - K_N$ we saw that $\Phi(P) \in \Theta(1/N^2)$, so that the bound becomes

$$\tau \in \Omega(N^2),$$

which is tight. Another interesting example is the tree graph.

Example 9 (Binary Tree). The complete binary tree $G = T_{2,k}$ of depth k, depicted in Figure 3.2, is commonly used to represent a data structure or event tree. This graph has $|\mathcal{V}| = 2^{k+1} - 1$ nodes, a diameter $2k$ and a total degree $d_{\mathrm{tot}} = 4(2^k - 1)$. From the trivial diameter bound we know that the random walk has a mixing time $\tau \geq k$. Using the cut depicted in the figure, and the bound (3.4) on the conductance, we can estimate the conductance

$$\Phi(P) \leq \frac{\pi(j)}{\pi(\mathcal{S})} \in O(1/2^k).$$

By the conductance bound this proves that the random walk mixing time $\tau \geq \frac{1}{4\Phi(P)} \in \Omega(2^k)$. This bound can be shown to be tight [5]: $\tau \in \Theta(2^k)$.

We see that the random walk mixing time is exponentially larger than the diameter of the graph. It is hence a natural question to ask whether a more complicated Markov chain exists that has a better mixing time. For a more general Markov chain P, obeying the graph locality and mixing to π, we see that the same bound on the conductance will hold: $\Phi(P) \leq \frac{\pi(j)}{\pi(\mathcal{S})} \in O(1/2^k)$. Therefore the bound on the mixing time carries over, and we see that in fact any Markov chain mixing to π has a mixing time $\tau \in \Omega(2^k)$.

We encountered a similar situation at the start of Section 1.4, introducing lifted Markov chains. Work by Gerencsér [7] has shown that any Markov chain on the cycle $\mathbb{Z}_N$ has an unsatisfactory mixing time $\Omega(N^2)$. We showed however how lifted Markov chains can circumvent this observation, allowing for an optimal mixing time $\Theta(N)$. We could similarly ask whether there exists a lifted Markov chain that improves the mixing time of simple Markov chains on the binary tree. Later on we will show that under the arguably natural condition of invariance there exists no such LMC. △

We see that the conductance bound is tight on both the dumbbell graph and the binary tree. It is however not always tight, as is seen for the random walk on the cycle $\mathbb{Z}_N$, which has a mixing time $\tau \in \Theta(N^2)$. The conductance $\Phi(P) \in \Theta(1/N)$ so that the conductance bound gives $\tau \in \Omega(N)$, which is off by a square. Surprisingly, this is the worst case: the conductance bound is tight up to a square! As a consequence, not only does it hold that if there is a small conductance cut, then the Markov chain will mix slowly, but it also holds that if there is no such cut, then the Markov chain

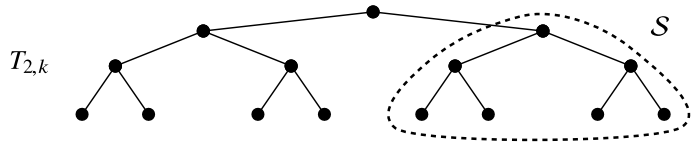


Fig. 3.2 Binary tree graph $T_{2,k}$ with depth $k = 3$. The cut $(\mathcal{S}, \mathcal{S}^c)$ that minimizes the random walk conductance $\Phi(P) \in O(1/2^k)$ is shown.

will mix fast. Proving such a converse result relies on entirely different techniques, going beyond the scope of this thesis.

Proposition 4 ([70]). *If P is an ergodic, reversible Markov chain with stationary distribution π, then*

$$\tau \in O\left(\frac{1}{\Phi(P)^2} \log \frac{1}{\min \pi(j)}\right).$$

Up to the log-factor this is indeed tight for the random walk on the cycle $\mathbb{Z}_N$. The log-factor is necessary, however, as is demonstrated by random walks on the aforementioned class of expander graphs. This is a remarkable class of bounded-degree graphs with an optimal conductance $\Phi(P) \in \Theta(1)$. In other words, they have no bad cuts. The diameter of any bounded-degree graph is in $\Omega(\log |\mathcal{V}|)$, so by the trivial diameter bound on the mixing time we know that $\tau \in \Omega(\log |\mathcal{V}|)$. A random walk on a bounded-degree graph has $\pi(j) \in \Theta(1/|\mathcal{V}|)$, so that the above converse bound leads to a tight estimate $\tau \in \Theta(\log |\mathcal{V}|)$.

Many proofs on the runtime of Markov chain algorithms rely crucially on the above lemma, which allows to bound the mixing time by showing that the graph has no bad cuts. Indeed this is a common strategy, allowing for elegant geometric arguments. See [5, 57, 70] for examples. For our purpose, we can use it for the following example. The example shows that in certain cases, contrarily to the tree graph, there do exist Markov chains that significantly improve the random walk mixing time, even up to the diameter bound.

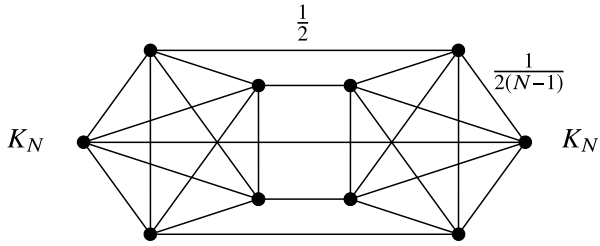


Fig. 3.3 Cartesian product graph $K_N \square K_N$, consisting of two copies of K_N with additional edges between all corresponding nodes. A random walk on $K_N \square K_N$ has a mixing time $\Theta(N)$, whereas the Markov chain defined by the depicted transition probabilities has a mixing time $\Theta(1)$.

Example 10 ($K_N \square K_N$). Consider the graph $K_N \square K_N$, where $\square$ denotes the Cartesian graph product, depicted in Figure 3.3. A random walk over this graph mixes

to the uniform distribution, and has a conductance $\Phi(P) \in \Theta(1/N)$. Therefore the random walk mixing time is bounded by $\tau \in \Omega(N)$, which can be shown to be tight. Again, this is quite unsatisfactory since the diameter of the graph is 2, hence we ask the question whether there exists another Markov chain that mixes more efficiently to the same stationary distribution.

Contrarily to the tree graph, we can indeed find such a Markov chain P', which nevertheless obeys the same locality as P and mixes to the same stationary distribution π. Thereto we simply increase the transition probability of the central edges from $1/N$ to $1/2$, and we symmetrically decrease the other transition probabilities accordingly to $1/(2(N-1))$, as is shown on Figure 3.3. By symmetry considerations, it is clear that the stationary distribution will still be the uniform distribution. The conductance of this Markov chain $\Phi(P') \in \Theta(1)$, so that by Proposition 4 the mixing time of this Markov chain $\tau \in O(\log N)$. With some more effort this can be shown to be $\tau \in \Theta(1)$. △

Chapter 4
Lifted Markov Chains Simulate Quantum Walks

We have already seen that adding memory to a random walk, giving rise to a lifted Markov chain, can speed up the mixing behavior. The central question answered in this section is whether quantum effects can accelerate mixing beyond what classical memory can do. This question is motivated from different angles:

- Some biological mechanisms have been observed to be more efficient than simple random walk dynamics. An example is the enhanced transport of excitations to reaction centers in photosynthetic systems [24]. It is an exciting and ongoing question whether this effect is caused by quantum effects. Our results elucidate whether an improved mixing behavior can be a signaler for quantum effects, or if this acceleration can also be caused by classical memory effects.
- Much ongoing research is devoted to the opportunities of building a quantum computer, clarifying which computational tasks can be accelerated or performed more efficiently than on a classical computer (see the "Quantum Algorithms Zoo" [71] for an updated database). Since Markov chain algorithms are a vital part of classical computing, they naturally raise the question of whether quantum computers can speed them up beyond what classical memory can do.

The main result in this chapter shows that for any quantum walk we can construct a lifted Markov chain that has the same mixing time. The size and design effort required for the construction render it an existence result, rather than a feasible acceleration speedup technique. However, as we discuss in for instance [72], on symmetric graphs such as lattices the LMC construction can be as simple as the quantum walk construction.

4.1 Quantum Walks as Local Stochastic Processes

In the following we will argue that Markov chains, lifted Markov chains, and quantum walks can in fact be captured by a generalizing class which we call *local stochastic processes*. We define this class using stochastic maps, where a *stochastic map* Γ over $\mathcal{V}$ is a linear map between probability distributions over $\mathcal{V}$. We can describe local stochastic processes by a family $\{\Gamma_t\}_{t\in\mathbb{N}}$ of stochastic maps, so that

the evolution over t time steps is given by

$$v \mapsto \Gamma_t[v].$$

A stochastic process is *local* with respect to a graph $G = (\mathcal{V}, \mathcal{E})$ if and only if for any subset $\mathcal{S} \subseteq \mathcal{V}$, probability distribution v over $\mathcal{V}$ and $t \geq 0$ it holds that

$$(\Gamma_{t+1}[v])(\mathcal{S}) \leq (\Gamma_t[v])(\mathcal{S}) + (\Gamma_t[v])(\partial\mathcal{S}^c), \tag{4.1}$$

where $\partial\mathcal{S}^c$ denotes the outer boundary of $\mathcal{S}$, i.e., the set of nodes outside of $\mathcal{S}$ that have an edge going to $\mathcal{S}$. We denote a local stochastic process by $\{\Gamma_t \sim G\}$. This definition of locality reflects the intuitive notion that the probability to be in a subset of the graph at time $t + 1$ has to be bounded by the probability of already being in that set at time t and the probability of being in the boundary of the set at time t. A family $\{\Gamma_t\}$ mixes to π if for all distributions v

$$\lim_{t\to\infty} \Gamma_t[v] = \pi.$$

Its ϵ-mixing time is defined as

$$\tau(\epsilon) = \min\{T \mid \max_v \|\Gamma_t[v] - \pi\|_{\mathrm{TV}} \leq \epsilon, \forall t \geq T\},$$

where the maximization again runs over all distributions over $\mathcal{V}$. We again call $\tau \equiv \tau(1/4)$ the mixing time of $\{\Gamma_t\}$. The family is called *invariant* if

$$\Gamma_t[\pi] = \pi.$$

We can represent the dynamics of a Markov chain P by setting $\Gamma_t[v] = P^t v$. Owing to the Markovianity of the evolution, the family has the semi-group property $\Gamma_{t+t'} = \Gamma_{t'} \circ \Gamma_t$ for all $t, t' \in \mathbb{N}$. A *lifted* Markov chain $\hat{P}$ with initialization map F and marginalizing map f, as introduced in Section 1.4, can be represented as a stochastic process by setting

$$\Gamma_t[v] = (f \circ \hat{P}^t \circ F)[v].$$

It is easy to check that the resulting stochastic processes are local if and only if the Markov chain P or the LMC $\hat{P}$ are local with respect to a graph G.

Towards representing a quantum walk as a stochastic process, recall the quantum mixing scheme that we introduced in Section 2.3, which maps initial probability distributions over $\mathcal{V}$ to the "induced probability distribution" over $\mathcal{V}$ of the quantum walk, defined by the outcome distribution of a hypothetical measurement. Given a general QW Ψ, with initialization map F and marginalization map f_q, this scheme defines a stochastic process by setting

$$\Gamma_t[v] = (f_q \circ \hat{\Psi}^t \circ F)[v].$$

Note that the proposed stochastic process only simulates the quantum walk for initial states which are classical distributions over the nodes, rather than quantum states. This is so because we will be interested in comparing mixing schemes whose input is restricted to classical distributions, allowing a fair comparison between quantum and classical schemes.

Besides (lifted) Markov chains and quantum walks, there still exists a wide range of mechanisms that give rise to stochastic processes. Examples of such are time-dependent processes such as simulated annealing [73] or polynomial filtering algorithms [74], and other graph exploring techniques such as self-avoiding walks [75], Cesaro mixing mechanisms [76] and depth- or breadth-first search [77].

4.2 Local Stochastic Processes as Lifted Markov Chains

Now consider a local stochastic process over a graph $G = (\mathcal{V}, \mathcal{E})$, described by a family $\{\Gamma_t\}$ of local stochastic maps. In the following we will show that for any time frame there exists an LMC, local with respect to G, that simulates this process. As a first step, we establish the existence and construction of *local stochastic bridges* for a given process, simulating the evolution of a stochastic process from some fixed initial state. With respect to an initial distribution v, such a bridge consists of an infinite set of local stochastic transition matrices $\{P_l^{(v)} \sim G\}$ such that

$$\Gamma_t[v] = \left(\Pi_{l=1}^{t} P_l^{(v)}\right) v, \quad \forall t \in \mathbb{N}.$$

Note the crucial dependency of the bridge on the initial distribution v. In general, $\Gamma_t[w] \neq \left(\Pi_{l=1}^{t} P_l^{(v)}\right) w$ if $w \neq v$. An exception is the case of Markov chains, for which $P_l^{(v)} = P$ for all v, l. Qualitatively, the existence of such a bridge connects the two notions of locality that we consider in this thesis, i.e., the zeros of a transition matrix and the intuitive notion expressed in (4.1).

Proposition 5 (Local Stochastic Bridges). *$\{\Gamma_t \sim G\}$ if and only if for all v and $t > 0$ there exists a stochastic bridge $P_t^{(v)} \sim G$ such that*

$$\Gamma_t[v] = P_t^{(v)} \Gamma_{t-1}[v].$$

Proof. The "if" direction is immediate by noting that condition (4.1), which is equivalent to $\{\Gamma_t \sim G\}$, is implied by the fact that $P_t^{(v)}(j, i) \neq 0$ for $j \in \mathcal{S}$ only if $i \in \mathcal{S}$ or $i \in \partial\mathcal{S}$.

To prove the "only if" direction we generalize a proof by Aaronson [58] from a hidden-variables context. He showed that for every quantum state $|\psi\rangle$ and unitary $U \sim G$ there exists a $P \sim G$ such that $v_{U|\psi\rangle} = P v_{|\psi\rangle}$, where $v_{|\chi\rangle}$ denotes the induced distribution of the quantum state $|\chi\rangle$ defined in (2.1). We generalize this result to general stochastic processes. To do so, call $y = \Gamma_{t-1}[v]$ and $z = \Gamma_t[v]$. In order to prove the above statement, it is convenient to resort to results concerning flows over capacitated networks [78]. In particular, we consider the weighted graph shown in Figure 4.1, where the edge weights represent the edge capacities. The network consists of a source node s, a sink node r, and two copies $\mathcal{W}$ and $\mathcal{W}'$ of the set of node states $\mathcal{V}$. The edges are defined as follows:

- node s is connected to any node $i \in \mathcal{W}$ with capacity $y(i)$,
- any node $i \in \mathcal{W}$ is connected to any node $i' \in \mathcal{W}'$ with capacity 1 if and only if $(i, i') \in \mathcal{E}$,
- any node $i' \in \mathcal{W}'$ is connected to node r with capacity $z(i')$.

The capacities $y(i)$ and $z(i')$, respectively from s and to r, thus reflect the probability distributions to be mapped. The key observation is the following: if this network can route a steady flow of value 1 from node s to node r, then the fraction from $i \in \mathcal{W}$ that is routed towards $i' \in \mathcal{W}'$ directly defines the entry $P_t^{(v)}(i', i)$ that we need. Indeed, to route a flow of value 1 from s to $\mathcal{W}$ (resp. $\mathcal{W}'$ to r) the outgoing edges from s (incoming edges from r) will have to be used to their full capacities $y(i)$

$(z(i'))$. By the central routing, the flow incoming in node i' is $\sum_{i\in\mathcal{V}} P_t^{(v)}(i',i)\, y(i)$. Since the incoming flow must equal the outgoing flow, this should equal $z(i')$ by our former argument, so that

$$\sum_{i\in\mathcal{V}} P_t^{(v)}(i',i)\, y(i) = z(i').$$

This implies that $P_t^{(v)} y = z$, as we claimed. In the below Figure 4.1 we give an example of this construction.

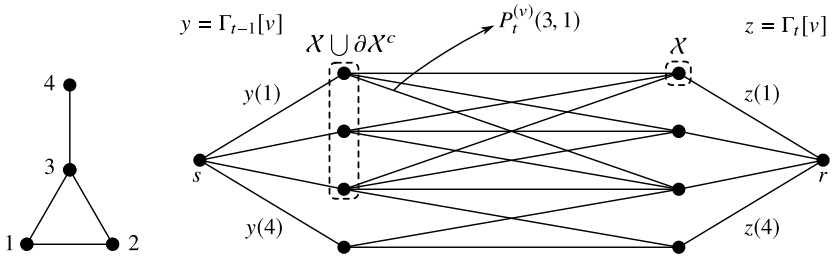


Fig. 4.1 (left) The "paw graph". (right) Illustration of the flow network construction between probability distributions $y = \Gamma_{t-1}[v]$ and $z = \Gamma_t[v]$.

To prove that a flow of value 1 exists, we invoke the max-flow min-cut theorem [78]. This classic theorem states that the maximum flow value that can be routed from node s to node r is equal to the minimum cut value of the graph, where a cut value is the sum of the capacities of a set of edges that disconnects s from r. Our proof thus comes down to showing that any cut has a value ≥ 1. It is clear that cutting all edges leaving s or arriving at r disconnects the graph, amounting to a cut value of 1. The minimum cut value will therefore be smaller than or equal to 1. Since cutting any middle edge between $\mathcal{W}$ and $\mathcal{W}'$ already gives a cut value ≥ 1, we know that any cut with a value < 1 cannot include any such edge. We can therefore restrict our further search to cuts consisting of outgoing edges from s and incoming edges from r. Thereto fix a subset $\mathcal{X} \subseteq \mathcal{W}'$, and assume that we cut the edges from its complement $\mathcal{X}^c$ to r. To block any flow from s to r, while keeping all middle edges, we must then cut the edges from s to all the $i \in \mathcal{W}$ which have an edge to $\mathcal{X}$. This corresponds to all the edges from s to the nodes in $\mathcal{W}$ which correspond to $\mathcal{X} \cup \partial\mathcal{X}^c$. The value of this cut is

$$z(\mathcal{X}^c) + y(\mathcal{X}) + y(\partial\mathcal{X}^c) = 1 - z(\mathcal{X}) + y(\mathcal{X}) + y(\partial\mathcal{X}^c).$$

Recalling that $y = \Gamma_{t-1}[v]$ and $z = \Gamma_t[v]$, the locality condition (4.1) imposes that

$$z(\mathcal{X}) \leq y(\mathcal{X}) + y(\partial\mathcal{X}^c).$$

From this it follows that the minimum value of the cut is exactly 1. This minimum is attained for example by setting $\mathcal{X} = \emptyset$ the empty set, corresponding to cutting all incoming edges of r. As a consequence, a solution $P_t^{(v)}$ to our problem exists. □

The existence of stochastic bridges being established, we can imagine the following transition rule, using the shorthand $P_t^{(i)} \equiv P_t^{(e_i)}$:

If $X_0 = i$ and the current state $X_t = j$, go to k with probability $P_{t+1}^{(i)}(k, j)$.

This transition rule simulates the original stochastic process. To see this, notice that if $X_0 = i$ then X_t is distributed according to $\left(\Pi_{l=1}^{t} P_l^{(v)}\right) e_i = \Gamma_t[e_i]$. By linearity, if X_0 is distributed according to some distribution v, then X_t will be distributed according to

$$\sum_i v(i) \left(\Pi_{l=1}^{t} P_l^{(v)}\right) e_i = \sum_i v(i) \Gamma_t[e_i] = \Gamma_t[v],$$

correctly simulating the original stochastic process.

The rule however is both time inhomogeneous and non-Markovian, depending on t as well as on the initial state X_0. For any finite time window $[0, T]$ however we can implement this rule with a lifted Markov chain. Thereto we lift the graph G so that the state of a walker not only consists of its current position $j \in \mathcal{V}$, but also holds its initial position $i = X_0$ and a time index or "clock" $t \in [0, T]$ (this is reminiscent of the "clock Hamiltonians" used by Feynman [59] and Kitaev [60]), resulting in the lifted node set

$$\hat{\mathcal{V}} = \mathcal{V} \times \{0, 1, \ldots, T\} \times \mathcal{V},$$

where the last register contains the current position of the walker. The first register containing X_0 will remain static during the walk, whereas the second register will be augmented with every time step. Now it is easy to construct an LMC $\hat{P}_T$ which implements the above transition rule:

$$\hat{P}_T = \sum_{j \in \mathcal{V}} \sum_{t=0}^{T-1} \Pi_j \otimes e_{t+1} e_t^\dagger \otimes P_{t+1}^{(j)} + \Pi_T, \tag{4.2}$$

where $\Pi_j = e_j e_j^\dagger$, and the last term $\Pi_T = I \otimes e_T e_T^\dagger \otimes I$ is a dummy term simply keeping the walk stationary after T steps. The initialization map F and marginalization map f associated to this LMC are trivially defined by

$$F[e_i] = e_i \otimes e_0 \otimes e_k, \quad f[e_i \otimes e_t \otimes e_j] = e_j, \quad \forall i, j \in \mathcal{V},\ t \in [0, T].$$

It is easy to check that this LMC effectively implements the proposed transition rule, mapping for instance

$$\begin{aligned} F(e_i) = e_i \otimes e_0 \otimes e_i &\mapsto \hat{P}_T(e_i \otimes e_0 \otimes e_i) = e_i \otimes e_1 \otimes (P_1^{(i)} e_i) \\ &\mapsto \hat{P}_T^2(e_i \otimes e_0 \otimes e_i) = e_i \otimes e_2 \otimes (P_2^{(i)} P_1^{(i)} e_i) \\ &\mapsto \ldots \end{aligned}$$

Finally we find the following proposition.

Proposition 6. *The LMC* (4.2) *simulates the stochastic process* $\{\Gamma_t\}$ *for* T *time steps. That is, for all* $t \in [0, T]$ *it holds that*

$$f \circ \hat{P}_T^t \circ F = \Gamma_t.$$

This shows that indeed local stochastic processes, and in particular quantum walks, can be simulated by LMCs for any finite time frame.

4.3 ... with the same Mixing Time

The LMC construction in (4.2) simulates a local stochastic process for a finite time window $[0, T]$. If we choose for instance $T = \tau^\Gamma$, with τ^Γ the mixing time of the original stochastic process, then clearly the mixing time of the LMC will also be τ^Γ. In general however we are interested in the long term behavior of the dynamics, as is reflected by the ϵ-mixing time for asymptotically small ϵ. By the amplification lemma 1 we know that for simple Markov chains it holds that $\tau(\epsilon) \leq \tau\lceil\log(1/\epsilon)\rceil$, hence the mixing time τ correctly characterizes the asymptotic mixing behavior. For more involved schemes such as LMCs this inequality does not hold as a rule. This is clearly demonstrated by our LMC construction (4.2) from the previous section: it simulates the original stochastic process for T steps, after which it stands still. As a consequence, if $T = \tau^\Gamma$ then indeed its mixing time $\tau = \tau^\Gamma$. However, after these first T steps it will stand still and not converge any further, so that its ϵ-mixing time $\tau(\epsilon)$ will in general be infinity for all $\epsilon < 1/4$.

In this section we show that if the original stochastic process is invariant, $\Gamma_t[\pi] = \pi$, then we can remedy this. Indeed in this case we can build an LMC whose ϵ-mixing time is $\tau(\epsilon) \leq \tau^\Gamma\lceil\log(1/\epsilon)\rceil$. Thereto we will first modify the original stochastic process into a "pseudo-Markov process" with the same ϵ-mixing time, critically relying on the invariance of the stochastic process. Then we will show that we can modify the LMC simulator (4.2) from the previous section into another LMC that simulates this modified stochastic process for all time, thus having the same ϵ-mixing time.

We call a stochastic process $\{\Gamma_t\}$ a *Markov process* if

$$\Gamma_t = \underbrace{\Gamma \circ \Gamma \circ \cdots \circ \Gamma}_{t \text{ times}} \equiv \Gamma^t.$$

This implies that the stochastic process is of the form $\Gamma_t = P^t = \Gamma^t$, with the Markov chain P defined by $P \equiv \Gamma$. We call a stochastic process $\{\Gamma_t\}$ a *pseudo-Markov process over $T > 0$ steps* if

$$\Gamma_t = \Gamma_{t \bmod T} \circ \Gamma_T^{\lfloor t/T \rfloor}.$$

This essentially means that the stochastic process is "Markovian over T time steps", i.e., the stroboscopic stochastic process $\{\Gamma_{t \cdot T} = \Gamma_T^t\}$ is Markovian. An example of such a process is a periodic time-inhomogeneous Markov process, such as a polynomial filter, described in work by the author [74]. Such process periodically cycles through a finite set of transition matrices $\{P_l\}_{l=1}^T$:

$$\Gamma_t = P_{t \bmod T} \ldots P_1 \left(P_T \ldots P_1\right)^{\lfloor t/T \rfloor}.$$

The following lemma shows that if a stochastic process is invariant, then we can always modify it into a pseudo-Markov process whose ϵ-mixing time is characterized by the original mixing time. This property will prove crucial for resolving the aforementioned mixing time issue of our LMC construction (4.2).

Lemma 4. *Let $\{\Gamma_t\}$ be an invariant stochastic process that mixes to π with a mixing time $\tau^\Gamma(\epsilon)$ for all $\epsilon > 0$. Let $\epsilon_0 < 1/2$ and $T = \tau^\Gamma(\epsilon_0)$. Then the pseudo-Markov process $\{\widetilde{\Gamma}_t\}$ defined by*

$$\widetilde{\Gamma}_t = \Gamma_{t \bmod T} \circ \Gamma_T^{\lfloor t/T \rfloor}$$

has an ϵ-mixing time

$$\tau(\epsilon) \leq \tau^{\Gamma}(\epsilon_0) \left\lceil \frac{\log \frac{1}{\epsilon}}{\log \frac{1}{2\epsilon_0}} \right\rceil, \quad \forall \epsilon > 0.$$

The proof of the lemma runs along the exact same lines as the proof of the amplification lemma 1, so we will omit it. For $\epsilon_0 = 1/4$ this gives a mixing time $\tau(\epsilon) \leq \tau^{\Gamma} \cdot \lceil \log \frac{1}{\epsilon} \rceil$, which is the same upper bound that can be derived for the original process. Under quite natural condition (e.g., uniform contraction rate) this upper bound will be tight, possibly up to log-factors, so that the stochastic process and the pseudo-Markov process effectively have the same ϵ-mixing time.

The key observation now is that we can simulate any pseudo-Markov process with an LMC for an indefinite number of steps. Thereto, let $\{\Gamma_t\}$ be a pseudo-Markov process over T steps, and consider its LMC simulator $\hat{P}_T$ (4.2) defined in previous section. By the fact that the stochastic process is pseudo-Markov, we can marginalize and reinitialize the simulator every T steps: since $f \circ \hat{P}^t \circ F = \Gamma_t$ for all $t \in [0, T]$, it holds that

$$\Gamma_t = \Gamma_{t \bmod T} \circ \Gamma_T^{\lfloor t/T \rfloor} = (f \circ \hat{P}^{t \bmod T} \circ F) \circ (f \circ \hat{P}^T \circ F)^{\lfloor t/T \rfloor}.$$

We can use the clock register of the LMC to encode this, basically by implementing the extra step

$$Q = (F \circ f)\Pi_T + (I - \Pi_T),$$

where Π_T projects on the clock state equaling e_T. If need be, we can rewrite $F \circ f$ in elementary terms. Thereto recall that $f[e_i \otimes e_t \otimes e_j] = e_j$ and $F[e_i] = e_i \otimes e_0 \otimes e_k$, so that

$$(F \circ f)[e_i \otimes e_t \otimes e_j] = e_j \otimes e_0 \otimes e_j, \quad \forall i, j \in \mathcal{V},\ t \in [0, T].$$

We can write this as

$$F \circ f = \sum_{i,j \in \mathcal{V}} \sum_{t=0}^{T} e_j e_i^{\dagger} \otimes e_0 e_t^{\dagger} \otimes e_j e_j^{\dagger}.$$

With $\hat{P}_T$ defined in (4.2), this results in the LMC simulator

$$\hat{P} = Q\hat{P}_T. \tag{4.3}$$

As the main result of this section we find the following proposition.

Proposition 7. *If $\{\Gamma_t\}$ is a pseudo-Markov process over T steps, then the LMC $\hat{P}$ simulates it for all time. That is, for all $t \geq 0$ it holds that*

$$f \circ \hat{P}^t \circ F = \Gamma_t.$$

From Lemma 4 we know that to any invariant stochastic process we can associate a pseudo-Markov process with a closely related ϵ-mixing time. Combined with the above proposition, this implies that we can construct an LMC with the same ϵ-mixing time, as formulated in the following theorem.

Theorem 3. *Let* $\{\Gamma_t\}$ *be an invariant stochastic process that mixes to* π *with a mixing time* $\tau^\Gamma(\epsilon)$ *for all* $\epsilon > 0$. *Let* $\epsilon_0 < 1/2$ *and* $T = \tau^\Gamma(\epsilon_0)$. *Then the LMC* $\hat{P} = Q\hat{P}_T$ *has an* ϵ*-mixing time*

$$\tau(\epsilon) \leq \tau^\Gamma(\epsilon_0) \cdot \left\lceil \frac{\log \frac{1}{\epsilon}}{\log \frac{1}{2\epsilon_0}} \right\rceil, \quad \forall \epsilon > 0.$$

This is the main result of this chapter. At the beginning of the chapter we raised the question of whether an observed speedup in mixing of some physical system can be diagnostic for quantum effects. By the above results we see that this is not the case, as the same speedup can always be achieved by a classical walking particle with memory. Even stronger, we see that the entire family of invariant stochastic processes cannot improve mixing beyond what LMCs can do.

This result should be nuanced by the fact that the LMC construction is very extensive in memory and design effort. The size of the local memory equals $\tau^\Gamma(\epsilon_0)|\mathcal{V}|$, and the construction of the stochastic bridges requires solving flow problems over the full graph. In subsequent work, Dervovic [79] showed that the full construction of the LMC requires a number of steps that is polynomial in $|\mathcal{V}|$. From an algorithmic perspective, such time and memory space costs are generally infeasible. For comparison, a MCMC algorithm is deemed efficient only if its construction and resources scale polynomially in $\log|\mathcal{V}|$, rather than $|\mathcal{V}|$. This open end creates room for a quantum advantage. Specifically, a quantum advantage can be found by answering the following two open questions:

1. *Is there an efficient QW construction that speeds up mixing?*
   A random walk can be implemented efficiently if we are merely given an initial node of a graph and access to a "neighborhood oracle" (which returns the neighbors of the node that you are at). Under these conditions, can we construct and implement a QW that accelerates the random walk mixing time? Currently the existing examples of similarly efficient quantum walk constructions for mixing are restricted to highly symmetric graphs such as lattices [15, 39, 49], yet on such graphs there also exist efficient LMC constructions [8, 9, 72].
2. *Is there an efficient LMC construction with the same mixing time as a QW?*
   That is, can LMCs simulate the mixing behavior of QWs with a memory size polynomial in $\log|\mathcal{V}|$?

In Appendix A we show that if the condition of invariance is dropped (which is a running assumption throughout this part) then there exist examples where efficient QW constructions exist, such as the binary tree graph, yet no efficient LMC constructions are known.

Chapter 5
Conductance Bound for Quantum Walks

Undoubtedly, quantum walk algorithms have been the most successful in accelerating search problems. Examples of such are the collision problem [4] for finding two equal elements among a list, and the problem of searching solutions in a decision tree [13]. A range of algorithms and settings have led to quantum walk speedups for search problems that range from quadratic [4, 18, 22] to exponential [13, 35, 36, 80].

Much less is known about quantum walk algorithms for mixing, to some extent the dual task to search. Our running example of the cycle graph $\mathbb{Z}_N$ shows that quantum walks can quadratically accelerate the random walk mixing time from $\Theta(N^2)$ to $\Theta(N)$. For the cycle this is optimal, as its diameter is $\lfloor N/2 \rfloor$ and the proof of the diameter bound in Proposition 2 easily generalizes to quantum walks. Alternatively, recall the binary tree $T_{2,k}$ from example 9 in Section 3.2. The diameter of this graph equals $2k$, yet the random walk mixing time is $\Theta(2^k)$. Can quantum walks quadratically improve the mixing time to $\Theta(\sqrt{2^k})$? Can they improve it to the optimal diameter bound $\Theta(k)$?

In the coming chapter we show that, under the arguably natural condition of invariance, the speedup of quantum walks can be bounded using properties of the graph. As an example, it proves that on graphs of bounded degree, the QW speedup can be no more than quadratic. This extends an earlier observation by [15] to general quantum Markov chains. These insights follow from the main contribution of this chapter: a conductance bound for quantum walks, and for stochastic processes in general. We achieve this bound by proving a conductance bound for the family of LMC simulators $\hat{P}$ defined in the previous chapter. Since for any stochastic process there exists such a simulator, this yields a conductance bound for stochastic processes, and for quantum walks in particular.

5.1 Conductance Bound for LMC Simulator

Let $\hat{P} \sim G$ denote the LMC simulator of a stochastic process $\{\Gamma_t \sim G\}$, defined in (4.3). To prove a conductance bound on $\hat{P}$ we will make use of the *induced Markov chain* of a lifted Markov chain, as introduced in [63]. This Markov chain

roughly represents the projection of an LMC back onto the original graph. Given an irreducible LMC $\hat{P}$ on a lifted graph $\hat{G}$, note first that $\hat{P}$ necessarily has a unique stationary distribution $\hat{\pi} > 0$ (although the LMC might not converge to $\hat{\pi}$) and $f(\hat{\pi}) = \pi$, with π the original stationary distribution. For all $i, j \in \mathcal{V}$, the induced Markov chain $P_{\mathcal{V}}$ is then defined as

$$P_{\mathcal{V}}(j,i) = \sum_{l \in f^{-1}(j), k \in f^{-1}(i)} \frac{\hat{\pi}(k)}{\hat{\pi}(f^{-1}(i))} \hat{P}(l,k),$$

so the transition probability $P_{\mathcal{V}}(j,i)$ is a weighted mean of the LMC transition probabilities $\hat{P}(l,k)$ from the child nodes of i to the child nodes of j. Now note that because $\hat{P}$ is irreducible, also $P_{\mathcal{V}}$ will be irreducible, and it is easy to check that its unique stationary distribution will be π. We will need the following lemma, wherein we use the shorthand $\hat{\mathcal{S}} = f^{-1}(\mathcal{S})$:

Lemma 5. *For any subset $\mathcal{S} \subset \mathcal{V}$, the conductance $\Phi_{P_{\mathcal{V}}}(\mathcal{S})$ of the induced Markov chain $P_{\mathcal{V}}$ equals the conductance $\Phi_{\hat{P}}(\hat{\mathcal{S}})$ of the original LMC $\hat{P}$.*

Proof. Recall the definition of the conductance

$$\Phi_{P_{\mathcal{V}}}(\mathcal{S}) = \frac{Q_{\mathcal{V}}(\mathcal{S}, \mathcal{S}^c)}{\pi(\mathcal{S})},$$

with $Q_{\mathcal{V}}(\mathcal{S}, \mathcal{S}^c) = \sum_{j \in \mathcal{S}^c, i \in \mathcal{S}} P_{\mathcal{V}}(j,i)\pi(i)$ the ergodic flow from $\mathcal{S}$ to $\mathcal{S}^c$. We can prove that this ergodic flow equals the total ergodic flow of the LMC $\hat{P}$ from the set of child nodes $\hat{\mathcal{S}}$ to $\hat{\mathcal{S}}^c$. Thereto notice that

$$\begin{aligned} Q_{\mathcal{V}}(i,j) = P_{\mathcal{V}}(j,i)\pi(i) &= \sum_{l \in f^{-1}(j), k \in f^{-1}(i)} \frac{\hat{\pi}(k)}{\hat{\pi}(f^{-1}(i))} \hat{P}(l,k)\hat{\pi}(f^{-1}(i)) \\ &= \sum_{l \in f^{-1}(j), k \in f^{-1}(i)} \hat{Q}(k,l) = \hat{Q}(f^{-1}(i), f^{-1}(j)), \end{aligned}$$

from which we immediately find that $Q_{\mathcal{V}}(\mathcal{S}, \mathcal{S}^c) = \hat{Q}(\hat{\mathcal{S}}, \hat{\mathcal{S}}^c)$. Combined with the fact that $\pi(\mathcal{S}) = \hat{\pi}(\hat{\mathcal{S}})$, we indeed find that

$$\Phi_{P_{\mathcal{V}}}(\mathcal{S}) = \frac{Q_{\mathcal{V}}(\mathcal{S}, \mathcal{S}^c)}{\pi(\mathcal{S})} = \frac{\hat{Q}(\hat{\mathcal{S}}, \hat{\mathcal{S}}^c)}{\hat{\pi}(\hat{\mathcal{S}})} = \Phi_{\hat{P}}(\hat{\mathcal{S}}). \qquad \square$$

An interesting consequence is that given a graph G, a stationary distribution π and a cut $(\mathcal{S}, \mathcal{S}^c)$, an LMC cannot improve the conductance over the best simple Markov chain:

$$\Phi_{\hat{P}}(\hat{\mathcal{S}}) = \Phi_{P_{\mathcal{V}}}(\mathcal{S}) \leq \max_{P \sim G : P\pi = \pi} \Phi_P(\mathcal{S}).$$

This captures the intuition that if there is a bad cut in the original graph, there will still be a bad cut in the lifted graph. Motivated by this, we define the *graph conductance* $\Phi_{G,\pi}$

$$\Phi_{G,\pi} = \max_{P \sim G : P\pi = \pi} \Phi(P),$$

as the conductance of the best simple Markov chain. From the above lemma we know that also $\Phi_{G,\pi} = \max_{\hat{P} \sim \hat{G} : \hat{P}\hat{\pi} = \hat{\pi}} \Phi_{\hat{P}}$, maximizing over all LMCs with a

stationary distribution $\hat{\pi}$ such that $f(\hat{\pi}) = \pi$. Using the bound on $\Phi(P)$ from (3.4), we can prove the useful estimate

$$\Phi_{G,\pi} \leq \min_{\pi(\mathcal{S}) \leq 1/2} \frac{\pi(\partial \mathcal{S})}{\pi(\mathcal{S})}. \tag{5.1}$$

We can now prove the following proposition, which bounds the mixing time of the LMC simulator in terms of the graph conductance.

Proposition 8. *The mixing time τ of the LMC simulator $\hat{P} = Q\hat{P}_T$ of an invariant pseudo-Markov process over T steps, defined in* (4.3)*, satisfies the inequality*

$$\tau + T \geq \frac{1}{4\Phi_{P_{\mathcal{V}}}} \geq \frac{1}{4\Phi_{G,\pi}}.$$

Proof. We will prove this statement by showing that (i) the LMC mixing scheme $(f \circ \hat{P}^t)$ without initialization map F, starting from general initial distributions over $\hat{\mathcal{V}}$, adds at most T steps to the LMC mixing scheme $(f \circ \hat{P}^t \circ F)$ with F, and (ii) the mixing scheme without F has a mixing time at least $1/(4\Phi_{P_{\mathcal{V}}})$.

We first prove the first point. By definition of the mixing time we know that for initialized distributions over $\hat{\mathcal{V}}$ of the form $F[v]$, with v some distribution over $\mathcal{V}$, it must hold that

$$\left\| (f \circ \hat{P}^{\tau}) F[v] - \pi \right\|_{\mathrm{TV}} = \left\| (f \circ \hat{P}^{\tau} \circ F)[v] - \pi \right\|_{\mathrm{TV}} \leq \frac{1}{4}. \tag{5.2}$$

By the structure of the LMC, we can use this to prove that for general initial distributions w over $\hat{\mathcal{V}}$ it holds that

$$\left\| (f \circ \hat{P}^{\tau+T}) w - \pi \right\|_{\mathrm{TV}} \leq \frac{1}{4}.$$

Thereto, first consider an initial distribution $w = e_v \otimes e_t \otimes e_{v'}$, localized on an arbitrary node of $\hat{\mathcal{V}}$. By construction of $\hat{P}$, it holds that after exactly $T - t$ steps of the LMC this state will be "reinitialized" by the operator Q, so that $\hat{P}^{T-t} = F[v']$ for some v'. As a consequence, for all $t' \geq \tau$ it holds that

$$\left\| (f \circ \hat{P}^{t'+T-t}) w - \pi \right\|_{\mathrm{TV}} = \left\| (f \circ \hat{P}^{t'}) F[v'] - \pi \right\|_{\mathrm{TV}} \leq \frac{1}{4},$$

where the last inequality follows from (5.2). Since $T - t \leq T$, this shows that $\left\| (f \circ \hat{P}^{t'+T}) w - \pi \right\|_{\mathrm{TV}} \leq 1/4$ for all initial distributions $w = e_v \otimes e_t \otimes e_{v'}$. For a general w over $\hat{\mathcal{V}}$, we can simply apply the triangle inequality to prove that indeed $\left\| (f \circ \hat{P}^{t'+T}) w - \pi \right\|_{\mathrm{TV}} \leq 1/4$. This finishes the first part of the proof.

Now we prove the second point. We can bound the mixing time of the LMC without initialization map by considering the initial state $\hat{\pi}_{\hat{\mathcal{S}}}$. By Lemma 3 in Chapter 3 we know that

$$(\hat{P}^t \hat{\pi}_{\hat{\mathcal{S}}})(\hat{\mathcal{S}}^c) \leq t\Phi_{\hat{P}}(\hat{\mathcal{S}}) = t\Phi_{P_{\mathcal{V}}}(\mathcal{S}),$$

with the last equality following from Lemma 5. We can hence bound the mixing time similarly to the proof of 3. Thereto recall the function $f_{\mathcal{S}}(w) = (w(\mathcal{S}), w(\mathcal{S}^c))$, and the fact that the total variation distance is contractive under coarse-graining, so that

$$\begin{aligned}\|f(\hat{P}^t\hat{\pi}_{\hat{\mathcal{S}}}) - \pi\|_{\mathrm{TV}} &\\ &\geq \|(f_{\mathcal{S}} \circ f)(\hat{P}^t\hat{\pi}_{\hat{\mathcal{S}}}) - f_{\mathcal{S}}(\pi)\|_{\mathrm{TV}} \\ &\geq \|f_{\mathcal{S}}(\pi) - f_{\mathcal{S}}(\pi_{\mathcal{S}})\|_{\mathrm{TV}} - \|f_{\mathcal{S}}(\pi_{\mathcal{S}}) - (f_{\mathcal{S}} \circ f)(\hat{P}^t\hat{\pi}_{\hat{\mathcal{S}}})\|_{\mathrm{TV}}.\end{aligned}$$

We can again bound the first term $\|f_{\mathcal{S}}(\pi) - f_{\mathcal{S}}(\pi_{\mathcal{S}})\|_{\mathrm{TV}} = 1 - \pi(\mathcal{S}) \geq 1/2$, and the second term $\|f_{\mathcal{S}}(\pi_{\mathcal{S}}) - (f_{\mathcal{S}} \circ f)(\hat{P}^t\hat{\pi}_{\hat{\mathcal{S}}})\|_{\mathrm{TV}} = (\hat{P}^t\hat{\pi}_{\hat{\mathcal{S}}})(\hat{\mathcal{S}}^c) \leq t\Phi_{P_{\mathcal{V}}}(\mathcal{S})$. Combining these inequalities gives

$$\|f(\hat{P}^t\hat{\pi}_{\hat{\mathcal{S}}}) - \pi\|_{\mathrm{TV}} \geq \frac{1}{2} - t\Phi_{P_{\mathcal{V}}}(\mathcal{S}) \geq \frac{1}{2} - t\Phi_{P_{\mathcal{V}}}.$$

This implies that the mixing time of the LMC without initialization map is lower bounded by $1/(4\Phi_{P_{\mathcal{V}}})$, and so the LMC with initialization has a mixing time

$$\tau \geq \frac{1}{4\Phi_{P_{\mathcal{V}}}} - T \geq \frac{1}{4\Phi_{G,\pi}} - T. \qquad \square$$

5.2 Conductance Bound for Local Stochastic Processes

As an almost direct consequence we find a conductance bound for the original stochastic process, which is the main contribution of this chapter and, to some extent, this part of the thesis.

Theorem 4 (Conductance bound). *Let $\{\Gamma_t \sim G\}$ represent an invariant stochastic process that mixes to π. Then its mixing time*

$$\tau^{\Gamma} \geq \frac{1}{12\Phi_{G,\pi}}.$$

This bound is tight up to a log-factor: there exists an invariant stochastic process that has a mixing time

$$\tau \in O\left(\frac{1}{\Phi_{G,\pi}} \log\left(\frac{1}{\min_{i\in\mathcal{V}} \pi(i)}\right)\right).$$

Proof. The lower bound follows by setting $\epsilon = \epsilon_0 = 1/4$ and $T = \tau^{\Gamma}$ in Theorem 3, which shows that the LMC $\hat{P} = Q\hat{P}_{\tau^{\Gamma}}$ has a mixing time $\tau \leq 2\tau^{\Gamma}$. Combining this with Proposition 8, stating that $\tau \geq 1/(4\Phi_{G,\pi}) - \tau^{\Gamma}$, we find that $\tau^{\Gamma} \geq 1/(12\Phi_{G,\pi})$.

The upper bound follows from applying Theorem 2, proven in [9], to the Markov chain P that maximizes the conductance, $P = \mathrm{argmax}_{P\sim G: P\pi=\pi}\Phi_P$. $\square$

This shows that invariant stochastic processes, in all their generality, must obey a non-trivial conductance bound. At the start of this chapter we discussed the speedup range of quantum walks over random walks. On graphs of bounded degree we can now prove that quantum walks can at most quadratically speed up random walks on graphs, up to a log factor. This extends an observation in [15][1] from unitary quantum walks to general quantum walks.

[1] Their bound for unitary quantum walks is better by a log-factor, yet this is a consequence of a faulty assumption that random walks on bounded degree graphs have a mixing time $O(1/\Phi(P)^2)$. The correct bound is stated in (5.3), which is tight on expander graphs as we discussed below Proposition 4.

Corollary 2. *If τ is the mixing time of a random walk on a bounded degree graph, then any quantum walk mixing scheme has a mixing time $\Omega(\sqrt{\tau}/\log \frac{1}{|V|})$.*

Proof. With π the random walk stationary distribution and $\Phi(P)$ the random walk conductance, we can use the estimate (5.1) and the expression (3.3) to bound

$$\begin{aligned}\Phi_{G,\pi} &\leq \min_{\pi(\mathcal{S})\leq 1/2} \frac{\pi(\partial\mathcal{S})}{\pi(\mathcal{S})} = \min_{\pi(\mathcal{S})\leq 1/2} \frac{d(\partial\mathcal{S})}{d(\mathcal{S})} \\ &\leq d_{\max} \min_{\pi(\mathcal{S})\leq 1/2} \frac{|E(\mathcal{S},\mathcal{S}^c)|}{d(\mathcal{S})} = d_{\max}\Phi(P).\end{aligned}$$

For graph of bounded degree it holds that $d_{\max} \in \Theta(1)$. Using $\Phi_{G,\pi} \geq \Phi(P)$, we see that $\Phi_{G,\pi} \in \Theta(\Phi(P))$. As a consequence, the bound in Theorem 4 shows that a quantum walk mixing scheme has a mixing time $\tau \in \Omega(1/\Phi(P))$. If we combine this with the upper bound on the random walk mixing time from Proposition 4,

$$\tau^{\mathrm{RW}} \in O\left(\frac{1}{\Phi(P)^2} \log \frac{1}{\min \pi(j)}\right), \tag{5.3}$$

we find the statement in the proposition. □

This bound cannot be attained in general, as follows from the below example.

Example 11 (Binary tree). At the start of the chapter we asked whether quantum walks could speed up the mixing time on the binary tree, maybe even up to the diameter bound. Directly applying Theorem 4 to a quantum walk on $T_{2,k}$ allows to prove that this is not the case. In Example 9 we already argued that the graph conductance for the binary tree $T_{2,k}$ is bounded by

$$\Phi_{G,\pi} \in \Theta(1/2^k),$$

with π the stationary distribution of the random walk. From our Theorem 4 we see that the mixing time of any invariant stochastic process on $T_{2,k}$ is therefore $\tau \in \Omega(2^k)$, which equals the random walk mixing time. Hence no invariant stochastic process, and in particular no LMC or quantum walk, can speed up the random walk mixing time. △

Contrary to the bounded degree cases, there is no limit on how much random walks on graphs of unbounded degree can be accelerated, as we saw earlier in Example 10. There we considered the graph $K_N \square K_N$, on which a random walk has a mixing time $\tau \in \Theta(N)$. We showed how already choosing a better Markov chain allows to improve this mixing time to $\tau \in \Theta(1)$.

5.3 Applications

The main contribution of this chapter is the fact that invariant stochastic processes, and in particular quantum walks, have a mixing time $\tau \in \Omega(1/\Phi_{G,\pi})$. In the following we discuss some applications.

Time-Inhomogeneous and Imprecise Markov Chains

A time-inhomogeneous Markov chain is described by a family of transition matrices $\{P_t\}_{t\in\mathbb{N}}$, so that X_t is distributed according to

$$v_t = P_t P_{t-1} \dots P_1 v_0.$$

If this chain mixes to π, and for each P_t it holds that $P_t\pi = \pi$, then this describes an invariant stochastic process. Examples of such are given in [74, 81, 82], and in [83, 84] for card shuffling. In these papers the difficulty of analyzing such time-inhomogeneous chains is stated explicitly. By our result we see directly that its mixing time is bounded by $\tau \geq 1/(12\Phi_{G,\pi})$. This shows that the gain of such time-varying strategies is limited, and can not improve the mixing time beyond what for instance lifted Markov chains or quantum walks can do.

Related to this case is the setting of "imprecise Markov chains", see for instance [85]. Here a set of Markov chains $\{P_l\}$ is considered, reflecting uncertainty on the precise transition mechanism. All that is known is that at any time step one of the Markov chains of the set is applied. Under the condition that there exists some π such that $P_l\pi = \pi$ for all l, we can use our conductance bound to infer that the mixing time will be lower bounded by $1/(12\Phi_{G,\pi})$, where G is the "minimal" graph (having the least amount of edges) such that $P_l \sim G$ for all l.

It might also be interesting, yet we leave it as an open direction, to apply our results to recent work by Onorati et al [53] on quantum scrambling processes, in which the mixing properties of unitary time-varying quantum processes are considered.

Time-Averaged Processes

Consider the following stochastic process:

$$v_t = \Gamma_t[v_0] = \frac{1}{t}\sum_{l=0}^{t-1} P^l v_0,$$

for some Markov chain P. We call v_t the *Cesaro average* of P, and it is sometimes analyzed for periodic Markov chains that otherwise do not converge (or unitary quantum walks, see below). The Cesaro average is a special case of the family of time-averaged processes (or *blind stopping rules*), as discussed in for instance [76]. A distribution v_t resulting from a time-averaged process corresponds to the output of a Markov chain run for a number of steps distributed according to some distribution over $[0, t]$. In certain cases, time-averaging can speed up the mixing time over a simple Markov chain. Consider for instance the simple shift $P = P^{\uparrow}$ on $\mathbb{Z}_N$. It is not difficult to show that the Cesaro average of $P^{\uparrow}$ will converge to the uniform distribution in $O(N)$ steps, which is quadratically faster than any simple Markov chain. By our conductance bound, we easily see that time-averaging cannot improve the mixing time beyond $\Theta(1/\Phi_{G,\pi})$, which is satisfied for instance on the cycle.

We already mentioned earlier that also in the quantum walk literature it is common to consider the Cesaro average of the output distribution. This is a way of dealing with the quasi-periodicity of a unitary quantum walk, whose output distribution might not converge, yet its Cesaro average will. We see that our LMC simulator and conductance bound also hold when considering the mixing time of this Cesaro average.

Finite-Time Convergence

Consider the following algebraic problem, related to finite-time convergence [86], sets of indecomposable and aperiodic matrices [87] and the inverse eigenvalue problem [88]:

> *What is the minimal number of symmetric stochastic matrices over a graph G whose product has rank one?*

From Theorem 4 it follows that this number is bounded by $1/(12\Phi_{G,\pi})$ with π the uniform distribution. To see this, note that any set of symmetric, stochastic matrices $\{P_l, 1 \leq l \leq T\}$ defines a time-inhomogeneous Markov chain that leaves the uniform distribution π invariant. Now assume that the product $P_T \dots P_1$ has rank one. Since $P_T \dots P_1$ is stochastic, and $P_T \dots P_1 \pi = \pi$, we find that for all distributions v it necessarily holds that

$$P_T \dots P_1 v = \pi,$$

and so the mixing time of this time-inhomogeneous Markov chain is $\tau \leq T$. By Theorem 4 this implies that $T \geq 1/(12\Phi_{G,\pi})$.

Search Algorithms

We briefly introduce search problems and algorithms, and show how we can apply our conductance bound to them. In the second part of this thesis we will treat these problems and algorithms in more detail.

Consider a graph $G = (\mathcal{V}, \mathcal{E})$ and a "marked node" $z \in \mathcal{V}$ that we are looking for, for example representing a solution to some problem. A common strategy to find z is to run a random walk or Markov chain P over G, starting from some initial probability distribution over $\mathcal{V}$, until we encounter z. We can cast this problem as a mixing problem by adapting P into an *interpolated Markov chain* P_s, where we add a strong self loop of strength s to the marked node z:

$$P_s(z, z) = s, \quad P_s(j \neq z, z) = (1 - s)P(j, z),$$

and $P_s(j, i) = P(j, i)$ elsewhere. The interpolated Markov chain is shown in Figure 5.1. It is proven in [22] that we can choose $1 - s \in \Theta(\pi(z))$ such that the stationary distribution π_s of P_s concentrates on z: $\pi_s(z) \in \Theta(1)$. Therefore we can run this Markov chain until it converges to π_s, which will return z with constant

probability. We can then repeat this scheme in order to find z with probability arbitrarily close to 1.

Our conductance bound allows to prove a lower bound on the runtime of this scheme, and in fact any invariant stochastic process that can be used to accelerate P_s.

Proposition 9. *Any invariant stochastic process that mixes to π_s has a mixing time $\tau \in \Omega\big(1/\pi(z)\big)$.*

Proof. Using our conductance bound, we can prove the statement by showing that $\Phi_{G,\pi_s} \in O(\pi(z))$. Thereto we set $\mathcal{S} = \{z\}$. With respect to a general $P \sim G$ with $P\pi_s = \pi_s$, it holds that

$$\Phi_P(\mathcal{S}) = \sum_{j \in \mathcal{S}^c} \frac{P(j,z)\pi_s(z)}{\pi_s(z)} = P(\mathcal{S}^c, z) \leq s = \pi(z).$$

This bound is independent of P and hence $\Phi_{G,\pi_s} \in O(\pi(z))$, finally showing that $\tau \in \Omega\big(1/\pi(z)\big)$. □

We retrieve the classic bound $\Omega(1/\pi(z))$ on the search time of an element z using any Markov chain P. In addition we find that any more advanced quantum or LMC scheme that mixes to π_s must also obey this bound. In the second part of this thesis we discuss more involved quantum walk algorithms that do break this bound, and must therefore also break invariance.

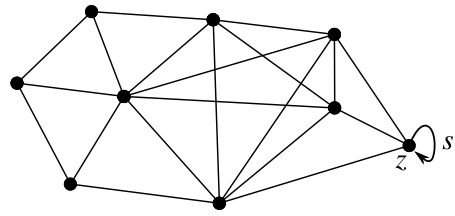


Fig. 5.1 Interpolated Markov chain P_s, adding a self-loop of strength s to the marked element z. If $1 - s \in \Theta(\pi(z))$ then the stationary distribution π_s of P_s has a large overlap with z: $\pi_s(z) = 1/2$. The conductance bound in Theorem 4 proves that any invariant scheme mixing to π_s has a mixing time $\Omega(1/\pi(z))$.

Chapter 6
Observations and Outlook

In this chapter we compare our results to some existing work relevant to this thesis part. We also provide some further observations that relate to our results, and we discuss remaining open questions.

6.1 Comparison to Existing Work

Simulation of Quantum Walks. In Chapter 4 we demonstrated how lifted Markov chains can be used to simulate quantum walks.

In a different line of work, as presented in [89–92], the classical simulation of quantum walks on lattices with light waves is discussed. They show how for instance the polarization and frequency of a classical light wave can be used to encode the coin and position, respectively, of a quantum walk on the cycle. The interference effects of classical optics then allow to simulate the dynamics of quantum walks on lattices. The main conclusion of these papers is that "the ingredients of QWs, namely, superposition, interference, and, indeed, a form of entanglement, are also present in classical optics" (quoted from [89]). This seems to point to superposition and interference as the source of the quantum advantage. Our results, in contrast, show that the QW behavior can also be obtained from dynamics that do not exhibit any such features: classical stochastic processes with memory, lacking these wave-like features of classical optics. In addition, the aforementioned papers consider the simulation of QWs using analog classical optics, whereas we use lifted Markov chains, an existing class of models in digital probabilistic computation. Indeed, LMCs were originally proposed to improve the performance of Markov chain Monte Carlo and similar methods on graphs. Since QWs have also been motivated by the same tasks, our result allows for a more direct comparison. It is also this fact that allowed to prove the conductance bounds from Theorem 4. To the best of the author's knowledge, the classical optics framework does not allow for any result of this kind. Finally, from the current literature it seems that classical optics simulations are limited to quantum walks on lattices. We impose no such constraints, allowing for the simulation of QWs on arbitrary graphs.

Conductance Bound for Quantum Walks. In Theorem 4 we show how invariant stochastic processes, in particular quantum walks and lifted Markov chains, have a mixing time $\tau \in \Omega(1/\Phi_{G,\pi})$. Thereto we prove that the LMC simulator from Theorem 3 obeys a conductance bound.

Conductance bounds for lifted Markov chains were already proven in the early LMC paper by Chen, Lovász and Pak [9]. We however relax their definition of LMC mixing in two ways, preventing the direct use of their result on our LMC simulator. First of all, they require that an LMC mixes on the lifted state space $\hat{\mathcal{V}}$, whereas we only care about mixing of the marginal distribution on $\mathcal{V}$ (that is, after applying the marginalization map f). An LMC can therefore be periodic and still mix, as is the case for our LMC simulator. Secondly, they consider the worst initial state over the entire lifted state space $\hat{\mathcal{V}}$. Since we allow initialization of $\hat{\mathcal{V}}$ through the initialization map F, we only consider the worst initial state in the image $\mathrm{Im}(F) \subset \hat{\mathcal{V}}$. We show that these relaxations still allow for a conductance bound on the mixing time, provided that the resulting LMC is invariant. In the below Section 6.3 we give some further insight in this invariance condition. In more recent work, Ramanan and Smith [93] prove similar conductance bounds for continuous time LMCs.

Conductance bounds for quantum walks have been discussed in the early paper by Aharonov et al [15] and a more recent paper by Temme et al [51]. As a main difference, both papers are limited to *unital quantum walks*. This is a subclass of quantum walks where at each time step a random unitary operator from a set $\{U_k\}$ is applied, according to some distribution p over the set. This results in a Kraus map $\Psi[\rho] = \sum p(k) U_k \rho U_k^\dagger$. Such quantum walks necessarily converge to the uniform distribution over $\hat{\mathcal{V}}$, hence limiting the possible limit distribution over $\mathcal{V}$. The proof in [15] takes a geometric approach, akin to the proof of the Markov chain conductance bound given in Proposition 3. To do so they strongly rely on the unitarity of the evolution, and we see no direct way of extending their proof to general quantum walks. We note that for the special case to which their bound applies, our bound reproduces their bound, as shown in the following lemma.

Lemma 6. *For a uniform distribution π, it holds that $\Phi_{G,\pi} \leq \Phi'$.*

Proof. Let P be any Markov chain on G that mixes to the uniform distribution, and let $\mathcal{S}$ be any subset. We see that

$$\begin{aligned}\Phi_P(\mathcal{S}) &= \frac{\sum_{i\in\mathcal{S}, j\in\mathcal{S}^c} P(j,i)\pi(i)}{\pi(\mathcal{S})} \\ &= \frac{\sum_{i\in\mathcal{S}, j\in\mathcal{S}^c} P(j,i)}{|\mathcal{S}|} \leq \frac{\sum_{j\in\partial\mathcal{S}^c} 1}{|\mathcal{S}|} = \frac{|\partial\mathcal{S}^c|}{|\mathcal{S}|},\end{aligned}$$

where the inequality follows from the fact that $\sum_{i\in\mathcal{S}, j\in\mathcal{S}^c} P(j,i) = \sum_{i\in\mathcal{S}, j\in\partial\mathcal{S}} P(j,i)$ and $\sum_{i\in\mathcal{V}} P(j,i) = 1$ since the stationary distribution of P is the uniform one. So for any subset $\mathcal{S}$ and Markov chain P with a uniform stationary distribution, it holds that $\Phi_P(\mathcal{S}) \leq |\partial\mathcal{S}|/|\mathcal{S}|$. This implies that $\Phi(P) \leq \Phi'$ and therefore $\Phi_{G,\pi} \leq \Phi'$. □

The proof in [51] takes a very different approach, estimating the mixing time of a unital quantum walk using estimates on the spectral gap of the dynamics. They consider a unital quantum walk defined by the Kraus map $\Psi[\rho] = \sum p(k) U_k \rho U_k^\dagger$,

and associate a "symmetrized map" $\widetilde{\Psi}[\rho] = \sum_{k,l} p(k)p(l)U_l^\dagger U_k \rho U_k^\dagger U_l$. They then show that the mixing time of Ψ can be bounded by

$$\tau \geq \left(\min_{0<|\mathcal{S}|\leq|\mathcal{V}|/2} \mathrm{tr}((I - \Pi_{\mathcal{S}})\widetilde{\Psi}(\Pi_{\mathcal{S}}/|\mathcal{S}|)) \right)^{-1}. \tag{6.1}$$

In words, the right hand side denotes the probability of being in $\mathcal{S}^c$, when starting from the stationary state on $\mathcal{S}$. When minimized over all unital quantum walks this gives the same lower bound as our bound, be it restricted to unital quantum walks having a uniform stationary distribution. Their proof relies strongly on the unitarity of the quantum dynamics, and they explicitly mention that there seems no direct way to extend their results to non-unital quantum walks. Interestingly, their technique does allow to prove a converse upper bound for unital quantum walks using a quantity similar to the above one, extending the bound presented in Proposition 4 to unital quantum walks.

6.2 Infinite Graphs

Our results prove that quantum walks cannot accelerate mixing over classical Markov chains with memory, provided that no further restrictions are put on construction or resources. This shows that observing an improved mixing behavior in some physical system can not be diagnostic for quantum effects, as we discussed at the end of Chapter 4. The basis of these results was the similarity between the Hadamard quantum walk in Example 6 and the LMC in Example 5, both quadratically accelerating the random walk mixing time on the cycle $\mathbb{Z}_N$. Considering the same quantum walk on the infinite line $\mathbb{Z}$ allows to uncover a caveat to our result, provided that we relax the definition of mixing time.

We consider the situation as presented in one of the initial quantum walk papers by Ambainis et al [14], studying the Hadamard quantum walk on the infinite line graph $\mathbb{Z}$. They are interested in a property that we will call *linear mixing*, where a stochastic process $\{\Gamma_t\}$ is linear mixing if and only if there exists a constant $\delta < 1$ such that for all t sufficiently large it holds that

$$\|\Gamma_t[e_0] - \pi_t\|_{\mathrm{TV}} \leq \delta,$$

where π_t denotes the uniform distribution over $[-t, t] \subset \mathbb{Z}$. In words this means that for all t sufficiently large, the distribution $\Gamma_t[e_0]$ will have some nonnegligible correlation with the uniform distribution π_t over $[-t, t]$. As an example, a random walk on $\mathbb{Z}$, whose distribution converges to a normal distribution with standard deviation in $O(\sqrt{t})$, is not linear mixing. To see this it suffices to note that the probability of finding this walk outside the interval $[-\sqrt{t}\log t, +\sqrt{t}\log t]$ is in $O(1/t)$. This implies that $\|\Gamma_t[e_0] - \pi_t\|_{\mathrm{TV}} \to 1$.

In [14] the following proposition is proved about the Hadamard quantum walk, defined in Section 2.2.

Proposition 10 ([14]). *The Hadamard quantum walk is linear mixing.*

It follows from our results on QW simulation that there exists a lifted Markov chain that simulates the Hadamard walk on $\mathbb{Z}$, and is hence linear mixing, yet it requires an infinitely large auxiliary register or memory. If we require that the memory remains finite, then we can prove that no translationally invariant lifted Markov chain is linear mixing.

Proposition 11. *There exists no translationally invariant lifted Markov chain, having a finite memory, that is linear mixing.*

Proof. Let $\hat{P}$ be a translationally invariant LMC that describes the probability distribution of a random variable $(X_t, Z_t) \in \hat{V} = \{1, 2, \ldots, \kappa\} \times \mathbb{Z}_N$, for some finite memory size $\kappa \in \mathbb{Z}$. If (X_0, Z_0) is distributed according to some distribution v_0, then (X_t, Z_t) will be distributed according to $v_t = \hat{P}^t v_0$. By the translational invariance of the LMC we know that

$$\begin{aligned}\mathbb{P}((X_{t+1}, Z_{t+1}) =&(k', z') \mid (X_t, Z_t) = (k, z))\\ &= \mathbb{P}((X_{t+1}, Z_{t+1} - Z_t) = (k', z' - z) \mid X_t = k),\end{aligned}$$

where by locality we know that $Z_{t+1} - Z_t \in \{0, \pm 1\}$. This equality shows that we can describe the dynamics using an associated Markov chain $(X_{t+1}, Y_{t+1}) = (X_{t+1}, Z_{t+1} - Z_t)$, with $Y_t \in \{0, \pm 1\}$, and simply set $Z_t = Z_0 + \sum_{l=1}^{t} Y_l$. If the LMC starts in the origin, $Z_0 = 0$, then $Z_t = \sum_{l=1}^{t} Y_l$ and so we can focus on this sum of the random variable Y_l.

First assume that the Markov chain $W_t = (X_t, Y_t)$ is irreducible with a stationary distribution π, so that we can apply the central limit theorem for Markov chains [94, 95]:

$$\frac{1}{\sqrt{t}}\left(\sum_{l=1}^{\tau} h(W_l) - \mathbb{E}_\pi(h)\right) \xrightarrow{d} \mathcal{N}(0, \sigma_h^2),$$

for any bounded function h, with $\mathbb{E}_\pi(h)$ the expectation of h with respect to distribution π and σ_h some finite constant that can be bounded as a function of the Markov chain hitting times. Setting $h(W_l) = Y_l$, this shows that $Z_t/\sqrt{t} = \sum_{l=1}^{t} Y_t/\sqrt{t}$ weakly converges to a normal distribution with a bounded variance and mean $\sqrt{\tau}\mathbb{E}_\pi(h)$. As a consequence, for any $\epsilon > 0$ there exists some c such that

$$\lim_{t\to\infty} \mathbb{P}(|Z_t - \sqrt{\tau}\mathbb{E}_\pi(h)|/\sqrt{t} \geq c) = \lim_{t\to\infty} \mathbb{P}(|Z_t - \sqrt{\tau}\mathbb{E}_\pi(h)| \geq c\sqrt{t}) \leq \epsilon.$$

With $p_t = f(v_t)$ describing the marginalized probability distribution of Z_t, we can use the operational definition $\|p_t - \pi_t\|_{\mathrm{TV}} = \max_{\mathcal{S}\subset\mathbb{Z}} |p_t(\mathcal{S}) - \pi_t(\mathcal{S})|$ to bound

$$\|p_t - \pi_t\|_{\mathrm{TV}} \geq p_t([-k, k]) - \pi_t([-k, k]) \geq 1 - \epsilon - 2k/t,$$

with $k = \lfloor c\sqrt{t} \rfloor$. For any $\delta < 1$, we can therefore find an ϵ and a t_0 such that if $t \geq t_0$ then $\|p_t - \pi_t\|_{\mathrm{TV}} > \delta$. This implies that the LMC cannot be linear mixing. In the remaining case that the Markov chain $W_t = (X_t, Y_t)$ is reducible, we can make use of the case that any finite, reducible Markov chain converges to a finite mixture of irreducible Markov chains. Applying the above reasoning for each of these irreducible Markov chains proves the proposition. □

The proof shows that any such LMC weakly converges to a finite mixture of normal distributions, moving with different drift velocity. This proves a clear distinction between quantum walks and lifted Markov chains. If restrictions can be put on the memory of an LMC in a certain setting, it should be possible to estimate from which time scale it should start behaving diffusively. Superdiffusive behavior could then serve as a signaler for quantum behavior.

As an interesting side remark, note that if we consider the Cesaro average of a Markov chain then this result no longer holds. Indeed, the Cesaro average of a walk that deterministically moves to the left or to the right will already be linear mixing. This highlights the gain that a Cesaro average can induce.

6.3 Breaking Invariance and the Conductance Bound

The invariance of quantum walks and stochastic processes is a crucial clause to our results, whose discussion we have so far omitted. Recall that the invariance of a stochastic process $\{\Gamma_t\}$, mixing to some distribution π, requires that

$$\Gamma_t[\pi] = \pi, \quad \forall t \geq 0.$$

In other words, if the process starts in its limiting state, it stays there. This condition is naturally fulfilled in most mixing schemes, where the dynamics have a stabilizing or contracting effect towards some minimal energy or maximal entropy state. As such, the limit distribution is effectively a stationary or equilibrium state of the dynamics. Indeed, it is automatically fulfilled for time-independent classical Markov chains and time-independent quantum walks or quantum Markov chains that mix to a classical distribution (diagonal density matrix), as well as for their time averages. Furthermore, it is also naturally satisfied on any vertex-transitive graph, where the dynamics do not break the symmetry. From a design perspective, invariance of a mixing scheme is an important resource that allows to *amplify* the closeness to the target distribution: it allows to design and run a scheme up to a certain threshold (e.g., mixing up to TV-distance $1/4$), and then simply reiterate this scheme, leading to exponential convergence. This is exactly what we prove in Lemma 4, which extends the amplification lemma 1 to general invariant stochastic processes. As a byproduct, such amplified process is to some extent resilient to errors: if at some point an error occurs in the dynamics (a broken link, a skipped time step), then the contractive nature of the dynamics will wash this away.

Contrarily, and this is the main point of this section, we can show that if invariance does not hold, and no alternative restrictions are imposed, then no conductance bound holds. Indeed, we can show that in this case the trivial diameter bound becomes tight. Specifically, for any graph G with diameter D, and distribution π having full support, we can construct a lifted Markov chain that exactly mixes to π in D steps, which is the absolute lower bound for any local process by the diameter lower bound from Proposition 2. Thereto we build on the following proposition from [96], proven in the rather different context of Schrödinger bridges for maximum entropy problems. It states the existence of stochastic bridges of diameter length that connect any two distributions over a graph. The effective construction of these stochastic bridges is discussed in [97].

Proposition 12 ([96]). *For any graph G with diameter D and distributions v and v' such that $v' > 0$, there exists a stochastic bridge $\{P_t^{(v,v')} \sim G \mid 1 \leq t \leq D\}$ such that*

$$P_D^{(v,v')} P_{D-1}^{(v,v')} \dots P_1^{(v,v')} v = v'.$$

We can combine these stochastic bridges into a local stochastic process $\{\Gamma_t \sim G\}$ by setting

$$\Gamma_t[e_i] = \begin{cases} P_t^{(i,\pi)} P_{t-1}^{(i,\pi)} \dots P_1^{(i,\pi)} e_i & t \leq D \\ P_D^{(i,\pi)} P_{D-1}^{(i,\pi)} \dots P_1^{(i,\pi)} e_i = \pi & t > D, \end{cases}$$

and extending it to arbitrary distributions by linearity. This stochastic process is local by the locality of the transition matrices. Exactly like in Section 4.2, we can combine these stochastic bridges in the LMC simulator (4.2). As stated in Proposition 6, this leads to an LMC that simulates the stochastic process defined above, resulting in the following proposition.

Proposition 13. *For any graph G with diameter D and distribution $\pi > 0$, we can construct an LMC $\hat{P}_D$ that mixes exactly to π in D steps: for any distribution v and all $t \geq D$ it holds that*

$$(f \circ \hat{P}_D^t \circ F)[v] = \pi.$$

By our conductance bound, the above LMC cannot be invariant in general. Indeed, consider for instance our example of the random walk stationary distribution π over the binary tree $T_{2,k}$, discussed in Section 5.2. This graph has a diameter $2k$, yet it has a conductance $\Phi_{G,\pi} \in \Theta(1/2^k)$. Our conductance bound in Theorem 4 hence proves that any invariant LMC has a mixing time $\tau \in \Omega(2^k)$. The LMC constructed in the above proposition has an exponentially lower mixing time $\tau \in \Theta(k)$, and therefore cannot be invariant.

The above proposition shows that indeed the invariance of a stochastic process is a necessary condition for the existence of a conductance bound, rather than it being an artifact of the proof. Moreover, we discussed how it is in fact a natural property of equilibrating dynamics, and is a useful resource for the design of mixing schemes, allowing to exponentially amplify the closeness to a goal distribution and make a scheme resistant to errors.

6.4 Outlook

To recapitulate, we have proved that for any invariant stochastic process, quantum walks in particular, we can construct a lifted Markov chain that has the same mixing time. By bounding the mixing time of this lifted Markov chain, we were able to prove a conductance bound on the mixing time of the original process, which is tight up to a log factor. The lift construction should be interpreted as an existence result, since its construction time and local memory size (size of the coin) scale as a polynomial in $|\mathcal{V}|$, the size of the graph.

Efficient Constructions. The main open question is the existence of efficiently constructible quantum walks and lifted Markov chains that accelerate the simple Markov chain mixing time. The existence of such constructions for quantum walks

on regular graphs, where every node has the same degree, was conjectured more than 10 years ago in [45]. Currently however the only known efficient quantum walk and lifted Markov chain constructions either rely strongly on the symmetry of the graph [8, 9, 15, 38, 39, 45], the examples being limited to Cayley graphs of Abelian groups, or require additional resources such as an approximation of the stationary state or a sequence of slowly varying Markov chains [40, 42, 43]. An intermediate question that we could address is whether for any bounded degree graph there even exists a quantum walk or lifted Markov chain that reaches the conductance bound (possibly up to a log factor) with a memory size polynomial in $\log |\mathcal{V}|$. This seems like a necessary condition for their efficient construction. Especially in the case of lifted Markov chains we feel that this is a manageable question, maybe building on the techniques from [7] used to show that no simple Markov chain on the cycle can reach the conductance bound.

Relaxing Invariance. A second open question relates to the invariance condition. From the former section we know that if we drop the invariance condition, and impose no alternative constraint, diameter time mixing becomes possible for LMCs and quantum walks. In the second part of this thesis we will treat a "computationally feasible" quantum walk algorithm that is not invariant, and mixes in $\Theta(\sqrt{2^k})$ steps on the binary tree $T_{2,k}$. This breaks the $\Omega(2^k)$ conductance bound discussed in Example 11, yet it is still exponentially larger than the diameter. Can we impose some intermediate condition, less stringent than invariance, for which this is optimal? Related to this is the running conjecture, stated for instance in [45], that quantum walks can always quadratically speed up random walks (up to maybe a log factor). This points to a lower bound $\widetilde{\Omega}(\sqrt{\tau_{RW}})$ that lies intermediate between our conductance bound and the trivial diameter bound.

Another interesting example where invariance is explicitly broken is *simulated annealing* [73]. This is often applied in cases where probabilities reflect a cost function with many local minima, so that the Markov chain can easily get stuck and take a long time to converge to the correct distribution. Simulated annealing provides a remedy where a time-dependent sequence of Markov chains is proposed, whose transition probabilities and stationary distributions converge gradually to the goal distribution, but where the local minima and related irregularity in the distribution only shows up progressively towards the end of the sequence. Simulated annealing is especially relevant in cases where the main slowdown is caused by the goal distribution π and its irregularities, rather than by some graph locality which is fixed, as we are mainly considering. We leave it as an open question whether our techniques can be extended to this setting.

Improving Results. Our results can certainly be enhanced in different places. The following are some relevant questions:

- For every graph G and stationary distribution π, the conductance bound $\tau \in \Omega(1/\Phi_{G,\pi})$ is an "absolute lower bound". The bound only depends on G and π and is thus equal for all quantum walks and lifted Markov chains over G that mix to π. Contrarily, the original conductance bound $\tau \in \Omega(1/\Phi(P)) \subset \Omega(1/\Phi_{G,\pi})$ for some Markov chain P depends on the specific Markov chain, and a poorly chosen Markov chain will have a larger upper bound than a better chosen one. We believe that a similarly improved bound should hold for quantum walks, depending not only on G and π but also in an intuitive way on the specific

structure of the quantum walk. The result from [51] in (6.1) is an example of such a result, be it restricted to unital quantum channels.

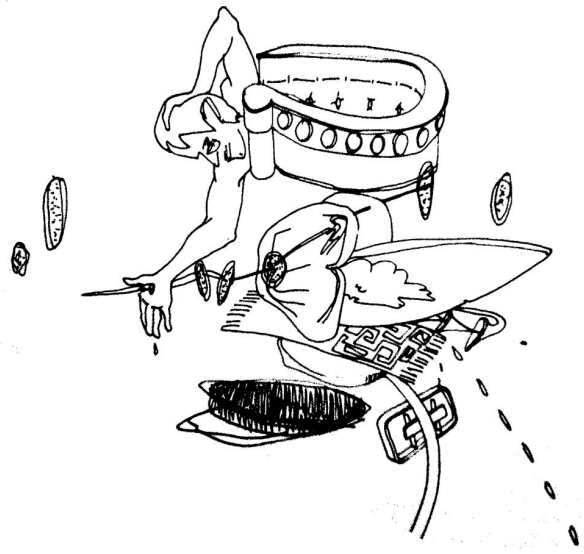
- The graph conductance

$$\Phi_{G,\pi} = \max_{P \sim G : P\pi = \pi} \Phi_P$$

is defined somewhat elusively, requiring a maximization over Markov chains. We have shown that it can be efficiently bounded (5.1), yet this bound is not tight. Can we find a closed form for $\Phi_{G,\pi}$?

Cheeger Inequalities. Finally we mention a recent surge of results, see [98, 99] and references therein, in which conductance bounds are used to estimate the spectral gap of Hermitian matrices. Since the spectral gap is closely related to the mixing time of the associated dynamics, it seems possible to apply our results to these issues.

2nd Part: Fast-Forwarding



In the first part of this thesis, we asked whether Markov chains can simulate quantum walks. We showed that for any quantum walk there exists a Markov chain with added memory, called a lifted Markov chain, that simulates and matches the quantum walk mixing dynamics. As early as 1998, Watrous [1] posed the opposite question of whether quantum walks can be used to *quantum simulate* Markov chains. For a Markov chain P and initial distribution v, quantum simulation denotes creating the *quantum state* $|P^t v\rangle$ associated to the classical distribution $P^t v$. It is defined by

$$|P^t v\rangle = \frac{1}{\|P^t v\|} \sum (P^t v)(j)\, |j\rangle. \tag{6.2}$$

For a general nonzero vector w, we will use the notation $|w\rangle = \frac{1}{\|w\|} \sum w(j)|j\rangle$ which associates a quantum state $|w\rangle$ to w, with the prefactor ensuring that the quantum state $|w\rangle$ is normalized, $\||w\rangle\| = 1$. Note the crucial difference with the trivial simulation discussed in Section 2.3, Example 7, where we showed how a quantum walk can create the *classical* distribution $P^t v$. Indeed creating the quantum state $|P^t v\rangle$, a task commonly referred to as *quantum state generation* [40], is generally much more difficult than creating $P^t v$. The generation of such quantum states is an important resource for quantum algorithms such as quantum search [21, 100], quantum machine learning [101], and state comparison [40]. Watrous constructed a unitary quantum walk operator from P, inheriting the same locality structure as P, that allows to generate $|P^t v\rangle$ with success probability $\|P^t v\|^2$, requiring t QW steps. His main goal was to prove that random walk algorithms, representative for the class of space-efficient randomized classical computations, could be simulated in a space-efficient way on a quantum computer, without the need for intermediate measurements.

In this thesis part we show that we can perform this quantum simulation quadratically faster, proving that *quantum walks can quadratically fast-forward a general reversible Markov chain*. The reversibility condition is crucial here, showing for instance that the LMC simulators from previous section cannot be fast-forwarded in this way. We build on the work of Ambainis [4] and Szegedy [3] that took inspiration from the work of Watrous. They use quantum walks in a different context, namely for speeding up search problems on graphs. The gist of our results is that we can apply these speedup techniques to the original problem of quantum simulation, giving rise to a technique that we call *quantum fast-forwarding* (QFF). For a symmetric Markov chain P (we consider general reversible MCs in the main text), QFF allows to generate the state $|P^t v\rangle$ with the same success probability $\|P^t v\|^2$ in $O\big(\sqrt{t} \log \|P^t v\|^{-1}\big)$ QW steps. This is a quadratic acceleration, up to a log-factor. Much of the previous work that builds on the Ambainis-Szegedy quantum walk scheme, such as [3, 21, 22, 41, 42], relies on a quadratic acceleration of the limit behavior of the original Markov chain. Our result, however, allows to capture and accelerate the intermediate dynamics.

In Chapter 9 we discuss how QFF very naturally allows to accelerate random walk algorithms for graph property testing. The latter was initiated by the seminal work of Goldreich, Goldwasser and Ron [102]. The goal is determine whether some given graph either has a certain property, such as bipartiteness or a large expansion, or is far away from having that property. This is a rewarding relaxation with respect to the much more difficult problem of deciding whether a given graph has that

property, typically leading to algorithms that are sublinear in the size of the graph. We first consider the Goldreich-Ron algorithm [103] for testing whether a graph has large expansion or not, and show how QFF allows to quantize and accelerate this algorithm very naturally. Second we consider the more recent line of testing clusterability of a graph [104, 105]. These algorithms allow to classify nodes of appropriately clustered graphs, solving a robust version of *s*-*t* connectivity which is also relevant outside of the setting of property testing. We again show how QFF leads to an immediate quantum speedup for these algorithms.

In the final Chapter 10 we discuss some more applications of QFF. First we demonstrate how QFF allows to escape large sets on a graph quadratically faster than classical walks. This is a new result which connects to a long line of research on quantum walk algorithms for search problems [3, 19, 21, 106, 107]. We show how our new result allows to make progress on the open problem of hitting large sets. We also discuss a direct application of QFF to the problem of quantum state generation, as proposed in [40]. Here we are given a classical algorithm that samples according to some distribution π, and the task is to create the corresponding quantum state $|\sqrt{\pi}\rangle = \sum \sqrt{\pi(j)}|j\rangle$. Most existing algorithms [40, 41, 108] assume that π is given as the stationary distribution of a Markov chain. We show how QFF can solve this problem in situations where this is not the case, for instance when π is described as an intermediate rather than a limit state of a Markov chain.

Chapter 7
Quantum Simulating Markov Chains

In the coming chapter we mainly review the work by Watrous on quantum simulating Markov chains. This work was one of the earliest proponents and inspirations for the large field of quantum walk algorithms. It lies at the basis of our further constructions.

7.1 Watrous Scheme

In Section 2.2 we already discussed that quantum walks must in general be defined on lifted graphs, where for instance some coin space is added to the original graph. Watrous [1] resolved this issue by generically appending an extra node register to the graph. Although not strictly necessary, we will also add an "initialization" state $\{\flat\}$ to the extra register. The new node space thus becomes

$$\hat{\mathcal{V}} = \mathcal{V} \times \{\flat, \mathcal{V}\} = \{(i, j) \mid i \in \mathcal{V}, j \in \{\flat, \mathcal{V}\}\}.$$

It will be useful to identify the ordered pairs in $\mathcal{V} \times \mathcal{V}$ with directed graph edges, as shown on the left in Figure 7.1. The associated Hilbert space is $\mathcal{H} = \mathcal{H}_\flat \oplus \mathcal{H}_{\mathcal{V}\times\mathcal{V}}$, with "$\oplus$" the direct sum. We will call

$$\mathcal{H}_\flat = \text{span}\{|i, \flat\rangle \mid i \in \mathcal{V}\}$$

the *flat subspace*, with associated projector $\Pi_\flat = I \otimes |\flat\rangle\langle\flat|$. Similar to the lifted Markov chain and quantum walk on the cycle that we introduced in the first thesis part, the quantum walk on $\mathcal{H}$ is defined by a shift and a coin toss operator. Note that these operators are slightly adapted from Watrous', generalizing them from random walks to general Markov chains, and allowing for a better streamlining with further constructions.

1. *Coin toss* $V = \sum_i |i\rangle\langle i| \otimes V_i$*:* local action on the coin space. Conditioned on the current node, which is held in the first register, the coin toss implements the operator V_i on the coin space:

$$V|i, \psi\rangle = |i\rangle \otimes V_i|\psi\rangle.$$

By the design of the QW, as we will see later, it suffices to characterize the action of V_i on the state $|\flat\rangle$, which we define as

$$V|i,\flat\rangle = |i\rangle \otimes V_i|\flat\rangle = |i\rangle \otimes |\psi_i\rangle = |i\rangle \otimes \sum_j \sqrt{P(j,i)}|j\rangle. \tag{7.1}$$

This maps a state $|i,\flat\rangle$ to a superposition over the outgoing edges which is defined by the original Markov chain. We show $|i,\psi_i\rangle$ in the middle of Figure 7.1. The operators V_i can be completed into a unitary matrix by setting

$$V_i = |\psi_i\rangle\langle\flat| + \sum_j |\psi_{i,j}^{\perp}\rangle\langle j|, \tag{7.2}$$

where the states $|\psi_{i,j}^{\perp}\rangle$ are chosen such that $\{|\psi_i\rangle, |\psi_{i,1}^{\perp}\rangle, \ldots, |\psi_{i,|\mathcal{V}|}^{\perp}\rangle\}$ forms an orthonormal basis. By its definition, this implies that also V will be unitary.

2. *Shift S:* defined by the permutation

$$|i,j\rangle \mapsto S|i,j\rangle = \begin{cases} |j,i\rangle & (i,j) \in \mathcal{E} \\ |i,j\rangle & \text{otherwise,} \end{cases}$$

and $S|i,\flat\rangle = S|i,\flat\rangle$. This operator performs the actual walk, hopping from node i to node j. We illustrate this on the right of Figure 7.1.

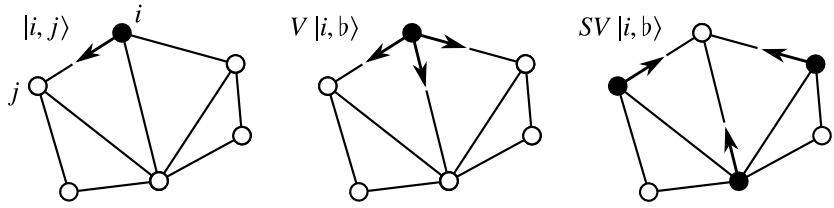


Fig. 7.1 Figure demonstrating the Watrous QW scheme. (l) A basis state $|i,j\rangle \in \hat{\mathcal{V}}$ can be identified with the directed edge (i,j). (m) The coin toss V maps an initial state $|i,\flat\rangle$ to a superposition of outgoing edges. (r) The shift S maps a state $|i,j\rangle$, localized on node i, to a state $|j,i\rangle$, localized on node j.

These operators closely resemble the shift and coin toss operators that we used to define the lifted Markov chain and quantum walk on the cycle. The final form of the quantum walk is

$$U_P = V^{\dagger}SV. \tag{7.3}$$

We will often write U instead of U_P when the context allows it. Note that this form deviates from the "coined QW" that we introduced in the first thesis part. It is however still a local QW, as we prove in the below lemma.

Lemma 7. *If P is local with respect to a graph G ($P \sim G$) then U_P is local with respect to G ($U_P \sim G$).*

Proof. Since V is of the form $V = \sum |i\rangle\langle i| \otimes V_i$, and hence also $V^\dagger = \sum |i\rangle\langle i| \otimes V_i^\dagger$, these operators only work on the auxiliary register, performing a local coin toss. The shift operator S performs the actual walk, yet it only maps states $|i, j\rangle$, localized on node i, to states $|j, i\rangle$, localized on node j, if $(i, j) \in \mathcal{E}$. □

It will be instrumental to analyze the working of U on an initial state $|i, \flat\rangle$ in $\mathcal{H}_\flat$. We can show that

$$U|i, \flat\rangle = \sum_j \sqrt{P(i, j)P(j, i)}|j, \flat\rangle + |\psi^\perp\rangle, \tag{7.4}$$

where $|\psi^\perp\rangle$ is some state perpendicular to the flat subspace $\mathcal{H}_\flat$. To see this, we first directly apply the definition of V and S to see that

$$SV|i, \flat\rangle = S|i, \psi_i\rangle = \sum_j \sqrt{P(j, i)}|j, i\rangle.$$

Towards the analyzing the action of $V^\dagger$, notice that $V^\dagger = \sum |i\rangle\langle i| \otimes V_i^\dagger$ and from (7.2) we can write $V_i^\dagger = |\flat\rangle\langle\psi_i| + \sum_j |j\rangle\langle\psi_{i,j}^\perp|$, so that

$$V^\dagger SV|i, \flat\rangle = \sum_j \sqrt{P(j, i)}|j\rangle V_j^\dagger|i\rangle = \sum_j \sqrt{P(j, i)}\langle\psi_j|i\rangle|j, \flat\rangle + |\psi^\perp\rangle.$$

By definition of $|\psi_i\rangle$, this proves the expression in (7.4).

From this, the below proposition by Watrous is immediate. It shows that the restriction of U to the flat subspace implements the *discriminant matrix*

$$D = \sqrt{P \circ P^T},$$

with the square root and product elementwise. We introduced this matrix earlier in Chapter 1. It is closely related to the original Markov chain P, and if P is reversible then they share the same eigenvalues. If P is symmetric, as was the case in Watrous' original setting of random walks on regular graphs, then $D = P$. In the following we write $D|v, \flat\rangle$ as shorthand for $(D \otimes I)|v, \flat\rangle$.

Proposition 14. *For any $|v, \flat\rangle$, it holds that*

$$\Pi_\flat U|v, \flat\rangle = D|v, \flat\rangle.$$

That is, if we perform a measurement $\{\Pi_\flat, I - \Pi_\flat\}$ on the state $U|v, \flat\rangle$ and the outcome is "$\flat$", we retrieve the state $|Dv, \flat\rangle$.

Proof. From (7.4) we see that $\Pi_\flat U|i, \flat\rangle = D|i, \flat\rangle$. By linearity, the proposition follows for general $|v, \flat\rangle$. □

We see that the quantum walk U maps a state $|v, \flat\rangle$ to a "good part" $\Pi_\flat U|v, \flat\rangle$ in the same flat subspace, and a part $(I - \Pi_\flat)U|v, \flat\rangle$ that we wish to reject.

7.2 Quantum Simulation Algorithm

The straightforward way of retrieving the good part $\Pi_\flat U|v,\flat\rangle = D|v,\flat\rangle$ of the quantum walk evolution is by performing a projective measurement $\{\Pi_\flat, I - \Pi_\flat\}$ on the state $U|v,\flat\rangle$. The measurement returns the quantum state $|Dv,\flat\rangle$ with probability

$$\mathbb{P}(\flat) = \|\Pi_\flat U|v,\flat\rangle\|^2 = \|D|v\rangle\|^2.$$

To simulate over multiple steps we can reiterate this scheme on its output, yielding the below quantum walk simulation algorithm. The scheme has a success probability $\|D^t|v\rangle\|^2$, as we show below. Later on, in Section 8.4, we will use a quantum search algorithm to quadratically improve this success probability to $\|D^t|v\rangle\|$. We also mention that in the original paper, Watrous proposed a somewhat more advanced scheme that allowed to perform quantum simulation without the intermediate measurements, at the cost of introducing an additional register of size $\log t$.

Algorithm 1 Quantum Simulation $\mathbf{QS}(|v\rangle, P, t)$

Input: quantum state $|v\rangle \in \mathcal{H}_\mathcal{V}$, Markov chain P, $t \in \mathbb{N}$
Do:

1: initialize the registers $\mathbf{R_1R_2}$ with the state $|v,\flat\rangle$
2: **for** $l = 1$ to t **do**
3: perform the quantum walk U_P on $\mathbf{R_1R_2}$
4: perform the measurement $\{\Pi_\flat, I - \Pi_\flat\}$
5: **if** outcome $\neq$ "$\flat$" **then** output "`Fail`" and stop
6: **end for**

Output: registers $\mathbf{R_1R_2}$

Complexity: t QW steps **Success probability:** $\|D^t|v\rangle\|^2$

Lemma 8. $\mathbf{QS}(|v\rangle, P, t)$ *outputs the state* $|D^t v,\flat\rangle$ *with success probability* $\|D^t|v\rangle\|^2$. *Otherwise it outputs "*`Fail`*". The algorithm uses* t *QW steps.*

Proof. Assume that the algorithm has been successful for l iterations of the **for**-loop. Then the state in registers $\mathbf{R_1R_2}$ is

$$|D^l v,\flat\rangle = \frac{1}{\|D^l|v\rangle\|} D^l|v,\flat\rangle.$$

By the former reasoning, applying U_P and the measurement $\{\Pi_\flat, I - \Pi_\flat\}$ on this state returns the state $|D^{l+1}v,\flat\rangle$ with a probability

$$\mathbb{P}_{l+1}(\flat) = \|\Pi_\flat U|D^l v,\flat\rangle\|^2 = \frac{\|D^{l+1}|v\rangle\|^2}{\|D^l|v\rangle\|^2}. \tag{7.5}$$

This represents the success probability of the $(l+1)$-th iteration. The total success probability over t iterations therefore becomes

$$\begin{aligned}\mathbb{P}_t(\flat)\mathbb{P}_{t-1}(\flat)\dots\mathbb{P}_1(\flat) &= \frac{\|D^t|v\rangle\|^2}{\|D^{t-1}|v\rangle\|^2}\frac{\|D^{t-1}|v\rangle\|^2}{\|D^{t-2}|v\rangle\|^2}\cdots\frac{\|D|v\rangle\|^2}{\||v\rangle\|^2}\\ &= \|D^t|v\rangle\|^2,\end{aligned}$$

using that the initial state is a normalized quantum state $\||v\rangle\| = 1$. □

In the following example we illustrate the use of this scheme to create superpositions over the nodes of a graph.

Example 12 (Creating Superpositions). Consider an ergodic symmetric Markov chain P on a graph $G = (\mathcal{V}, \mathcal{E})$. This random walk will converge to its stationary distribution π, which is the uniform distribution over the node set. We can use the above defined scheme to simulate this Markov chain and retrieve the corresponding quantum state $|\pi\rangle$, a uniform superposition over the node set.

By Lemma 8, and since $D = P$ for a symmetric transition matrix, we can create the state $|P^t i, \flat\rangle$ with probability $\|P^t|i\rangle\|^2$ using t QW steps. For t sufficiently large, this state will closely approximate the uniform superposition $|\pi\rangle$. To see this, we first bound the 2-norm by the 1-norm so that $\|P^t|i\rangle - \pi\| \leq \|P^t|i\rangle - \pi\|_1$. By the definition of the ϵ-mixing time $\tau(\epsilon)$ we know that $\|P^t|i\rangle - \pi\|_{\mathrm{TV}} = \frac{1}{2}\|P^t|i\rangle - \pi\|_1 \leq \epsilon'$ if $t \geq \tau(\epsilon')$. Now we can bound $\||P^t i\rangle - |\pi\rangle\|$ using the elementary fact that for any two nonzero vectors v and w it holds that

$$\left\|\frac{v}{\|v\|} - \frac{w}{\|w\|}\right\| \leq \frac{2\|v - w\|}{\max(\|v\|, \|w\|)}. \tag{7.6}$$

As a consequence, $\|P^t|i\rangle - \pi\| \leq 2\epsilon'$ implies that $\||P^t i\rangle - |\pi\rangle\| \leq \frac{4\epsilon'}{\|\pi\|}$. If we choose $\epsilon' = \frac{\|\pi\|}{4}\epsilon$, then we see that

$$\||P^t i\rangle - |\pi\rangle\| \leq \epsilon, \quad \forall t \geq \tau(\epsilon').$$

We can similarly bound the success probability $\|P^t|i\rangle\|^2$. If $\|P^t|i\rangle - \pi\| \leq 2\epsilon'$, then by the reverse triangle inequality we know that

$$\|P^t|i\rangle\|^2 \geq (\|\pi\| - 2\epsilon')^2 = (1 - \epsilon)^2\|\pi\|^2.$$

Since $\|\pi\|^2 = \sum_j \frac{1}{|\mathcal{V}|^2} = \frac{1}{|\mathcal{V}|}$, the success probability becomes $(1 - \epsilon)^2/|\mathcal{V}|$.

In summary, running $\mathbf{QS}(|j\rangle, P, t)$ for $t = \tau(\epsilon' = \|\pi\|\epsilon/4)$ returns a state $|D^t j\rangle$ ϵ-close to the uniform superposition $|\pi\rangle$ with success probability $(1 - \epsilon)^2/|\mathcal{V}|$, requiring $\tau(\epsilon' = \|\pi\|\epsilon/4)$ QW steps. Using Lemma 1, we can bound $\tau(\epsilon' = \|\pi\|\epsilon/4) \in O(\tau \log \frac{|\mathcal{V}|}{\epsilon})$ QW steps, with τ the mixing time of the Markov chain. The total expected runtime for creating a state ϵ-close to the uniform superposition can thus be bounded by $O(\tau|\mathcal{V}| \log \frac{|\mathcal{V}|}{\epsilon})$. On for example the cycle $\mathbb{Z}_N$, having $\tau \in \Theta(N^2)$, this comes down to $O\big(N^3 \log \frac{N}{\epsilon}\big)$ QW steps. △

This summarizes the main contribution from [1] that will be of interest to us: quantum walks allow to create the state $|D^t v\rangle$, with success probability $\|D^t|v\rangle\|^2$, using t QW steps. In the next section we will show that we can quadratically accelerate this scheme: it is possible to create the state $|D^t v\rangle$, with a success probability $\Theta(\|D^t|v\rangle\|^2)$, using $O(\sqrt{t})$ QW steps.

Chapter 8
Quantum Fast-Forwarding

Watrous' scheme shows that quantum walks can quantum simulate Markov chains. He used this result to prove that a quantum computer can simulate the broad class of classical random walk algorithms in a unitary, space and time efficient way. Key to his scheme is to isolate after every step the "good" part of the evolution $\Pi_\flat U$ from interfering with its complement. This can be done by interspersing measurements, as we discussed, and Watrous showed in his original paper that this can also be done by keeping an extra time register. This elimination of interference of the good part ensures that the quantum walk evolution keeps in line with the original Markov chain.

In subsequent work, Ambainis [4] ingeniously showed that these interference effects, avoided in the simulation schemes, can actually lead to speedups on classical algorithms. Indeed he used a quantum walk to show that the element distinctness problem, deciding whether there exist two equal elements in a list of N elements, can be solved in $\Theta(N^{2/3})$ queries to the list, which is optimal and improves over the classical optimum of $\Theta(N)$ queries. Earlier attempts based on amplitude amplification required a suboptimal $\Theta(N^{3/4})$ queries [109]. The quantum walk techniques leading to this optimal speedup were formalized more generally by Szegedy in [3]. They lie at the basis of an impressive stream of quantum algorithms for problems such as triangle finding [18], matrix product verification [110], quantum state generation [41, 42], Hamiltonian simulation [111], and Markov chain hitting on graphs [21, 22].

In the coming chapter we show that we can actually apply these quantum walk speedup techniques to the original problem of quantum simulating Markov chains. The simulation schemes introduced in the previous chapter take t quantum walk steps to simulate a Markov chain over t steps, with a certain success probability. Taking inspiration from the above quantum walk speedup techniques, we show that this can in fact be done quadratically faster. We construct a quantum algorithm that requires $\widetilde{O}(\sqrt{t})$ quantum walk steps to simulate a Markov chain over t steps, with the same success probability. We call this technique *quantum fast-forwarding* (QFF), and it is the main conceptual contribution of this thesis part. In the first section we will introduce the quantum walk speedup techniques from Ambainis and Szegedy. In the second and third section we apply these techniques to the quantum simulation problem, and show how they lead to our QFF algorithm. In

the final section we demonstrate how we can use amplitude amplification, the generalization of Grover's quantum search algorithm, to quadratically improve the success probability of both the QFF algorithm and the original quantum sampling algorithm.

8.1 Ambainis-Szegedy Scheme

Recall the Watrous quantum walk scheme associated to a Markov chain P. It is defined by the quantum walk operator $U = V^\dagger SV$ on the lifted node set $\hat{\mathcal{V}} = \mathcal{V} \times \{\mathcal{V}, \flat\}$, with coin toss operator V mapping $|i, \flat\rangle$ to

$$V|i, \flat\rangle = |i, \psi_i\rangle = \sum_j \sqrt{P(j,i)}|i, j\rangle$$

and shift operator S defined by $S|i, j\rangle = |j, i\rangle$ for $(i, j) \in \mathcal{E}$ and the identity elsewhere. We proved that

$$D|v, \flat\rangle = \Pi_\flat U|v, \flat\rangle,$$

with $D = \sqrt{P \circ P^T}$ the discriminant matrix, and we used this to quantum simulate the original Markov chain P.

In the work by Ambainis [4], generalized and formalized by Szegedy [3], an alternative quantum walk operator is proposed:

$$W = R_\flat U = (2\Pi_\flat - I)U,$$

where $R_\flat = 2\Pi_\flat - I$ is the reflection operator around the flat subspace. The critical insight in Szegedy's work stemmed from the fact that we can rewrite

$$W^{2t} = V^\dagger \Big[\big(VR_\flat V^\dagger\big)\big(SVR_\flat V^\dagger S\big) \Big]^t V = V^\dagger \big[R_\psi R_{S\psi}\big]^t V.$$

Therefore, for all even t and up to the unitary transformation V, the quantum walk W comes down to a product of two reflections $R_\psi = 2V\Pi_\flat V^\dagger - I$, reflecting around the subspace spanned by the states $|i, \psi_i\rangle$, and $R_{S\psi} = 2SV\Pi_\flat V^\dagger S - I$, reflecting around the subspace spanned by the states $S|i, \psi_i\rangle$. This product of reflections allows for a geometric analysis, and led to Szegedy's characterization of the spectrum of W. An interesting side note is that his result, describing the spectral properties of such a product of reflections, had already been proven in an 1875 paper by Jordan [112]. Since we only need part of his theorem, stated in the below proposition, we can provide a new proof which is more straightforward. We will denote by T_t the t-th Chebyshev polynomial of the first kind, and by $T_t(D)$ the polynomial in the discriminant matrix D.

Proposition 15. *For any $|v, \flat\rangle$, it holds that*

$$\Pi_\flat W^t|v, \flat\rangle = T_t(D)|v, \flat\rangle.$$

That is, if we perform a measurement $\{\Pi_\flat, I - \Pi_\flat\}$ on the state $W^t|v, \flat\rangle$ and the outcome is "$\flat$", we retrieve the state $T_t(D)|v, \flat\rangle / \|T_t(D)|v, \flat\rangle\|$. As a consequence, if $(\cos\theta, |v\rangle)$ is an eigenpair of D, then

$$\Pi_\flat W^t|v, \flat\rangle = T_t(\cos\theta)|v, \flat\rangle = \cos(t\theta)|v, \flat\rangle.$$

Proof. We easily find a recursion formula for $\Pi_\flat W^t$:

$$\begin{aligned}\Pi_\flat W^t &= \Pi_\flat R_\flat U(2\Pi_\flat - I)UW^{t-2} \\ &= 2\Pi_\flat U(\Pi_\flat W^{t-1}) - \Pi_\flat W^{t-2},\end{aligned}$$

using the fact that $\Pi_\flat R_\flat = \Pi_\flat$, and $U^\dagger = U$ so that $U^2 = UU^\dagger = I$. Since $\Pi_\flat W^0 = \Pi_\flat$ and $\Pi_\flat W = \Pi_\flat U$, this shows that we can express $\Pi_\flat W^t$ as a polynomial in $\Pi_\flat U$.

The Chebyshev polynomials of the first kind T_t are defined by

$$T_0(x) = 1, \quad T_1(x) = x, \quad T_t(x) = 2xT_{t-1}(x) - T_{t-2}(x).$$

Setting $x = \Pi_\flat U$ and $T_0(\Pi_\flat U) = \Pi_\flat$, this shows that we can express $\Pi_\flat W^t$ as

$$\Pi_\flat W^t = T_t(\Pi_\flat U).$$

From Proposition 14 we know that $(\Pi_\flat U)^t|v,\flat\rangle = D^t|v,\flat\rangle$, and therefore

$$\Pi_\flat W^t|v,\flat\rangle = T_t(D)|v,\flat\rangle.$$

Using the geometric definition of T_t, $T_t(\cos\theta) = \cos(t\theta)$, we see that if $D|v,\flat\rangle = \cos\theta|v,\flat\rangle$ then

$$\Pi_\flat W^t|v,\flat\rangle = T_t(\cos\theta)|v,\flat\rangle = \cos(t\theta)|v,\flat\rangle. \qquad \square$$

This proposition constitutes the basis from which most of the aforementioned quantum algorithms depart, and it will be the basis from which this work departs.

The gist of many quantum algorithms builds on the following observation: we can compare the original action of D^t on an eigenpair $(\cos\theta, |v\rangle)$

$$D^t|v,\flat\rangle = \cos^t(\theta)|v,\flat\rangle,$$

with the action of $\Pi_\flat W^t$

$$\Pi_\flat W^t|v,\flat\rangle = \cos(t\theta)|v,\flat\rangle.$$

Taylor expanding the respective *eigenvalue functions* $g^t(\theta) = \cos^t(\theta)$ and $f_s(\theta) = \cos(s\theta)$ yields

$$g^t(\theta) = 1 - \frac{t\theta^2}{2} + O(t^2\theta^4), \qquad \text{whereas} \quad f_s(\theta) = 1 - \frac{s^2\theta^2}{2} + O(s^4\theta^4).$$

Setting $s = \sqrt{t}$, we see that both expressions are equal up to second order in t. This suggests that the quantum walk *quadratically fast-forwards* the Markov chain, and so $\Pi_\flat W^{\sqrt{t}} \approx \Pi_\flat D^t$.

This observation underlies a range of quantum walk speedup results which are mainly concerned with accelerating the Markov chain asymptotics, where one is interested in the limit regime $\lim_{t\to\infty} P^t v = \pi$ and one wishes to approximate the quantum state $|\pi\rangle$. In these cases, the timescale for the classical Markov chain is for instance set by the inverse of the spectral gap $\frac{1}{\delta} = \frac{1}{1-\lambda_2}$ (for mixing tasks and Gibbs sampling, see [41, 42]), or by the sum of the inverses $\sum \frac{1}{1-\lambda_k}$ (for hitting tasks, see [3]). For these purposes, the low order conclusions from the above expansion generally suffice to achieve a quantum walk speedup in generating $|\pi\rangle$.

The main issue with the above analysis is that it breaks down for t and eigenvalues θ such that $t\theta \approx 1$, and the eigenvalue functions $g^t(\theta)$ and $f_t(\theta)$ start to diverge from each other. Figure 8.1 in the next section shows both functions for $\sqrt{t}\theta \in \Theta(1)$. Since quantum simulation aims to simulate the full dynamics of the Markov chain, without resorting to any limit behavior, this analysis cuts short. In the following section we construct a more involved and fine-grained quantum walk scheme whose eigenvalue function closely approximates the Markov chain eigenvalue function $g^t(\theta)$ for all values of t and θ, without losing the quadratic fast-forward. This will allow us to effectively fast-forward the full dynamics of the original Markov chain.

8.2 LCU and Chebyshev Truncation

We can create some wiggle room on the implementation of $\Pi_\flat W^t$, and therefore on its eigenvalue function, by implementing linear combinations of $\Pi_\flat W^t$ for different t. A similar approach has been used in for instance [113, 114] for Hamiltonian simulation and in [115] for optimizing quantum SDP solvers, where they call this technique *linear combination of unitaries* (LCU). We extract the below lemma from this work, and elaborate its details for completeness. Below the lemma we discuss how it can be used for our purpose

The below lemma shows how to implement a linear combination

$$\sum_{l=0}^{\tau} q_l \Pi_\flat W^l,$$

where we assume that $q_l \geq 0$ and $\sum q_l = 1$. To do so, we will again enlarge the state space from $\mathcal{H}_{\hat{\mathcal{V}}}$ to $\mathcal{H}_{\hat{\mathcal{V}}\times[\tau]}$, with $[\tau] = \{0, 1, 2, \ldots, \tau\}$. We will identify $|0\rangle = |\flat\rangle$ and call the span of states $|j, \flat, \flat\rangle \equiv |j, \flat\flat\rangle$, $j \in \mathcal{V}$, the flat subspace of $\mathcal{H}_{\hat{\mathcal{V}}\times[\tau]}$. The projector $\Pi_\flat$ will either denote the projector on the flat subspace of $\mathcal{H}_{\hat{\mathcal{V}}}$ or $\mathcal{H}_{\hat{\mathcal{V}}\times[\tau]}$, whichever it is will be clear from the context. The construction is very similar to the Watrous quantum walk scheme. It builds on a coin toss V_q on $\mathcal{H}_{\hat{\mathcal{V}}\times[\tau]}$, defined by the coefficients q_l as

$$V_q|\psi, \flat\rangle = \sum_{l=0}^{\tau} \sqrt{q_l}|\psi, l\rangle.$$

Then the *controlled W-operator* $cW = \sum_{l=0}^{\tau} W^l \otimes |l\rangle\langle l|$ is applied which, conditioned on the integer l in the last register, applies the operator W^l:

$$cW\, V_q|\psi, \flat\rangle = cW \sum_{l=0}^{\tau} \sqrt{q_l}|\psi, l\rangle = \sum_{l=0}^{\tau} \sqrt{q_l} W^l|\psi, l\rangle.$$

Finally, as in the Watrous QW, the operator $V_q^\dagger$ is applied, returning a state

$$V_q^\dagger\, cW\, V_q|\psi, \flat\rangle = \sum_{l=0}^{\tau} q_l W^l|\psi, \flat\rangle + |\psi^\perp\rangle, \tag{8.1}$$

where $|\psi^\perp\rangle$ is some quantum state perpendicular to the flat subspace. This leads to the following lemma, where we set $W_\tau = V_q^\dagger\, cW\, V_q$.

Lemma 9 (LCU). *For any* $|v, \flat\flat\rangle$*, it holds that*

$$\Pi_\flat W_\tau |v, \flat\flat\rangle = \left(\sum_{l=0}^{\tau} q_l \Pi_\flat W^l |v, \flat\rangle \right) \otimes |\flat\rangle = \left(\sum_{l=0}^{\tau} q_l T_l(D) |v\rangle \right) \otimes |\flat\flat\rangle.$$

Implementing the operator W_τ *requires* $O(\tau)$ *QW steps.*

Proof. From (8.1) we see that

$$\Pi_\flat \, V_q^\dagger \, cW \, V_q |v, \flat\flat\rangle = \left(\sum_{l=0}^{\tau} q_l \Pi_\flat W^l |v, \flat\rangle \right) \otimes |\flat\rangle.$$

Combined with Proposition 15, and by linearity, this proves the equality. In for instance [46, 111] it is discussed that the operator cW can be implemented in $O(\tau)$ QW steps, and the local coin tosses V_q and $V_q^\dagger$ require no QW steps. □

This lemma shows that if we apply the operator W_τ on $|v, \flat\flat\rangle$, and we perform a measurement $\{\Pi_\flat, I - \Pi_\flat\}$, then we retrieve the state $\left(\sum_{l=0}^{\tau} q_l T_l(D)|v\rangle \right) \otimes |\flat\flat\rangle / \| \sum_{l=0}^{\tau} q_l T_l(D)|v\rangle \|$ with probability $\| \sum_{l=0}^{\tau} q_l T_l(D)|v\rangle \|^2$. The corresponding eigenvalue function is then

$$\tilde{f}_t(\theta) = \sum_{l=0}^{\tau} q_l \cos(l\theta).$$

We are now interested in a choice of the coefficients q_l so that $\tilde{f}_t(\theta)$ approximates the original eigenvalue function $g^t(\theta) = \cos^t(\theta)$. This is exactly what the Chebyshev expansion does. Indeed, from [116] we know that

$$x^t = \sum_{l=0}^{t} p_l T_l(x),$$

where p_l represents the probability that $|X_t| = l$ for X_t a t step random walk on the infinite line, starting in the origin:

$$p_l = \mathbb{P}(|X_t| = l) = \begin{cases} \frac{1}{2^{t-1}} \binom{t}{\frac{t-l}{2}} & l > 0,\ t = l \bmod 2, \\ \frac{1}{2^t} \binom{t}{\frac{t}{2}} & l = 0,\ t = 0 \bmod 2, \\ 0 & \text{elsewhere.} \end{cases} \tag{8.2}$$

Again using the geometric definition of the Chebyshev polynomials, $T_t(\cos(\theta)) = \cos(t\theta)$, and setting $x = \cos(\theta)$, this implies that g^t can be exactly expanded into the eigenfunctions f_t:

$$g^t(\theta) = \cos^t(\theta) = \sum_{l=0}^{t} p_l \cos(l\theta) = \sum_{l=0}^{t} p_l f_t(l\theta). \tag{8.3}$$

Using the above lemma we can now choose $q_l = p_l$ to exactly simulate the original dynamics. The problem is that in this case $\tau = t$, and implementing W_τ therefore requires $O(t)$ QW steps, giving no speedup with respect to the simple quantum simulation scheme. We can resolve this by noting that p_l approaches a normal distribution with variance $\Theta(t)$, so that we can approximate it exponentially well by its support on a $O(\sqrt{t})$ interval, as we elaborate in the below lemma.

Lemma 10. *For any $\epsilon > 0$, if $C \geq 2\ln(2/\epsilon)$ then*

$$\left|\cos^t(\theta) - \sum_{l=0}^{\lceil\sqrt{Ct}\rceil} p_l \cos(l\theta)\right| \leq \epsilon.$$

Proof. Let $t' = \lceil\sqrt{Ct}\rceil$ and $p_{>t'} = \sum_{l=t'+1}^{t} p_l$. The proof comes down to bounding the quantity $p_{>t'}$. Indeed, by (8.3) we can easily calculate that

$$\left|\cos^t(\theta) - \sum_{l=0}^{t'} p_l \cos(l\theta)\right| \leq p_{>t'},$$

so that it suffices to prove that $p_{>t'} \leq \epsilon$. We can bound $p_{>t'}$ since it represents the probability that $|X_t| > t'$ where X_t is a t step random walk X_t. By Hoeffding's inequality we know that $p_{>t'} \leq 2\exp\left(-t'^2/(2t)\right)$. For $t' = \lceil\sqrt{Ct}\rceil$ and $C \geq 2\ln\frac{2}{\epsilon}$ this shows that $p_{>\lceil\sqrt{Ct}\rceil} \leq \epsilon$, which proves the lemma. □

This lemma shows that it is possible to pointwise approximate the original eigenvalue function $\cos^t(\theta)$, up to error ϵ, using the truncated Chebyshev expansion

$$g_\tau^t(\theta) = \sum_{l=0}^{\tau} p_l \cos(l\theta)$$

for $\tau \in O(\sqrt{t}\log\frac{1}{\epsilon})$. We show this function in Figure 8.1 for $\tau = \sqrt{t}$ and $\tau = 1.2\sqrt{t}$. In the next section we show how this approximation lemma leads to our quantum fast-forwarding scheme.

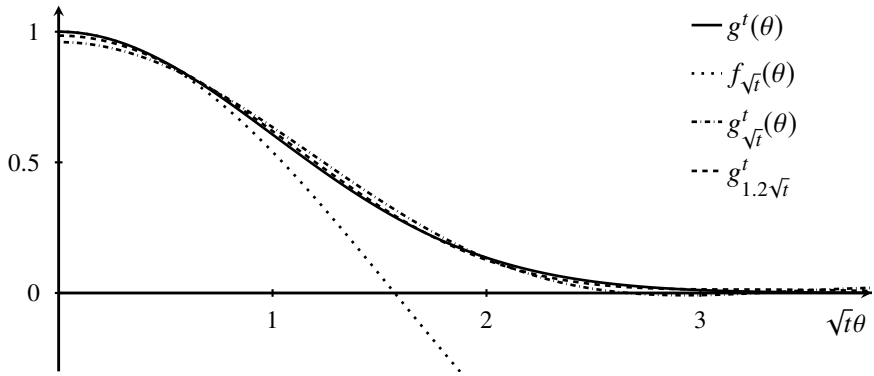


Fig. 8.1 Plot showing the different eigenvalue functions for $t = 1000$. The eigenvalue function $f_{\sqrt{t}}(\theta) = \cos(\sqrt{t}\theta)$, resulting from the Ambainis-Szegedy scheme, only approximates the original eigenvalue function $g^t(\theta) = \cos^t(\theta)$ closely for $\sqrt{t}\theta$ sufficiently small. It is clear that the truncated Chebyshev expansion $g_\tau^t(\theta)$, cutting off the Chebyshev expansion of g^t after τ terms, approximates g^t increasingly well for $\tau = \lceil\sqrt{t}\rceil$ and $\tau = \lceil 1.2\sqrt{t}\rceil$.

8.3 Quantum Fast-Forwarding Algorithm

Finally we can propose our quantum fast-forwarding algorithm, which combines the above results. It builds on the operator W_τ from Lemma 9 with q a truncation of p as defined in (8.2), so that

$$\begin{aligned} \Pi_\flat W_\tau |v, \flat\flat\rangle &= \frac{1}{1-p_{>\tau}} \sum_{l=0}^{\tau} p_l \Pi_\flat W^l |v, \flat\flat\rangle \\ &= \Big(\frac{1}{1-p_{>\tau}} \sum_{l=0}^{\tau} p_l T_l(D)|v\rangle \Big) \otimes |\flat\flat\rangle. \end{aligned} \tag{8.4}$$

Algorithm 2 Quantum Fast-Forwarding $\mathbf{QFF}(|v\rangle, P, t, \epsilon)$

Input: quantum state $|v\rangle \in \mathcal{H}_\mathcal{V}$, Markov chain P, $t \in \mathbb{N}$, $\epsilon > 0$

Do:

1: set $\epsilon' = \dfrac{\|D^t|v\rangle\|\epsilon}{2}$ and $\tau = \left\lceil \sqrt{2t \ln \frac{2}{\epsilon'}} \right\rceil$

2: initialize the registers $\mathbf{R_1 R_2 R_3}$ with the state $|v, \flat\flat\rangle$

3: apply the LCU operator W_τ as in (8.4) on $\mathbf{R_1 R_2 R_3}$

4: perform the measurement $\{\Pi_{\flat\flat}, I - \Pi_{\flat\flat}\}$

5: **if** outcome $\neq$ "$\flat\flat$" **then** output "`Fail`" and stop

Output: registers $\mathbf{R_1 R_2 R_3}$

Complexity: τ QW steps **Success probability:** $(1-\epsilon)\|D^t|v\rangle\|^2$

Theorem 5 (Quantum Fast-Forwarding). *Algorithm* $\mathbf{QFF}(|v\rangle, P, t, \epsilon)$ *outputs a state* ϵ*-close to* $|D^t v, \flat\flat\rangle$ *with success probability at least* $(1-\epsilon)\|D^t|v\rangle\|^2$*. Otherwise it outputs "*`Fail`*". The algorithm uses a number of QW steps*

$$\tau \in O\left(\sqrt{t \log \left(\frac{1}{\epsilon \|D^t|v\rangle\|} \right)} \right).$$

Proof. Let $\{(\cos\theta_k, |v_k\rangle), 1 \le k \le |\mathcal{V}|\}$ be a complete orthonormal set of eigenpairs of D. Then we can write $|v\rangle = \sum_k \alpha_k |v_k\rangle$ and the goal state $|D^t v\rangle = \sum_k \alpha_k \cos(\theta_k)^t |v_k\rangle / \|D^t|v\rangle\|$. From Lemma 9 we know that if we apply the operator W_τ on $|v, \flat\flat\rangle$, and we perform a measurement $\{\Pi_\flat, I - \Pi_\flat\}$, then we retrieve the state

$$\begin{aligned} &\frac{1}{\left\| \frac{1}{1-p_{>\tau}} \sum_{l=0}^{\tau} q_l T_l(D)|v\rangle \right\|} \Big(\frac{1}{1-p_{>\tau}} \sum_{l=0}^{\tau} q_l T_l(D)|v\rangle \Big) \otimes |\flat\flat\rangle \\ &\qquad = \frac{1}{\left\| \sum_{l=0}^{\tau} q_l T_l(D)|v\rangle \right\|} \Big(\sum_{l=0}^{\tau} q_l T_l(D)|v\rangle \Big) \otimes |\flat\flat\rangle \end{aligned}$$

with success probability $\| \frac{1}{1-p_{>\tau}} \sum_{l=0}^{\tau} p_l T_l(D)|v\rangle \|^2$. We will denote the state $|\psi_\tau\rangle = \sum_{l=0}^{\tau} q_l T_l(D)|v\rangle$. By the approximation from Lemma 10 we know that if $\tau = \left\lceil \sqrt{2t \ln \frac{2}{\epsilon'}} \right\rceil$ then $|\psi_\tau\rangle$ will be ϵ'-close to $D^t|v\rangle$:

$$\left\| |\psi_\tau\rangle - D^t|v\rangle \right\| = \sqrt{\sum_k \left| \sum_{l=0}^{\tau} q_l \cos(l\theta) - \cos^t(\theta_k) \right|^2 \cdot |\alpha_k|^2}$$
$$\leq \sqrt{\epsilon'^2 \sum_k |\alpha_k|^2} = \epsilon'.$$

From (7.6) in Example 12 we know that for any two nonzero vectors it holds that $\left\| \frac{v}{\|v\|} - \frac{w}{\|w\|} \right\| \leq \frac{2\|v-w\|}{\|w\|}$, so we can bound

$$\left\| \frac{|\psi\rangle}{\||\psi\rangle\|} - \frac{D^t|v\rangle}{\|D^t|v\rangle\|} \right\| \leq \frac{2\epsilon'}{\|D^t|v\rangle\|} = \epsilon,$$

using the fact that we chose $\epsilon' = \|D^t|v\rangle\|\epsilon/2$. We can now also bound the success probability using the reverse triangle inequality:

$$\left\| \frac{1}{1-p_{>\tau}} \sum_{l=0}^{\tau} p_l T_l(D)|v\rangle \right\|^2 \geq \left\| \sum_{l=0}^{\tau} p_l T_l(D)|v\rangle \right\|^2$$
$$\geq (\|D^t|v\rangle\| - \epsilon')^2 \geq (1-\epsilon)\|D^t|v\rangle\|^2.$$

By Lemma 9 we know that implementing the operator W_τ requires a number of QW steps

$$\tau = \left\lceil \sqrt{2t \ln\left(\frac{4}{\epsilon\|D^t|v\rangle\|}\right)} \right\rceil \in O\left(\sqrt{t \log\left(\frac{1}{\epsilon\|D^t|v\rangle\|}\right)}\right). \qquad \square$$

This theorem establishes our algorithm for quantum fast-forwarding Markov chains. It winds back the quantum walk speedup of the Ambainis-Szegedy scheme to the original problem of quantum simulation, showing that we can achieve the same quadratic acceleration that is characteristic for this scheme. The success probability is proportional to $\|D^t|v\rangle\|^2$, which will be small when the Markov chain spreads out mass over a large set.

Notice that if we choose $\tau = t$ in the above algorithm, then the Chebyshev expansion in (8.4) will not be truncated, and $\Pi_\flat W_\tau |v, \flat\flat\rangle = (D^t|v\rangle) \otimes |\flat\flat\rangle$. As a consequence, there will be no error of approximation $\epsilon = 0$ and the algorithm will output the exact state $|D^t v\rangle$ with success probability $\|D^t|v\rangle\|^2$, requiring $O(t)$ QW steps. This retrieves the features of the original quantum simulation scheme $\mathbf{QS}(|v\rangle, P, t)$. In the below example we illustrate the QFF scheme by showing how we can create a superposition over the nodes of a graph faster than using quantum simulation.

Example 13 (Creating Superpositions with QFF). In the previous chapter, Example 12, we demonstrated how we can use the quantum simulation of a symmetric Markov chain $P = D$ to create a uniform superposition over the nodes of a graph. For $t \in \Theta(\tau \log \frac{1}{\epsilon|\mathcal{V}|})$, with τ the mixing time of P, the quantum simulation scheme over t steps returns a state ϵ-close to a uniform superposition with a success probability $\Theta(1/|\mathcal{V}|)$, where $|\mathcal{V}|$ is the total number of nodes. The expected runtime is therefore $\Theta(\tau|\mathcal{V}| \log \frac{|\mathcal{V}|}{\epsilon})$.

Using QFF we can easily improve the expected runtime to

$$\Theta\Big(\sqrt{\tau}|\mathcal{V}|\log\frac{|\mathcal{V}|}{\epsilon}\Big).$$

Indeed, starting from any node $j \in \mathcal{V}$, we can run $\mathbf{QFF}(|j\rangle, P, t, \epsilon')$ for $t = \tau(\|\pi\|\epsilon/8)$ and $\epsilon' = \epsilon/2$. This will generate an $(\epsilon/2)$-approximation of the state $|P^t j\rangle$, which will be $(\epsilon/2)$-close to the uniform superposition $|\pi\rangle$. By the triangle inequality, the generated state will be ϵ-close to $|\pi\rangle$. By a similar reasoning as in Example 12 the success probability $\Theta(\|P^t|j\rangle\|^2) \in \Theta(1/|\mathcal{V}|)$, and by Theorem 5 the scheme requires a number of QW steps in

$$O\left(\sqrt{\tau\Big(\frac{\|\pi\|\epsilon}{8}\Big)\log\frac{2|\mathcal{V}|}{\epsilon}}\right) \in O\left(\sqrt{\tau}\log\frac{|\mathcal{V}|}{\epsilon}\right).$$

The total expected runtime to generate an ϵ-approximation of $|\pi\rangle$ therefore becomes $O\big(\sqrt{\tau}|\mathcal{V}|\log\frac{|\mathcal{V}|}{\epsilon}\big)$. On for instance the cycle $\mathbb{Z}_N$, having mixing time $\tau \in \Theta(N^2)$, this comes down to $O\big(N^2\log\frac{N}{\epsilon}\big)$ QW steps. △

We conclude this section with some remarks on the algorithm:

- Most of the previous work using the Ambainis-Szegedy scheme builds on a quadratic acceleration of the limit behavior of some Markov chain [3, 21, 22, 41, 42]. Our result complements this by showing that it is also possible to capture the intermediate dynamics, without losing the quadratic speedup. In Chapter 10 we will make explicit use of this for search algorithms and state generation.
- We simulate the symmetric discriminant dynamics $D^t|v\rangle$ rather than the Markov chain dynamics $P^t|v\rangle$. If P is symmetric, then $D = P$ so that these dynamics are equivalent. If P is reversible, which is commonly the case, then these dynamics are also very closely related, as we already discussed in Chapter 1. Indeed if π is the stationary distribution of a reversible Markov chain P, then $D = \mathrm{diag}(\sqrt{\pi})^{-1}P\,\mathrm{diag}(\sqrt{\pi})$ and therefore

  $$D^t = \mathrm{diag}(\sqrt{\pi})^{-1}P^t\,\mathrm{diag}(\sqrt{\pi}). \tag{8.5}$$

  As a consequence, the dynamics $D^t|v\rangle$ and $P^t|v\rangle$ are equivalent up to a rescaling. This implies that in particular P and D have the same eigenvalues.
- In [117, 118] the Ambainis-Szegedy scheme was generalized from the symmetric discriminant matrix of a Markov chain to more general Hermitian matrices H, for which $H = H^\dagger$. In particular, this allows to quantize the Hamiltonian that describes the dynamics of a continuous time quantum system, leading to efficient discrete time schemes for the simulation of these continuous dynamics [111, 113, 118]. Our results carry over, verbatim, to this setting as we will discuss in Chapter 10.
- Childs, Kothari and Somma [119] recently used a similar truncation of the Chebyshev expansion to implement the function D^{-1} in their quantum algorithm for systems of linear equations. In their algorithm D represents a Hermitian linear system matrix, rather than the discriminant matrix of a Markov chain.

8.4 Grover Speedup

Both quantum simulation $\mathbf{QS}(|v\rangle, P, t)$ and quantum fast-forwarding $\mathbf{QFF}(|v\rangle, P, t, \epsilon)$ approximate the quantum state $|D^t v\rangle$ with probability $(1 - O(\epsilon))\||D^t|v\rangle\|^2$. As a consequence, the expected number of runs needed to create the goal state will scale as $1/\||D^t|v\rangle\|^2$. In this section we show that we can use the Grover quantum search algorithm to quadratically improve this dependency to $1/\||D^t|v\rangle\|$, provided that we can *reflect* around the initial state $|v\rangle$, that is, implement the reflection operator

$$R_v = 2|v\rangle\langle v| - I.$$

Such a reflection is generally considered feasible if $|v\rangle$ is some known basis state, as is for instance the case if $|v\rangle = |j\rangle$ for j a given initial node of a graph (see [46] for details. Contrarily if $|v\rangle$ denotes a general superposition, such as the uniform superposition $|\pi\rangle$ over the nodes of a graph, then it is not a priori clear whether we can implement the reflection R_v.

Grover Search Algorithm

Grover's quantum search algorithm, invented by Lov Grover in 1996 [100], is a quantum algorithm for searching an unsorted list quadratically faster than a classical computer. It is both one of the earliest and one of the most widely used subroutines in the field of quantum computing. Preceding and inspiring the Watrous and Ambainis-Szegedy quantum walk schemes, it is similarly constructed by alternating two reflections. The use of quantum walks for search problems [3, 14, 18, 22] is therefore commonly seen as a generalization of Grover's algorithm.

Whereas Grover considered the search of a single element in some space, we will describe and use a generalized form of his algorithm where we search for projections in some subspace. This generalization is called *amplitude amplification*, invented by Brassard and Høyer in 1997 [120], and independently by Grover in 1998 [121]. Consider a general quantum state $|\psi\rangle$ and a projector $\Pi_\flat$ such that

$$|\psi\rangle = \Pi_\flat|\psi\rangle + (I - \Pi_\flat)|\psi\rangle,$$

and assume that we wish to retrieve the component $\Pi_\flat|\psi\rangle$. It will prove useful to think about the complex plane spanned by the initial state $|\psi\rangle$ and the goal state $\Pi_\flat|\psi\rangle$, or equivalently by $(I - \Pi_\flat)|\psi\rangle$ and $\Pi_\flat|\psi\rangle$, depicted in Figure 8.2.
Amplitude amplification builds on two reflection operators:

$$R_\psi = 2|\psi\rangle\langle\psi| - I,$$

reflecting around the initial state $|\psi\rangle$, and

$$R_\flat = 2\Pi_\flat - I,$$

reflecting around the flat subspace $\mathrm{Im}(\Pi_\flat)$. As is discussed in [122], the dynamics of $|\psi\rangle$ under these reflections are restricted to the plane spanned by $(I - \Pi_\flat)|\psi\rangle$ and $\Pi_\flat|\psi\rangle$. We depict both reflections in Figure 8.2. The product $-R_\psi R_\flat$ applies a rotation in that plane by an angle 2θ, where $\theta \in (0, \pi/2]$ is defined by

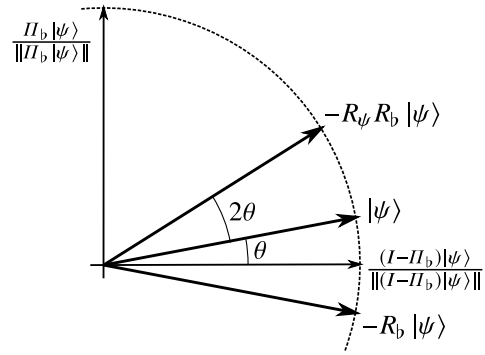


Fig. 8.2 Illustration of the complex plane spanned by $|\psi\rangle$ and $\Pi_\flat|\psi\rangle$. The reflections $R_\flat$ and R_ψ invoke a rotation in this plane. This rotation allows to turn the initial state $|\psi\rangle$ into the goal state $\Pi_\flat|\psi\rangle/\|\Pi_\flat|\psi\rangle\|$.

$$\sin\theta = \|\Pi_\flat|\psi\rangle\|.$$

Repeating this operation allows to rotate the initial state $|\psi\rangle$ close to the state $\Pi_\flat|\psi\rangle/\|\Pi_\flat|\psi\rangle\|$. This is detailed in the following proposition. We include their proof for completeness.

Proposition 16 (Amplitude amplification [122]). *Let θ be defined as above, and let $m = \lfloor \pi/(4\theta) \rfloor$. If we apply the operator $(-R_\psi R_\flat)$ m times on the state $|\psi\rangle$, and we perform a measurement $\{\Pi_\flat, I - \Pi_\flat\}$, then we find back the state $\Pi_\flat|\psi\rangle/\|\Pi_\flat|\psi\rangle\|$ with probability at least* $\max\{1 - \sin^2\theta, \sin^2\theta\} \geq 1/2$.

Proof. We can write

$$|\psi\rangle = \sin\theta \frac{\Pi_\flat|\psi\rangle}{\|\Pi_\flat|\psi\rangle\|} + \cos\theta \frac{(I-\Pi_\flat)|\psi\rangle}{\|(I-\Pi_\flat)|\psi\rangle\|}.$$

By the above reasoning, the operator $-R_\psi R_\flat$ performs a rotation in the plane by an angle 2θ. So if we apply the rotation operator $-R_\psi R_\flat$ m times, we find back the state

$$(-R_\psi R_\flat)^m|\psi\rangle = \sin((1+2m)\theta)\frac{\Pi_\flat|\psi\rangle}{\|\Pi_\flat|\psi\rangle\|} + \cos((1+2m)\theta)\frac{(I-\Pi_\flat)|\psi\rangle}{\|(I-\Pi_\flat)|\psi\rangle\|}.$$

If we perform a measurement $\{\Pi_\flat, I - \Pi_\flat\}$ on this state, we retrieve the state $\Pi_\flat|\psi\rangle/\|\Pi_\flat|\psi\rangle\|$ with probability $\sin^2((1+2m)\theta)$. Therefore we prove the theorem by showing that $\sin^2((1+2m)\theta) \geq \max\{1-\sin^2\theta, \sin^2\theta\}$.

To do so, note that $\sin^2((1+2m)\theta) = 1$ if m equals $\tilde{m} = \pi/(4\theta) - 1/2$. However m has to be an integer, so we set $m = \lfloor \pi/(4\theta) \rfloor$, and therefore $|m - \tilde{m}| \leq 1/2$. Now if $\theta > \pi/4$, then $m = 0$ and the success probability is $\sin^2\theta > 1 - \sin^2\theta$, which proves the statement. If $\theta \leq \pi/4$, then $\sin^2\theta \leq 1 - \sin^2\theta$, and so we must prove that $\sin^2((1+2m)\theta) \geq 1 - \sin^2\theta$. Thereto we use that

$$|(1+2m)\theta - (1+2\tilde{m})\theta| \leq 2\theta|m - \tilde{m}| \leq \theta,$$

which implies that $(1+2m)\theta = \frac{\pi}{2} + x$ for some x with $|x| \leq \theta$. We can therefore bound

$$|\cos((1+2m)\theta)| = |\cos(\pi/2 + x)| \leq |\sin x| \leq |\sin\theta|.$$

As a consequence, $\sin^2((1+2m)\theta) \geq 1 - \sin^2\theta$. This finalizes the proof. □

Since $\theta \geq \sin\theta = \|\Pi_\flat|\psi\rangle\|$, this proposition allows to retrieve the good part $\Pi_\flat|\psi\rangle/\|\Pi_\flat|\psi\rangle\|$ with high probability after $\leq \frac{\pi}{4\|\Pi_\flat|\psi\rangle\|}$ applications of the reflections R_ψ and $R_\flat$. Contrarily, performing a measurement $\{\Pi_\flat, I - \Pi_\flat\}$ on the initial state would have returned the good part with probability $\|\Pi_\flat|\psi\rangle\|^2$, thus requiring $1/\|\Pi_\flat|\psi\rangle\|^2$ expected copies of the initial state to retrieve the good part. As we will see later, it is often the case that the reflection R_ψ requires the same amount of resources or time as creating a new copy $|\psi\rangle$. The amplitude amplification scheme therefore requires quadratically less resources or time to retrieve the good part, as compared to simply measuring the state.

We note that implementation of amplitude amplification requires a good estimate of the initial success probability $\|\Pi_\flat|\psi\rangle\|^2$ to determine m. If m is chosen either too small or too large, then the guarantees on the success probability are lost, a problem often referred to as the "soufflé problem". A remedy is however proposed in [122], in which iteratively different guesses for m are used. They show this approach also yields a success probability $\geq 1/2$, while still applying the operator $(-R_\psi R_\flat)$ only $O(1/\|\Pi_\flat|\psi\rangle\|)$ times. For clarity we will implicitly rely on this fact, and throughout assume that we can appropriately determine the parameter m.

The following is an elementary illustration of the search algorithm.

Example 14 (Boolean Formula). Consider some complicated boolean function f on N-bit strings $x \in \{0,1\}^N$, so that $f : \{0,1\}^N \to \{0,1\}$. We wish to determine whether this function can be satisfied, that is, whether there exists an $x \in \{0,1\}^N$ such that $f(x) = 1$. Classically we can always do a brute force search over the N-bit strings. If the function is satisfiable then there exist $M > 0$ satisfying strings in $\{0,1\}^N$. We can then randomly generate an N-bit string x and check it by evaluating $f(x)$. This strategy will return a satisfying string after $\Theta(2^N/M)$ expected trials and evaluations of f.

Using a quantum computer that is able to implement Grover search, we can do this quadratically faster. Thereto we first construct the superposition over N-bit strings

$$|\psi\rangle = \frac{1}{\sqrt{2^N}} \sum_{x\in\{0,1\}^N} |x\rangle.$$

If we can transform $|\psi\rangle$ into the state $|\psi_M\rangle = \frac{1}{\sqrt{M}} \sum_{f(y)=1} |y\rangle$, then a simple measurement in the standard basis will return a satisfying string. By the above proposition we can use amplitude amplification to do so, building on the reflections $R_\psi = 2|\psi\rangle\langle\psi| - I$ around the initial state and $R_\flat = 2\Pi_\flat - I$ around the subspace of satisfying strings, where the projector $\Pi_\flat$ on this subspace is defined as

$$\Pi_\flat = \sum_{f(y)=1} |y\rangle\langle y|.$$

See for instance [100] and [46] for details on the implementation of these reflections, showing most importantly that the rotation $(-R_\psi R_\flat)$ can be implemented using a single evaluation of the boolean function f. Since

$$\sin\theta = \|\Pi_\flat|\psi\rangle\| = \Big\|\frac{1}{\sqrt{2^N}}\sum_{f(y)=1}|y\rangle\Big\| = \sqrt{\frac{M}{2^N}},$$

we see from Proposition 16 that applying the operator $(-R_\psi R_k)$ $\Theta(\sqrt{2^N/M})$ times, and performing a measurement $\{\Pi_\flat, I-\Pi_\flat\}$, returns the state $|\psi_M\rangle$ with probability $\geq 1/2$. As a consequence, we can retrieve a satisfying string with $\Theta(\sqrt{2^N/M})$ expected evaluations of f, which quadratically improves the number of evaluations in the classical case. △

Improved Algorithms

We can use amplitude amplification to quadratically improve the success probability of our QFF algorithm. Similar to the above exposition, the QFF algorithm applies the operator W_τ on some input state $|v,\flat\flat\rangle$ ($\equiv |v,\flat\rangle$, we will use the shorthand $\flat$ to denote $\flat\flat$)

$$|\psi\rangle = W_\tau|v,\flat\rangle = \Pi_\flat W_\tau|v,\flat\rangle + (I-\Pi_\flat)W_\tau|v,\flat\rangle,$$

and we wish to retrieve the part $\Pi_\flat W_\tau|v,\flat\rangle$. In the previous chapter we did so by performing a measurement $\{\Pi_\flat, I-\Pi_\flat\}$, retrieving the good part with probability $\|\Pi_\flat|\psi\rangle\|^2 = \|\Pi_\flat W_\tau|v,\flat\rangle\|^2$.

Amplitude amplification shows that we can boost this success probability to a constant by implementing the operator $(-R_\psi R_\flat)$ for a number of times in

$$\Theta(1/\|\Pi_\flat|\psi\rangle\|).$$

If on this state we perform a measurement $\{\Pi_\flat, I-\Pi_\flat\}$ then we find back the good part with probability $\geq \max\{1-\|\Pi_\flat|\psi\rangle\|^2, \|\Pi_\flat|\psi\rangle\|^2\} \geq 1/2$. Implementing the operator $(-R_\psi R_\flat)$ however has a cost. Whereas the reflection $R_\flat = 2\Pi_\flat - I$ is considered an elementary operation on the basis states, with a negligible cost, the cost of implementing the reflection $R_\psi = 2|\psi\rangle\langle\psi| - I$, with $|\psi\rangle$ some constructed superposition over basis states, should be taken into account.

Lemma 11. *The operator R_ψ can be implemented using 2τ QW steps and a reflection R_v around the initial state $|v,\flat\rangle$.*

Proof. We can rewrite the reflection $R_\psi = 2|\psi\rangle\langle\psi| - I$ as follows:

$$\begin{aligned} 2|\psi\rangle\langle\psi| - I &= 2W_\tau|v,\flat\rangle\langle v,\flat|W_\tau^\dagger - I \\ &= W_\tau(2|v,\flat\rangle\langle v,\flat| - I)W_\tau^\dagger = W_\tau R_v W_\tau^\dagger. \end{aligned}$$

Therefore we can implement the reflection by implementing the operators W_τ and $W_\tau^\dagger$, and the reflection around the initial state R_v. To implement the operator $W_\tau^\dagger$, we recall that $W_\tau = V_q^\dagger\Big[\sum_{l=0}^\tau |l\rangle\langle l|\otimes(R_\flat U)^l\Big]V_q$ and so

$$W_\tau^\dagger = V_q^\dagger\Big[\sum_{l=0}^\tau |l\rangle\langle l|\otimes(UR_\flat)^l\Big]V_q.$$

Here we also used the fact that $U = V^\dagger SV$ with $S = S^\dagger$, as in (7.3), so that $U^\dagger = U$. We already discussed in Lemma 9 that the controlled operator $\sum_{l=0}^\tau |l\rangle\langle l|\otimes UR_\flat$

can be implemented in τ QW steps. As a consequence, both W_τ and $W_\tau^\dagger$ can be implemented in τ QW steps. □

It follows that the total operator $(-R_\psi R_\flat)$ can be implemented using $2\tau R$ QW steps, a reflection around the initial state $|v, \flat\rangle$, and a number of elementary operations. In many cases the initial state will be an elementary basis state, so that the reflection R_v will also be elementary, and the main cost boils down to $2R\tau$ QW steps. We can now propose the improved QFF algorithm, **QFFg**, in Algorithm 3.

Algorithm 3 Quantum Fast-Forwarding with Reflections **QFFg**$(|v\rangle, P, t, \epsilon)$

Input:
quantum state $|v\rangle \in \mathcal{H}_\mathcal{V}$, Markov chain P, $t \in \mathbb{N}$, $\epsilon > 0$
Do:
1: set $\epsilon' = \dfrac{\|D^t|v\rangle\|\epsilon}{2}$ and $\tau = \left\lceil \sqrt{2t \ln \frac{2}{\epsilon'}} \right\rceil$
2: set $m = \lfloor \pi/(4\theta) \rfloor$, where $0 < \theta \leq \pi/2$ s.t. $\sin\theta = \|\Pi_\flat W_\tau |v\rangle\|$
3: initialize registers $\mathbf{R_1R_2R_3}$ with the state $|v, \flat\flat\rangle$
4: apply the LCU operator W_τ on $\mathbf{R_1R_2R_3}$
5: apply the operator $(-R_\psi R_\flat)^m = (-W_\tau R_v W_\tau^\dagger R_\flat)^m$ ▷ *Amplitude Amplification*
6: perform the measurement $\{\Pi_{\flat\flat}, I - \Pi_{\flat\flat}\}$
7: **if** outcome ≠ "$\flat\flat$" **then** output "`Fail`" and stop
Output: registers $\mathbf{R_1R_2R_3}$

Complexity: $(2m+1)\tau$ QW steps **Success Probability:** $\frac{1}{2}$

Theorem 6. *The QFFg algorithm* **QFFg**$(|v\rangle, P, t, \epsilon)$ *outputs a state ϵ-close to $|D^t v\rangle$ with success probability at least $1/2$. Otherwise, it outputs "`Fail`". The algorithm uses $\Theta(1/\|D^t|v\rangle\|)$ reflections around the initial state, and a number of QW steps*

$$(2m+1)\tau \in O\left(\frac{\sqrt{t}}{\|D^t|v\rangle\|}\sqrt{\log\left(\frac{1}{\epsilon\|D^t|v\rangle\|}\right)}\right).$$

Proof. The algorithm straightforwardly applies the amplitude amplification scheme on the state $W_\tau|v, \flat\rangle$. From Proposition 16 we know that the scheme has a success probability $\geq \max\{1 - \sin^2\theta, \sin^2\theta\} \geq 1/2$. The number of QW steps for implementing W_τ and $(-R_\psi R_\flat)^m$ is τ respectively $2m\tau$. We know that $m \in O(1/\|\Pi_\flat W_\tau|v, \flat\rangle\|)$, and from the proof of Theorem 5 we can bound $\|\Pi_\flat W_\tau|v, \flat\rangle\| \geq \|D^t|v\rangle\| - \epsilon' = (1-\epsilon/2)\|D^t|v\rangle\| \in \Theta(\|D^t|v\rangle\|)$ for all $\epsilon < 1/2$. □

We already mentioned below Theorem 5 that if we choose $\tau = t$ in the QFF scheme, so that the Chebyshev expansion is not truncated, then the error term vanishes: $\epsilon = 0$. A similar reasoning shows that if we choose $\tau = t$ in the above QFFg scheme, then the scheme exactly returns the state $|D^t v\rangle$ with a success probability $\geq 1/2$ in $(2m+1)t \in O(t/\|D^t|v\rangle\|)$ QW steps. Hence we also find the following result, which improves on the quantum simulation scheme from Chapter 4.

Proposition 17 (Quantum Simulation with Reflections). *Algorithm 3 for $\tau = t$ outputs the state $|D^t v\rangle$ with success probability at least $1/2$. Otherwise it outputs "*`Fail`*". The algorithm uses $\Theta(1/\||D^t|v\rangle\|)$ reflections around the initial state, and a number of QW steps*

$$(2m+1)\tau \in O\left(\frac{t}{\||D^t|v\rangle\|}\right).$$

For comparison, the original scheme required $\Theta(\||D^t|v\rangle\|^{-2})$ expected copies of the initial state and $\Theta(t\||D^t|v\rangle\|^{-2})$ expected QW steps to succeed.

Example 15 (Creating Superpositions with QFFg). Again consider a symmetric Markov chain $P = D$ with mixing time and (uniform) stationary distribution π. In the previous section we demonstrated how we can apply $\mathbf{QFF}(|j\rangle, P, \tau, \epsilon)$, for some initial node j, to create an ϵ-approximation of the uniform superposition $|\pi\rangle$ in $O\big(\sqrt{\tau}|\mathcal{V}|\log\frac{|\mathcal{V}|}{\epsilon}\big)$ QW steps. Using the above scheme, assuming that we can reflect around the initial state $|j\rangle$, we can improve this to $O\big(\sqrt{\tau|\mathcal{V}|}\log\frac{|\mathcal{V}|}{\epsilon}\big)$ QW steps. Indeed, using an analogous reasoning as in Example 13, we can implement $\mathbf{QFFg}(|j\rangle, P, t, \epsilon')$, for $\epsilon' = \epsilon/2$ and $t = \tau(\|\pi\|\epsilon/8)$, which returns an ϵ-approximation of $|\pi\rangle$. Since $\|D^t|j\rangle\| \in \Theta(1/\sqrt{|\mathcal{V}|})$, the number of QWs required is $O\big(\sqrt{\tau|\mathcal{V}|}\log\frac{|\mathcal{V}|}{\epsilon}\big)$. On for instance the cycle $\mathbb{Z}_N$, having $\tau \in \Theta(N^2)$, this requires $O\big(N^{3/2}\log\frac{N}{\epsilon}\big)$ QW steps.

We show in Appendix A that this scheme can be generalized to a reversible non-symmetric Markov chain P with a non-uniform stationary distribution π. Starting from a node j, the generalized scheme returns an ϵ-approximation of the quantum state $|\sqrt{\pi}\rangle = \sum\sqrt{\pi(i)}|i\rangle$ in an expected number of QW steps in

$$O\left(\sqrt{\frac{\tau}{\pi(j)}}\log\frac{\epsilon}{\|\pi(j)\|}\right).$$

For π the uniform distribution this reduces to the above scheme. Building on work by Krovi et al [22] we also give an improved algorithm which returns an ϵ-approximation of $|\sqrt{\pi}\rangle$, starting from a node j, in an expected number of QW steps in

$$O\left(\sqrt{\frac{\mathrm{HT}(j)}{\epsilon}}\log\frac{1}{\epsilon}\right).$$

Here $\mathrm{HT}(j)$ is the hitting time of node j, defined as the expected number of steps of P before element j is hit, starting from the stationary distribution over the complement of j. It is known that $\mathrm{HT}(j) \in O\big(\frac{\tau}{\pi(j)}\big)$ [63], and in some cases $\mathrm{HT}(j)$ is much smaller. As a consequence, if we neglect the error term, this improves over the former scheme. On for example the cycle $\mathbb{Z}_N$, $\mathrm{HT}(j) \in \Theta(N^2)$, and hence this improved scheme requires $O\Big(\frac{N}{\sqrt{\epsilon}}\log\frac{1}{\epsilon}\Big)$. Since the diameter of $\mathbb{Z}_N$ is $\Theta(N)$, the dependence on N is optimal. △

Chapter 9
Quantum Property Testing

In this chapter we discuss the use of QFF for property testing on graphs: given query access to a graph, property testing aims to determine whether it has a certain property, or whether it is far from having this property. For example, is a given graph bipartite, or is it far from any bipartite graph? This is a relaxation as compared to effectively deciding whether a graph is bipartite or not (but possibly very close to bipartite). The property testing relaxation typically allows for algorithms that work in sublinear time, i.e., scale as $o(N)$ with N the number of nodes in the graph. This in contrast to the complexity of deciding properties, which typically requires a number of queries at least linear in the graph size. The field of property testing was largely initiated by the work of Goldreich, Goldwasser and Ron [102], in which they showed how graph properties such as bipartiteness and k-colorability could be tested with a number of queries independent of the graph size.

We will consider property testers for the expansion and the clusterability of graphs. We start by discussing the expansion tester of Goldreich and Ron (GR), which they presented in later work [103], and we prove how QFF allows to accelerate this tester very naturally. The expansion of a graph closely ties to the graph conductance, and hence forms a measure for the random walk mixing time over the graph. The idea behind the GR tester is to run a number of random walks and count the number of pairwise collisions between the end points. If a random walk is congested in some low expansion set, than this number will be greater than when the random walk spreads out efficiently, thus forming a measure for the spreading behavior and expansion of the random walk. To speed up this task, we propose the use of QFF to quantum simulate a random walk, and then use a tool called quantum amplitude estimation to estimate the 2-norm of the random walk probability distribution. This 2-norm will similarly be large if the random walk congests and small otherwise, thus allowing to detect whether the random walk is able to efficiently spread out or not. In preceding work, Ambainis, Childs and Liu [123] have also used quantum walks to speed up the GR tester, be it in an indirect way, using a very different approach. Interestingly this leads to a complementary speedup to ours. In the final section we discuss the more recent line of algorithms for testing graph clusterability, forming a natural generalization of the work of Goldreich and Ron. We discuss how these techniques make use of algorithms for classifying nodes in clusters, and show how QFF allows to accelerate these algorithms. Such node

classification is of relevance beyond the setting of property testing, allowing for instance nearest-neighbor classification of nodes in a learning problem.

We remark that work by Valiant and Valiant [124] shows that estimating the distance in 2-norm between given probability distributions is much easier and more stable than estimating the distance in 1-norm, which would otherwise be the natural choice. This underlies the fact that many graph property testing algorithms estimate the 2-distance between random walk distributions. QFF, and quantum simulation more generally, allows to cast a probability distribution p as a quantum state $|p\rangle = p/\|p\|$, which is naturally associated to the 2-norm. As a consequence, QFF very naturally leads to quantum algorithms for estimating the 2-norm distance between random walk distributions, directly leading to the quantization and speedup of many existing graph property testers that we discuss in this chapter.

9.1 Classical Expansion Tester

To formalize the concept of a graph property tester, we must introduce the notion of *oracle or query access* to a graph as used throughout the literature on property testing. Query access to an N-node undirected graph with degree bound d means that we can query the graph with a string $(v, i) \in [N] \times [d]$, upon which we receive either the i-th neighbor $w \in [N]$ of v, or a special symbol in case v has less than i neighbors. Clearly this model allows to perform a random walk over the graph. In addition, and in contrast to our former work in the first part of this thesis, this also allows to generate a uniformly random node by simply generating a random number in $[N]$, identifying one of the nodes.

Given such query access to a graph, the task of a property tester is to determine whether the graph has a certain property or is far from any graph having that property. To formalize what this means, we introduce a distance measure between graphs. We define the distance between two graphs G and G' as the number of edges that have to be added or removed from G to transform it into G'. With $\mathcal{E}$ and $\mathcal{E}'$ the edge sets of G resp. G', this equals the size of the symmetric difference $|\mathcal{E} \triangle \mathcal{E}'|$. We say that two N-node graphs G and G' with degree bound d are ϵ-far from each other if $|\mathcal{E} \triangle \mathcal{E}'| \geq \epsilon N d$. Given a graph and a parameter ϵ, a property tester should distinguish between the graph having a certain property, and the graph being at least ϵ-far from any graph with that property (i.e., the distance between the graph and any graph with that property is $\geq \epsilon N d$). When given a graph that is neither, the algorithm can output whatever. The efficiency of a tester is measured by the number of queries it makes to the graph: a good tester should make as few queries to the graph as possible. For the bounded degree graphs and algorithms that we consider, this will be equivalent to minimizing the number of random walk steps performed on the graph.

Goldreich and Ron [103] studied a property tester for the expansion of a graph[1]. The expansion or vertex expansion of a graph $G = (\mathcal{V}, \mathcal{E})$ is defined by

[1] They actually studied a property tester for the spectral gap of a graph, for which currently there is no known sublinear algorithm. All follow-up work however, as well as our work, considers the closely-related graph expansion.

$$\min_{|\mathcal{S}|\leq|\mathcal{V}|/2} |\partial\mathcal{S}^c|/|\mathcal{S}|,$$

where $\partial\mathcal{S}^c$ denotes the outer boundary of $\mathcal{S}$, i.e., the set of nodes in $\mathcal{S}^c$ that have an edge going to $\mathcal{S}$. This quantity is closely related to the random walk conductance that we have introduced in Chapter 3 – for bounded degree graphs it even holds that the vertex expansion is of the same order as the conductance. For some given parameter Υ, an expansion tester should determine whether a given graph has expansion $\geq \Upsilon$, or whether it is at least ϵ-far from any such graph. GR, and the subsequent literature [123, 125–127], have relaxed this setting somewhat. They propose the following definition:

Definition. *An algorithm is a* $(\Upsilon, \epsilon, \mu)$ *expansion tester if there exists a constant* $c > 0$*, possibly dependent on* d*, such that given parameters* N*,* d*, and query access to an* N*-node graph with degree bound* d *it holds that*

- *if the graph has expansion* $\geq \Upsilon$*, then the algorithm outputs "*`accept`*" with probability at least* $2/3$*,*
- *if the graph is* ϵ*-far from any graph having expansion* $\geq c\mu\Upsilon^2$*, then the algorithm outputs "*`reject`*" with probability at least* $2/3$*.*

In the strict setting of property testing, the expression "$\geq c\Upsilon^2$" in the second bullet should be replaced by "$\geq \Upsilon$". Although unproven, the relaxation in this definition seems necessary to allow for efficient (sublinear) testing. GR [128] conjectured that the below Algorithm 4 is a $(\Upsilon, \epsilon, \mu)$ expansion tester. They also proved that any classical expansion tester must make at least $\Omega(\sqrt{N})$ queries to the graph.

Algorithm 4 Graph Expansion Tester

Input: parameters N and d; query access to N-node graph G with degree bound d; expansion parameter Υ; accuracy parameter ϵ; running time parameter $\mu < 1/4$
Do: do the following $T \in \Theta(\epsilon^{-1})$ times

1: select a uniformly random starting vertex s
2: perform $m \in \Theta(N^{1/2+\mu})$ independent random walks of length $t \in \Theta(d^2\Upsilon^{-2}\log N)$, starting in s
3: count number of pairwise collisions between the endpoints of the m random walks
4: if the count is greater than $M \in \Theta(N^{2\mu})$, abort and output "`reject`"

Output: if no "`reject`", output "`accept`"

Complexity: $\Theta(N^{1/2+\mu}\Upsilon^{-2}\epsilon^{-1}\log N)$ QW steps

The intuition behind the algorithm is very clear. It builds on the use of a random walk P on the given graph, which starting from a node v jumps to any of its $d(v)$ neighbors with a probability $1/(2d)$, and stays put otherwise:

$$P(u,v) = \begin{cases} 1/(2d) & (v,u) \in E \\ 1 - d(v)/(2d) & u = v \\ 0 & \text{elsewhere.} \end{cases} \tag{9.1}$$

This walk is lazy and symmetric, and hence converges to the uniform distribution. We can easily implement it in the oracle model: at some node v, with probability

$1/2$ we pick a random integer $i \in [d]$, we then query the graph with the string (v, i). If we receive a neighbor of v then we move to this neighbor, in any other case we stay on node v. If the graph has a vertex expansion Υ, then the conductance of this random walk, as defined in Chapter 3, is Υ/d. By Proposition 4 this implies that the mixing time of this random walk is $O(d^2\Upsilon^{-2}\log N)$. As a consequence, the probability distribution of the random walks in step 2 of the algorithm must be close to uniform. If p describes the probability distribution of the endpoint of a random walk starting from some fixed node, then the probability of a pairwise collision between two independent endpoints, i.e., the probability that the random walks end in the same node, is given by

$$\sum p(j)^2 = \|p\|^2.$$

This will be close to $1/N$ if p is close to uniform. If on the other hand the expansion of the graph is $\ll \Upsilon$, then random walks can get stuck in a small region, leading to an increase in the 2-norm or collision probability. It follows that the collision probability of a random walk forms a measure for the expansion of the graph. The key insight is then that, by a refinement of the birthday paradox, $\Theta(N^{1/2+\mu})$ independent samples of the same random walk suffice to estimate the collision probability to within a multiplicative factor $1 + O(N^{-2\mu})$. As a consequence, it is possible to estimate the 2-norm of a t-step random walk on an N-node graph using $\Theta(N^{1/2+\mu}t\epsilon^{-1})$ random walk steps.

The conjecture that Algorithm 4 is an expansion tester was later resolved as the conclusion of a series of papers by Czumaj and Sohler [125], Kale and Seshadhri [126] and Nachmias and Shapira [127], leading to the following theorem.

Theorem 7 ([127]). *If $d \geq 3$, then Algorithm 4 is a $(\Upsilon, \epsilon, \mu)$ expansion tester with runtime*

$$O(N^{1/2+\mu}\Upsilon^{-2}d^2\epsilon^{-1}\log N).$$

In the following section we show that we can use QFF to accelerate this tester very naturally by quantum simulating the random walks, and using quantum techniques to estimate the 2-norm.

9.2 Quantum Expansion Tester

It is possible to extend the classical query model to the quantum setting, a proper definition of which can be found in [123, 129]. For this work it suffices to know that (i) we can generate a uniformly random node as in the classical case, and (ii) we can implement a single step of the quantum walk operator as defined in 7 using $O(1)$ queries to the graph.

To accelerate the classical tester we will quantum simulate the random walks, and then perform a standard routine called *quantum amplitude estimation* to estimate the 2-norm of the simulated random walks. This routine is very similar to the aforementioned algorithm for quantum amplitude amplification, and is described in the same work by Brassard et al [122]. It is captured by the following lemma. We

note that in the original statement in [122] the number of reflections scales as $1/\delta$ for a success probability $1-\delta$. We use standard tricks to improve this to $\log(1/\delta)$.

Lemma 12 (Quantum Amplitude Estimation). *Consider a quantum state $|\psi\rangle$ and a general projector $\Pi_\flat$. Give some $\delta > 0$, there exists a quantum algorithm that outputs an estimate a such that*

$$\left| \|\Pi_\flat|\psi\rangle\| - a \right| \leq \epsilon$$

with probability at least $1-\delta$, using $O(\log(1/\delta)\epsilon^{-1})$ reflections around $|\psi\rangle$ and around the image of $\Pi_\flat$.

Proof. We can use the quantum amplitude estimation algorithm from [122, Theorem 12] to output an $\epsilon' = \epsilon/3$-close estimate with success probability at least $5/6$. This algorithm requires $O(1/\epsilon)$ reflections around $|\psi\rangle$ and around the image of $\Pi_\flat$. We can boost the success probability to $1-\delta$ by running their algorithm $T = \lceil 18 \ln \delta^{-1} \rceil$ times, which by Hoeffding's inequality implies that, with a probability at least $1-\delta$, at least $2T/3$ iterations have been successful. Therefore, with probability $1-\delta$, it holds that (i) at least $2T/3$ estimates lie in the interval $[\|\Pi_\flat|\psi\rangle\| - \epsilon', \|\Pi_\flat|\psi\rangle\| + \epsilon']$, and therefore (ii) we can find a subset $\mathcal{S}$ of estimates, $|\mathcal{S}| \geq 2T/3$, all of which lie in a $2\epsilon'$-interval. This subset must overlap with the interval $[\|\Pi_\flat|\psi\rangle\| - \epsilon', \|\Pi_\flat|\psi\rangle\| + \epsilon']$.

If now we output any element of this subset $\mathcal{S}$, we know that it lies in the interval $[\|\Pi_\flat|\psi\rangle\| - 3\epsilon', \|\Pi_\flat|\psi\rangle\| + 3\epsilon']$. This proves that with probability $1-\delta$ we can output an estimate of $\|\Pi_\flat|\psi\rangle\|$ with precision $3\epsilon' = \epsilon$, using T runs of the quantum amplitude estimation algorithm in [122], each of which requires $O(1/\epsilon)$ reflections around $|\psi\rangle$ and around the image of $\Pi_\flat$. This proves the claimed statement. □

We will use this amplitude estimation algorithm to estimate the 2-norm of a random walk. Thereto we recall the QFF scheme as discussed in Section 8.3. Note that the random walk (9.1) proposed in the GR tester is symmetric, so that we can simply replace the discriminant matrix D in the QFF algorithm by the random walk matrix P. Given a quantum state $|s, \flat\flat\rangle$, QFF applies an operator W_τ so that

$$\Pi_\flat W_\tau |s, \flat\flat\rangle = \left(\frac{1}{1-p_{>\tau}} \sum_{l=0}^{\tau} p_l T_l(P)|s\rangle \right) \otimes |\flat\flat\rangle \approx (P^t|s\rangle) \otimes |\flat\flat\rangle,$$

as in (8.4), where the summation corresponds to the truncated Chebyshev expansion of P^t. This requires $O(\tau)$ QW steps. If we set $\tau \in \Theta\big(\sqrt{t\log(N\epsilon^{-1})}\big)$ (replacing $\|P^t|s\rangle\|$ by its lower bound $N^{-1/2}$ in Algorithm 3) then the 2-norm of $\frac{1}{1-p_{>\tau}} \sum_{l=0}^{\tau} p_l T_l(P)|v\rangle$ approximates the 2-norm of $P^t|v\rangle$ to precision $O(\epsilon)$. Applying quantum amplitude estimation on the state $W_\tau|v, \flat\flat\rangle$ and projector $\Pi_\flat$ will therefore allow to estimate the 2-norm of $P^t|v\rangle$, as was our initial goal. This scheme is easily formalized in the below algorithm and theorem.

Algorithm 5 Quantum 2-norm Estimator

Input: parameters N and d; query access to N-node graph G with degree bound d; starting vertex s; running time t; accuracy parameter ϵ; confidence parameter δ

Do:

1: set $\tau \in O(\sqrt{t \log(N/\epsilon)})$
2: apply the QFF operator W_τ on the quantum state $|s, \natural\rangle$
3: use quantum amplitude estimation to output estimate of $\|\Pi_\natural W_\tau |s, \natural\rangle\|$ to error $\epsilon/2$ with probability $1 - \delta$

Theorem 8 (Quantum 2-norm Estimator). *With probability at least* $1 - \delta$*, Algorithm 5 outputs an estimate* a *such that*

$$\left| \|P^t|s\rangle\| - a \right| \leq \epsilon.$$

The algorithm requires a number of QW steps bounded by

$$O\left(\frac{\sqrt{t}}{\epsilon} \log \frac{1}{\delta} \log^{1/2} \frac{N}{\epsilon} \right).$$

Proof. We will prove the theorem for

$$\tau = \left\lceil \sqrt{2t \ln \left(8\sqrt{N}/\epsilon\right)} \right\rceil.$$

By this choice, and the fact that $\|P^t|s\rangle\| \geq N^{-1/2}$ on any N-node graph, we can deduce from the proof of Theorem 5 that

$$\left| \|\Pi_\natural W_\tau |s\rangle\| - \|P^t|s\rangle\| \right| \leq \epsilon/2.$$

Applying quantum amplitude estimation on $\Pi_\natural W_\tau |s\rangle$ with a precision $\epsilon/2$ therefore leads to an estimate of $\|P^t|s\rangle\|$ up to error ϵ. By Lemma 12 we can do so with a probability $1 - \delta$ using $O(\epsilon^{-1} \log \delta^{-1})$ reflections around $W_\tau |s\rangle$. By Lemma 11 in Section 8.4, we can implement a single such reflection using 2τ QW steps, and a reflection around the initial state (which is a basis state and can hence be neglected). □

We can compare this with the classical 2-norm tester proposed by Czumaj, Peng and Sohler [104, Lemma 3.2], building on the GR tester. For $\delta = 1/3$ their tester requires $O(t/\epsilon)$ queries to the graph, whereas our algorithm only requires $\widetilde{O}(\sqrt{t}/\epsilon)$ queries. We can now use our quantum 2-norm tester to create a quantum tester for the graph expansion. The proof makes use of some details from the classical proof of Nachmias and Shapira [127].

Algorithm 6 Quantum Graph Expansion Tester

Input: parameters N and d; query access to N-node graph G with degree bound d; expansion parameter Υ; accuracy parameter ϵ; running time parameter $\mu < 1/4$

Do:

1: set $t \in O(d^2\Upsilon^{-2}\log N)$, $M \in O(N^{-1/2})$, $\epsilon' \in O(N^{-1/2+\mu})$, $\delta \in O(\epsilon)$
2: **for** $T \in O(\epsilon^{-1})$ times **do**
3: select a uniformly random starting node s
4: use Algorithm 5 to create estimate a of $\|P^t|s\rangle\|$ to precision ϵ', with probability $1-\delta$
5: if $a > M + \epsilon'$, abort and output "`reject`"

Output: if no "`reject`", output "`accept`"

Theorem 9 (Quantum Graph Expansion Tester). *If $d \geq 3$ then Algorithm 6 is a $(\Upsilon, \epsilon, \mu)$ expansion tester. The algorithm has a runtime*

$$O(N^{1/2+\mu} d \Upsilon^{-1} \epsilon^{-1} \log(\epsilon^{-1}) \log N).$$

Proof. We will prove the theorem for $t = 16d^2\Upsilon^{-2}\log N$, $M = \sqrt{N^{-1}(1+N^{-1})}$, $\epsilon' = N^{-1/2+\mu}/(16\sqrt{2})$, $\delta = \epsilon/300$ and $T = 90/\epsilon$.

In each iteration the estimate a will be such that $\left|a - \|P^t|s\rangle\|\right| \leq \epsilon'$ with probability $1-\delta$, and hence

$$\left|a^2 - \|P^t|s\rangle\|^2\right| = \left|(a - \|P^t|s\rangle\|)(a + \|P^t|s\rangle\|)\right| \leq 2\epsilon'.$$

Nachmias and Shapira [127] showed that if G has a conductance $\geq \Upsilon$, then for all nodes s it holds that

$$\|P^t|s\rangle\| \leq M = \sqrt{N^{-1}(1+N^{-1})}.$$

Given such a graph, in each iteration the estimate $a \leq M + \epsilon'$ with probability $1-\delta$, so that the probability of a faulty rejection is at most δ per iteration. The total probability of a faulty rejection can then be bounded by $T\delta < 1/3$.

In the negative case, [127] showed that there exists a constant $c > 0$ such that if G is ϵ-far from any graph with max degree d and conductance $\geq c\mu\Upsilon^2$, then there exist at least $\epsilon N/128$ vertices s for which it holds that

$$\|P^t|s\rangle\| \geq \sqrt{N^{-1}(1+32^{-1}N^{-2\mu})}.$$

Given such a graph, in each iteration the estimate a will now be such that $a \geq \|P^t|s\rangle\| - \epsilon' \geq \sqrt{N^{-1}(1+32^{-1}N^{-2\mu})} - \epsilon'$ with probability $1-\delta$. To show that this quantity $> M + \epsilon'$, we bound $M = \sqrt{N^{-1}(1+N^{-1})} \leq N^{-1/2}(1+N^{-1/2})$ and $\sqrt{N^{-1}(1+32^{-1}N^{-2\mu})} \geq N^{-1/2}(1 - N^{-\mu}/(4\sqrt{2}))$, which shows that

$$\sqrt{N^{-1}(1+32^{-1}N^{-2\mu})} - M \geq N^{-1/2-\mu}/(4\sqrt{2}).$$

This proves that indeed

$$\sqrt{N^{-1}(1+32^{-1}N^{-2\mu})} - \epsilon' > M + \epsilon'$$

for $\epsilon' = N^{-1/2-\mu}/(16\sqrt{2})$. As a consequence, a single iteration will correctly output a rejection with probability $(1-\delta)\epsilon N/128$. The total probability of correctly

rejecting at least once is therefore lower bounded by $T(1-\delta)\epsilon/128 \geq 2/3$. This concludes the proof that Algorithm 6 is a $(\Upsilon, \epsilon, \mu)$ graph expansion tester. Its runtime is given by T times the runtime of the 2-norm tester, which by Theorem 8 is

$$O\left(\frac{\sqrt{t}}{\epsilon'}\log\frac{1}{\delta}\log^{1/2}\frac{N}{\epsilon'}\right) \in O\left(\left(d\Upsilon^{-1}\log^{1/2}N\right)N^{1/2+\mu}\log\frac{1}{\epsilon}\log^{1/2}N^{1+\mu}\right).$$

□

We recall that the classical GR tester has a runtime

$$O(N^{1/2+\mu}d^2\Upsilon^{-2}\epsilon^{-1}\log N).$$

Up to the $\log(\epsilon^{-1})$-factor we improve this runtime with a factor $d\Upsilon^{-1}$, which basically follows from the fact that we quadratically accelerate the random walk runtime to $t \in \widetilde{\Theta}(d\Upsilon^{-1})$. There exist graphs for which

$$\Upsilon \in \Omega(1/N),$$

so that in some cases we improve the runtime by a factor $d\Upsilon^{-1} \in \Theta(dN)$.

In summary, we have shown that QFF allows to speed up the GR expansion tester in a very natural way. We use it to quantum simulate and estimate the 2-norm of a t-step random walk in $\widetilde{O}(N^{1/2+\mu}\sqrt{t})$ queries to the graph, rather than $\widetilde{O}(N^{1/2+\mu}t)$ as is the runtime of the original tester. We can now compare this result to the preceding work by Ambainis, Childs and Liu [123]. They used a very different approach to speed up the GR expansion tester, using quantum walks only indirectly, which results in a runtime improvement of a different nature. In rough strokes they speed up the classical 2-norm tester by making use of Ambainis' quantum walk algorithm for element distinctness [4] to count collisions between pairs of classical random walks more efficiently. This allows them to improve the runtime of the 2-norm tester to $\widetilde{O}(N^{1/3+\mu}t)$, which provides a speedup complementary to the speedup of our 2-norm tester which in this context has a runtime $\widetilde{O}(N^{1/2+\mu}\sqrt{t})$. Using this 2-norm tester in the above Algorithm 6 leads to a runtime

$$\widetilde{O}(N^{1/3+\mu}d^2\Upsilon^{-2}\epsilon^{-1}).$$

Combined with our approach, this shows that there exists a quantum expansion tester with runtime $\widetilde{O}(\min(N^{1/3+\mu}d^2\Upsilon^{-2}, N^{1/2+\mu}d\Upsilon^{-1})\epsilon^{-1})$, simply by running either algorithm dependent on the relative size of N, Υ and d. We currently see no viable approach for simultaneously achieving the $\tilde{\Theta}(d\Upsilon^{-1})$ gain in random walk runtime of our algorithm, and the $\tilde{\Theta}(N^{1/6})$ gain in collision counting of the algorithm in [123].

We note that a property tester in the same spirit as the GR expansion tester was proposed by Batu et al [130] for testing the mixing time of general Markov chains on a graph. For the special case of reversible Markov chains we can straightforwardly apply the above ideas to their algorithm, yielding a similar speedup.

9.3 Clusterability and Robust *s*-*t* Connectivity

The GR technique of using collision counting to estimate the 2-norm of a distribution can also be used to estimate the inner product of any two given distributions p and q, defined by

$$\langle p, q\rangle = \sum_j p(j)q(j).$$

Indeed, this quantity is equal to the collision probability between the two distributions. This observation underlies the subsequent work by Batu et al [130], Valiant [124] and Chan et al [131] for estimating the 1-norm and 2-norm distance between a given pair of distributions. Below we will present a quantum algorithm based on QFF to estimate the 2-norm between random walk distributions, accelerating these existing classical algorithms. The latter formed the inspiration for the more recent line of *graph clusterability testers* by Czumaj et al [104] and Chiplunkar et al [105]. Here the goal is to test whether a graph can be appropriately clustered into k parts for some given k. The GR expansion tester can be seen as a graph clusterability tester for $k = 2$. As a subroutine they use an algorithm to distinguish whether a pair of nodes lie in the same cluster or not, leading to a robust notion of *s*-*t* connectivity. We will show that our quantum algorithm for estimating the 2-distance leads to a speedup on this subroutine, which is of independent interest outside of the setting of property testing, and in extent leads to a speedup on their clusterability tester.

2-distance Estimator

To estimate the 2-distance of a pair of random walks, we will combine QFF with the *SWAP test*, which is a standard tool in quantum algorithms. The test assumes that we are given two quantum states $|\psi\rangle$ and $|\phi\rangle$ and an ancillary qubit in the state $|0\rangle$, yielding the state $|0\rangle|\psi\rangle|\phi\rangle$. On this state we apply the following operations:

$$\begin{aligned}|0\rangle|\psi\rangle|\phi\rangle &\overset{H\otimes I\otimes I}{\to} \tfrac{|0\rangle+|1\rangle}{\sqrt{2}}|\psi\rangle|\phi\rangle \\ &\overset{cS}{\to} \tfrac{1}{\sqrt{2}}|0\rangle|\psi\rangle|\phi\rangle + \tfrac{1}{\sqrt{2}}|1\rangle|\phi\rangle|\psi\rangle \\ &\overset{H\otimes I\otimes I}{\to} \tfrac{1}{2}|0\rangle(|\psi\rangle|\phi\rangle + |\phi\rangle|\psi\rangle) + \tfrac{1}{2}|1\rangle(|\psi\rangle|\phi\rangle - |\phi\rangle|\psi\rangle),\end{aligned}$$

where we used the Hadamard gate

$$H = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix},$$

and the *conditional swap operation* cS swapping the second and third registers conditional on the first register being in the state $|1\rangle$. We will call the combined unitary operation $U_{\mathrm{SWAP}} = (H \otimes I \otimes I)cS(H \otimes I \otimes I)$. We can now either measure the first register, or apply quantum amplitude estimation to the projector $\Pi_1 = |1\rangle\langle 1| \otimes I \otimes I$, to estimate the quantity

$$\||\psi\rangle|\phi\rangle - |\phi\rangle|\psi\rangle\|^2 = 2(1 - |\langle\psi|\phi\rangle|^2).$$

This quantity will be small if $|\psi\rangle$ and $|\phi\rangle$ are close and large otherwise, allowing to estimate the distance between the input states $|\psi\rangle$ and $|\phi'\rangle$. We can combine

this test with our QFF algorithm, and the 2-norm estimator in previous section, to obtain a tester for the 2-distance, the description of which we postpone to Section 9.4.

Theorem 10 (Quantum 2-distance Estimator). *With probability at least* $1-\delta$*, Algorithm 7 outputs an estimate* a *such that*

$$\left| \|P^t|u\rangle - P^t|v\rangle\|^2 - a \right| \leq \epsilon.$$

For $\overline{a} = \max\{\|P^t|u\rangle\|, \|P^t|v\rangle\|\}$ *and* $\underline{a} = \min\{\|P^t|u\rangle\|, \|P^t|v\rangle\|\}$*, the algorithm requires an expected number of QW steps bounded by*

$$O\left(\sqrt{t}\left(\frac{\overline{a}}{\epsilon} + \frac{\overline{a}^4}{\underline{a}\epsilon^2}\right)\log\frac{\log N}{\delta}\log^{3/2}\frac{N}{\epsilon}\right).$$

For comparison, the classical estimator presented in Czumaj et al [104, Theorem 3.1] requires a number of graph queries or random walk steps $O\big(t\frac{\overline{a}}{\epsilon}\log\frac{1}{\delta}\big)$. Chan et al [131] give an information theoretical proof that classically $\Omega(\overline{a}/\epsilon)$ samples are needed to estimate the 2-distance between a pair of distributions.

Classifying Nodes

Czumaj et al [104] use their classical 2-distance estimator to propose a property tester for the clusterability of a graph. While ambiguous in nature, the notion of clusterability used in that and other lines of work follows the intuitive ideas presented for instance by Oveis Gharan and Trevisan [132]. A partition $\mathcal{V} = \mathcal{S}_1 \cup \cdots \cup \mathcal{S}_h$ of the nodes into disjoint clusters is quantified as a good clustering if the clusters are well-connected internally as well as poorly connected externally. For a single cluster $\mathcal{S}_j$ the internal connectivity is quantified by the *inner conductance* of the induced subgraph $G[\mathcal{S}_j]$, where $G[\mathcal{S}_j]$ is defined as the graph consisting of the nodes in $\mathcal{S}_j$ and the edges between these nodes. Following our former definition (3.3) in Chapter 3, we define the inner conductance $\Phi(G[\mathcal{S}_j])$ of a subset $\mathcal{S}_j$ as the conductance of the induced subgraph $G[\mathcal{S}_j]$ with respect to the symmetric random walk defined in (9.1). For a graph with degree bound d this becomes

$$\Phi(G[\mathcal{S}_j]) = \min_{\mathcal{T}\subset\mathcal{S}_j, |\mathcal{T}|\leq|\mathcal{S}_j|/2} \frac{|E(\mathcal{T}, \mathcal{S}_j\backslash\mathcal{T})|}{d|\mathcal{T}|}.$$

Again in accordance with Chapter 3, we define the *outer conductance* of a single cluster $\mathcal{S}_j$ as

$$\Phi(\mathcal{S}_j) = \frac{|E(\mathcal{S}_j, \mathcal{S}_j^c)|}{d|\mathcal{S}_j|},$$

Given these definitions, a graph is called $(k, \Phi_{\text{in}}, \Phi_{\text{out}})$-*clusterable* if there exists a partition $\mathcal{V} = \mathcal{S}_1 \cup \cdots \cup \mathcal{S}_h$, $h \leq k$, such that the clusters are well-connected internally, $\phi(G[\mathcal{S}_j]) \geq \Phi_{\text{in}}$, and the clusters are poorly-connected externally, $\phi(\mathcal{S}_j) \leq \Phi_{\text{out}}$. It turns out that graph clusterability can be detected when the gap between Φ_{in} and Φ_{out} is sufficiently large - typically quadratic, $\Phi_{\text{out}} \in \widetilde{O}(\Phi_{\text{in}}^2)$.

To test the graph clusterability, Czumaj et al [104] propose a subroutine for *classifying nodes*, i.e., determining whether two nodes lie in the same cluster or not. This routine is of independent interest, outside of setting of graph property testing. Indeed, when given for instance a similarity graph of a set of objects, it can serve as a way to classify an object between a number of exemplary objects. It turns out that it is possible to classify nodes quite intuitively by comparing random walks starting from the nodes: the 2-distance between random walks starting from nodes of the same cluster will typically be smaller than the 2-distance between nodes from different clusters. This is formalized in the following lemma, which we extract from Czumaj et al [104, Lemma 4.1 and 4.3]. Given an appriopriately clusterable graph, having a gap $\Phi_{\text{out}} \in \widetilde{O}(\Phi_{\text{in}}^2)$, it gives bounds on the 2-distance between pairs of nodes coming from the same or different clusters. Following the work of Spielman and Teng [133] on local graph clustering, the lemma is confined to the *internal nodes* $\tilde{\mathcal{S}} \subseteq \mathcal{S}$ of a cluster $\mathcal{S}$. These nodes are characterized by the fact that a t-step random walk, starting from an internal node of a cluster $\mathcal{S}$, will have a probability bounded by $t\Phi(\mathcal{S})$ to exit $\mathcal{S}$. One can show that $|\tilde{\mathcal{S}}| \geq |\mathcal{S}|/2$, so that at least half of the nodes inside a cluster are internal nodes.

Lemma 13 ([104]). *Consider a $(k, \Phi_{\text{in}}, \Phi_{\text{out}})$-clusterable graph with degree bound d, and let $\mathcal{S}$ and $\mathcal{S}'$ be clusters of such a partition. Assume that*

$$\Phi_{\text{out}} \leq c\Phi_{\text{in}}^2/\log N,$$

with c some constant dependent on d, k, $|\mathcal{S}|/N$ and $|\mathcal{S}'|/N$. Then there exist subsets $\tilde{\mathcal{S}} \subseteq \mathcal{S}$, $|\tilde{\mathcal{S}}| \geq |\mathcal{S}|/2$, and $\tilde{\mathcal{S}}' \subseteq \mathcal{S}'$, $|\tilde{\mathcal{S}}'| \geq |\mathcal{S}'|/2$, and a universal constant c', such that for $t = \lceil c' k^4 \Phi_{\text{in}}^{-2} \log N \rceil$ it holds that

- *if two nodes $u, v \in \tilde{\mathcal{S}}$ or $u, v \in \tilde{\mathcal{S}}'$, then $\|P^t|u\rangle - P^t|v\rangle\|^2 \leq 1/(4N)$.*
- *if two nodes $u \in \tilde{\mathcal{S}}$ and $v \in \tilde{\mathcal{S}}'$, then $\|P^t|u\rangle - P^t|v\rangle\|^2 \geq 1/N$.*

Under the additional assumption that the size of each of the clusters is of the same order, $|\mathcal{S}_j| \in \Theta(N/k)$, we can show that the lower bound in the second bullet can be improved to $\Omega(1/|\mathcal{S}|) \in \Omega(k/N)$, thus scaling with the size of the local cluster. This is relevant when working with *massive graphs*, as considered for instance by Spielman and Teng [134], where the scaling with the total graph size should be at most logarithmic.

The lemma shows that under appropriate conditions we can distinguish whether two nodes u and v are internal nodes of the same cluster or not, simply by estimating the 2-distance between random walks from u and v. This leads to the below proposition. It can be seen as a robust version of s-t connectivity, more relevant to e.g. social networks, where mere connectivity between two nodes is no longer deemed an interesting quantity. Alternatively, in a learning context, it allows to perform nearest-neighbor classification of nodes on a graph. The idea of the proof is again quite clear. Under the conditions of the above lemma, we can distinguish both cases by estimating $\|P^t|u\rangle - P^t|v\rangle\|^2$ to error $\epsilon = 1/N$. Following Theorem 10, we can efficiently do so provided that $\overline{a} = \max\{\|P^t|u\rangle\|, \|P^t|v\rangle\|\}$ is not too large. Czumaj et al [104, Lemma 4.2] prove that, under the conditions of the above lemma, for at least $9/10$-th of the nodes u it holds that $\|P^t|u\rangle\| \in O(k/N)$. This allows us to prove the following proposition, showing that a large fraction of the

nodes of an appropriately clustered graph can be efficiently classified: given two such nodes, we can determine whether they lie in the same cluster or not.

Proposition 18 (Classifying Nodes).

- *Under the conditions of Lemma 13, we can use the quantum 2-distance estimator to determine with probability at least* $2/3$ *whether two internal nodes lie in the same cluster or not.*
- *There exists a subset* $\tilde{\mathcal{V}} \subseteq \mathcal{V}$, $|\tilde{\mathcal{V}}| \geq 9|\mathcal{V}|/10$, *such that if in addition both nodes lie in* $\tilde{\mathcal{V}}$, *then the algorithm requires* $O(N^{1/2}k^4\Phi_{\text{in}}^{-1}\log^{3/2}N)$ *expected QW steps.*

Proof. To prove the first bullet, it suffices to use Lemma 13 which states that if both lie in the same cluster, then $\|P^t|u\rangle - P^t|v\rangle\|^2 \leq 1/(4N)$, whereas if both lie in different clusters, then $\|P^t|u\rangle - P^t|v\rangle\|^2 \geq 1/N$. By Theorem 10 we can estimate $\|P^t|u\rangle - P^t|v\rangle\|^2$ to error $\epsilon = 1/N$, which allows to distinguish both cases.

To prove the second bullet, let $\tilde{\mathcal{V}}$ denote a set of nodes u for which $\|P^t|u\rangle\| \in O(k/N)$, which by [104, Lemma 4.2] we know we can choose of size at least $9|\mathcal{V}|/10$. If both nodes lie in $\tilde{\mathcal{V}}$, then in Theorem 10 we can set $\overline{a} \in O(k/N)$, and $\underline{a} \in O(1/N)$ since necessarily $\|P^t|u\rangle\| \geq 1/N$ for any node u. In this case, the expected number of QW steps becomes $O(\sqrt{tN}\log^{3/2}N)$. For t as in Lemma 13, this proves the second bullet. □

We can compare the runtime in the second bullet by the runtime when using classical collision counting, which requires a number of RW steps $\widetilde{O}(N^{1/2}k^4\Phi_{\text{in}}^{-2})$. Applying the element distinctness technique by Ambainis et al [123] requires a number of QW steps $\widetilde{O}(N^{1/3}k^4\Phi_{\text{in}}^{-2})$.

Lemma 13, and a classifier as in Proposition 18, underlies the graph clusterability tester proposed by [104]. Since the tester is in the same vein as the GR expansion tester, we will not state it explicitly but merely summarize the idea. The algorithm selects a uniformly random set of $\Theta(k\log k)$ nodes over which it constructs a *similarity graph* by adding an edge between any pair of nodes if their random walk probabilities are closer than some threshold. This similarity graph serves as a *graph sketch*, reminiscent of the recent surge of results on graph sketching and sparsification [135]. They then prove that if the graph is appropriately clusterable in at most k components, then with high probability this small similarity graph will have at most k connected components, which they then check by brute force. Using the classical 2-distance estimator to estimate the distance between random walk distributions, this leads to a clusterability tester requiring $\widetilde{O}\big(N^{1/2}k^7\Phi_{\text{in}}^{-2}\epsilon^{-5}\big)$ RW steps. We can improve this to $\widetilde{O}\big(N^{1/2}k^7\Phi_{\text{in}}^{-1}\epsilon^{-4}\big)$ QW steps using Proposition 18. It seems feasible that using the element distinctness technique in [136] an alternative speedup to $\widetilde{O}\big(N^{1/3}k^7\Phi_{\text{in}}^{-2}\epsilon^{-5}\big)$ RW steps can be achieved.

It seems appropriate to compare these results to the alternative yet related and extensive line of algorithms for *local graph clustering* [133, 134, 137–139] These algorithms aim to explicitly retrieve a local cluster by accessing it through graph queries, which could evidently be used to classify nodes or test the graph clusterability. Given an internal node of a cluster $\mathcal{S}$ for which $\Phi_{\text{out}} \in O(\Phi_{\text{in}}^2/\log|\mathcal{S}|)$, Spielman and Teng [133, 134] and Andersen et al [137] explicitly output an approximation of $\mathcal{S}$ by explicitly computing a random walk and a PageRank vector over $\mathcal{S}$,

respectively. The algorithm in [137] requires $\widetilde{O}(|\mathcal{S}|\Phi_{\text{in}}^{-2})$ steps and graph queries. Andersen and Peres [138] were able to improve this runtime to $\widetilde{O}(|\mathcal{S}|\Phi_{\text{in}}^{-1})$ by using a sophisticated *evolving-set process*, which corresponds to a Markov chain over subsets of the nodes, rather than over single nodes. The speedup that they achieve is the same in nature as our speedup, which basically amounts to speeding up the random walk behavior. The linear dependence on $|\mathcal{S}|$ is however inevitable in their approach, since it requires to effectively keep track of subsets of size $\Theta(|\mathcal{S}|)$. In a graph consisting of k clusters, at least one cluster has size N/k so that this implies a runtime $O(Nk^{-1}\Phi_{\text{in}}^{-1})$, as compared to the runtime $\widetilde{O}(N^{1/2}k^4\Phi_{\text{in}}^{-1})$ that we achieve.

As a finishing note to this chapter, we remark that QFF allows for a very natural quantization and speedup of random walk algorithms for property testing. The speedup is complementary to a quantum speedup achieved by Ambainis et al [123], yet makes a more direct use of quantum walks as a way of simulating and speeding up random walks. This properly illustrates the benefits of our new QFF technique, which allows to effectively simulate the dynamics of random walks on any given timescale, as is crucial for the demonstrated applications. We can contrast this to the many aforementioned existing quantum walk algorithms which rely on a speedup of the random walk limit behavior, which would not suffice for testing e.g. the cluster structure of a graph.

9.4 Quantum 2-distance Estimator: Algorithm and Proof

In this extra section we present the algorithm and proof underlying Theorem 10, which concerns the estimation of the distance between two random walk distributions $p = P^t|u\rangle$ and $q = P^t|v\rangle$. To construct our algorithm, we rewrite

$$\|p - q\|^2 = \|p\|^2 + \|q\|^2 - 2\|p\|\|q\|\langle p|q\rangle,$$

using the notation $\langle p|q\rangle = \langle p, q\rangle/(\|p\|\|q\|)$. As a consequence, we can retrieve an estimate by separately estimating $\|p\|$, $\|q\|$ and $\langle p|q\rangle$. Towards estimating $\|p\|$ and $\|q\|$, we present at the end of this section a simple extension of the quantum 2-norm tester presented earlier in this chapter that allows to estimate the 2-norm up to multiplicative error, instead of additive error. Towards estimating $\langle p|q\rangle$, we first create approximations of $|p\rangle = p/\|p\|$ and $|q\rangle = q/\|q\|$, on which we subsequently apply the SWAP test and amplitude estimation. A subtlety is that we cannot simply use our QFF algorithm to create $|p\rangle$ and $|q\rangle$ with high probability. Indeed, in order to apply amplitude estimation we must reflect around these states, and it is not clear that we can reflect around the output of the QFF algorithm. Instead, we will apply the unitary amplitude amplification operator to the states $W_\tau|u, \text{bb}\rangle$ and $W_\tau|v, \text{bb}\rangle$ to unitarily rotate these states close to $|p\rangle$ and $|q\rangle$, omitting the final measurement in Algorithm 3. This invertible operation will allow to reflect around the output states, using a similar argument as in Lemma 11. Instead of the amplitude amplification operator introduced in Section 8.4, we will make use of an enhanced operator by Yoder and Low [140]. This operator, as described in the below lemma, is better suited for the case where we only have a lower bound on the success probability.

Lemma 14 (Fixed Point Amplitude Amplification [140]). *Consider a state* $|\psi\rangle$ *and a projector* $\Pi_\flat$ *such that* $\|\Pi_\flat|\psi\rangle\| = \lambda > 0$*. For any constant* $\delta > 0$*, there exists a family of unitary transformations* U_L *such that if* $L \geq \lambda^{-1}\log(2/\delta)$ *then*

$$|\langle\psi_\flat|U_L|\psi\rangle|^2 \geq 1 - \delta^2,$$

where $\psi_\flat = \Pi_\flat|\psi\rangle/\|\Pi_\flat|\psi\rangle\|$*. We can implement* U_L *using* $O(L)$ *reflections around* $|\psi\rangle$ *and around the image of* $\Pi_\flat$*.*

Using the appropriate operator U_L, we can therefore retrieve approximations $|\psi_u\rangle = U_L W_\tau|u, \flat\flat\rangle \approx |p\rangle$ and $|\psi_v\rangle = U_L W_\tau|v, \flat\flat\rangle \approx |q\rangle$. We can now apply the SWAP test to these states, combined with amplitude amplification, to retrieve an estimation of $\langle p|q\rangle$. To see this, note that

$$\Pi_1(U_{\text{SWAP}}|0\rangle|\psi_u\rangle|\psi_v\rangle) = \frac{1}{2}|1\rangle(|\psi_u\rangle|\psi_v\rangle - |\psi_v\rangle|\psi_u\rangle).$$

As a consequence we can apply quantum amplitude estimation on the state $U_{\text{SWAP}}|0\rangle|\psi_u\rangle|\psi_v\rangle$ with respect to the projector Π_1 to estimate the quantity

$$\frac{1}{2}\big\||\psi_u\rangle|\psi_v\rangle - |\psi_v\rangle|\psi_u\rangle\big\|^2 = 1 - |\langle\psi_u|\psi_v\rangle|^2 \approx 1 - |\langle p|q\rangle|^2.$$

Combined with the former estimates of $\|p\|$ and $\|q\|$ this leads to an estimate of the 2-distance we were looking for. We formalize this in the following algorithm and theorem.

Algorithm 7 Quantum 2-distance Estimator

Input: parameters N and d; query access to N-node graph G with degree bound d; starting vertices u and v; running time t; accuracy parameter ϵ; confidence parameter δ

Do:

1: use Algorithm 8 to create estimates α and β of $\|P^t|u\rangle\|$ resp. $\|P^t|v\rangle\|$ to multiplicative error $1/4$, with probability $1 - \delta/4$

2: set $\mu \in O(\epsilon \max(\alpha, \beta)^{-2})$

3: use Algorithm 8 to create new estimates α and β of $\|P^t|u\rangle\|$ resp. $\|P^t|v\rangle\|$ to multiplicative error μ, with probability $1 - \delta/4$

4: set $L \in \Omega(\min(\alpha, \beta)^{-1}\log\min(\alpha, \beta)^{-1})$ and $\tau \in \Omega(\sqrt{t\ln(N/\mu)})$

5: apply W_τ, U_L and U_{SWAP} to create the state

$$|\psi\rangle = U_{\text{SWAP}}|0\rangle\big(U_L W_\tau|u, \flat\flat\rangle\big)\big(U_L W_\tau|v, \flat\flat\rangle\big)$$

6: use amplitude estimation to create an estimate γ of $\|\Pi_1|\psi\rangle\|$ to error μ, with probability $1 - \delta/2$

Output: estimate $a = \alpha^2 + \beta^2 - 2\alpha\beta\sqrt{1 - \gamma^2/2}$

Theorem 11 (Quantum 2-distance Estimator). *With probability at least* $1 - \delta$*, Algorithm 7 outputs an estimate* a *such that*

$$\big|\|P^t|u\rangle - P^t|v\rangle\|^2 - a\big| \leq \epsilon.$$

With $\overline{a} = \max\{\|P^t|u\rangle\|, \|P^t|v\rangle\|\}$ *and* $\underline{a} = \min\{\|P^t|u\rangle\|, \|P^t|v\rangle\|\}$*, the algorithm requires an expected number of QW steps bounded by*

$$O\left(\sqrt{t}\left(\frac{\overline{a}}{\epsilon} + \frac{\overline{a}^4}{\underline{a}\epsilon^2}\right)\log\frac{\log N}{\delta}\log^{3/2}\frac{N}{\epsilon}\right).$$

Proof. We prove the theorem for

$$\mu = \frac{1}{26}\min\left(1, \frac{9\epsilon}{16\max(\alpha,\beta)^2}\right), \quad L = \left\lceil\frac{1}{\lambda}\log\frac{2}{\nu}\right\rceil, \quad \tau = \left\lceil\sqrt{2t}\ln^{1/2}\frac{4}{\lambda\nu}\right\rceil,$$

with $\lambda = \min(\alpha,\beta)/(1+\nu)$ and $\nu = \mu^2/11$. We will denote $p = P^t|u\rangle$, $q = P^t|v\rangle$, $|p\rangle = p/\|p\|$, $|q\rangle = q/\|q\|$, $\overline{a}^2 = \max(\|p\|, \|q\|)$ and $\underline{a} = \min(\|p\|, \|q\|)$. The algorithm estimates the quantity $\|p-q\|^2 = \|p\|^2 + \|q\|^2 - 2\|p\|\|q\|\langle p|q\rangle$ by separately estimating $\|p\|$, $\|q\|$ and $\langle p|q\rangle$ to error $O(\epsilon/\overline{a}^2)$.

After the first step, we retrieve with probability at least $1-\delta/4$ estimates α and β such that

$$\frac{3}{4}\|p\| \le \alpha \le \frac{5}{4}\|p\|, \qquad \frac{3}{4}\|q\| \le \beta \le \frac{5}{4}\|q\|.$$

This proves that the parameter

$$\mu = \frac{1}{26}\min\left(1, \frac{\epsilon}{(4\max(\alpha,\beta)/3)^2}\right) \le \frac{1}{26}\min\left(1, \frac{\epsilon}{\overline{a}^2}\right), \tag{9.2}$$

and $\mu \in \Theta(\min(1, \epsilon/\overline{a}^2))$. In step 3 we then create new estimates of $\|p\|$ and $\|q\|$ to multiplicative error μ. The combined success probability of both steps is $(1-\delta/4)^2 \ge 1-\delta/2$. Following Theorem 12 these steps require an expected number of QW steps in

$$O\left(\frac{\sqrt{t}\overline{a}}{\epsilon}\log\frac{\log N}{\delta}\log^{1/2}\frac{N}{\epsilon}\right).$$

In the following steps of the algorithm we estimate $\langle p|q\rangle = \frac{\langle p,q\rangle}{\|p\|\|q\|}$ to additive error μ by combining QFF, amplitude amplification, the SWAP test and amplitude estimation. Thereto we first rewrite

$$\langle p|q\rangle = \sqrt{1 - \frac{\||p\rangle|q\rangle - |q\rangle|p\rangle\|^2}{2}},$$

showing that we can use an estimate on $\||p\rangle|q\rangle - |q\rangle|p\rangle\|$ to estimate $\langle p|q\rangle$. Indeed, it is easily seen from a function plot that if we create an estimate $\kappa \in [0, \sqrt{2}]$ such that $\left|\||p\rangle|q\rangle - |q\rangle|p\rangle\| - \kappa\right| \le \mu^2$, then the estimate $\sqrt{1-\kappa^2/2}$ will be μ-close:

$$\left|\sqrt{1-\kappa^2/2} - \langle p|q\rangle\right| \le \mu. \tag{9.3}$$

We now create an estimate of $\||p\rangle|q\rangle - |q\rangle|p\rangle\|$. By Lemma 14 and Theorem 6, and our choice of L and τ, it holds that

$$\begin{aligned}\|U_L W_\tau|u,\flat\flat\rangle - |p,\flat\flat\rangle\| &\le (1-\nu^2)\|W_\tau|u,\flat\flat\rangle/\|W_\tau|u,\flat\flat\rangle\| - |p,\flat\flat\rangle\| + \nu \\ &\le (1-\nu^2)\nu + \nu \le 2\nu,\end{aligned}$$

with $\nu = \mu^2/11$, and similarly for $U_L W_\tau |v, \flat\flat\rangle$. If we set $|\psi_u\rangle = U_L W_\tau |u, \flat\flat\rangle$ and $|\psi_v\rangle = U_L W_\tau |v, \flat\flat\rangle$, then this implies that

$$\big| \| |\psi_u\rangle|\psi_v\rangle - |\psi_v\rangle|\psi_u\rangle \| - \| |p\rangle|q\rangle - |q\rangle|p\rangle \| \big| \leq 8\nu(1+2\nu).$$

Now we can apply amplitude estimation, as in Lemma 12, to the state $U_{\mathrm{SWAP}}|0\rangle|\psi_u\rangle|\psi_v\rangle$ and projector Π_1 with success probability $1-\delta/2$ and error ν. If successful this returns an estimate γ of $\| |\psi_u\rangle|\psi_v\rangle - |\psi_v\rangle|\psi_u\rangle \|$ to error ν. Combined with the above inequality this shows that

$$\big| \| |p\rangle|q\rangle - |q\rangle|p\rangle \| - \gamma \big| \leq \nu + 8\nu(1+2\nu) \leq \mu^2.$$

By (9.3) this leads to the promised bound $\big|\sqrt{1-\gamma^2/2} - \langle p|q\rangle\big| \leq \mu$.

Implementing W_τ, U_L and U_{SWAP} requires a number of QW steps $O(\tau) + O(L)$, bounded by

$$O\left(\frac{\sqrt{t}}{\underline{a}} \log \frac{\overline{a}}{\epsilon \underline{a}} \log^{1/2} \frac{N\overline{a}}{\epsilon} \right).$$

Applying amplitude estimation with success probability $1-\delta/2$ and error $\nu \in \Theta(\epsilon^2/\overline{a}^4)$ requires $O\big(\frac{\overline{a}^4}{\epsilon^2} \log \frac{1}{\delta}\big)$ reflections around the state $U_{\mathrm{SWAP}}|0\rangle|\psi_u\rangle|\psi_v\rangle$. Following the argument in Lemma 11, we can implement each such reflection using the same number of QW steps required to implement the operators W_τ, U_L and U_{SWAP}. This leads to a total number of QW steps bounded by

$$O\left(\frac{\sqrt{t}\overline{a}^4}{\underline{a}\epsilon^2} \log \frac{1}{\delta} \log \frac{\overline{a}}{\epsilon \underline{a}} \log^{1/2} \frac{N\overline{a}}{\epsilon} \right).$$

Combined with the first approximation part, we find estimates α, β and γ such that $|\alpha - \|p\|| \leq \mu\|p\|$, $|\beta - \|q\|| \leq \mu\|q\|$ and $|\gamma - \langle p|q\rangle| \leq \mu$. This allows to prove the claimed error of the estimate

$$\begin{aligned} \big|\alpha^2 + \beta^2 - 2\alpha\beta\gamma - \|p-q\|^2\big| &\leq \mu(2+\mu)(\|p\|^2 + \|q\|^2) \\ &\quad + 2\|p\|\|q\|\big[\mu(2+\mu)(\langle p|q\rangle + \mu) + (1+\mu)^2\mu\big] \\ &\leq 3\mu(\|p\|^2 + \|q\|^2) + 20\mu\|p\|\|q\| \\ &\leq 26\mu \max(\|p\|, \|q\|)^2 \leq \epsilon, \end{aligned}$$

using the bound (9.2). The total success probability can be bounded by $(1-\delta/2)^2 \geq 1-\delta$, and the expected number of QW steps by

$$O\left(\sqrt{t}\left(\frac{\overline{a}}{\epsilon} + \frac{\overline{a}^4}{\underline{a}\epsilon^2} \right) \log \frac{\log N}{\delta} \log^{3/2} \frac{N}{\epsilon} \right).$$

□

2-norm Estimator to Multiplicative Error

In the above estimator for the 2-distance we wish to estimate $\|P^t|u\rangle\|$ to some multiplicative error ϵ, without having a bound on $\|P^t|u\rangle\|$. We present such an estimator in the below algorithm and theorem.

Algorithm 8 Quantum Multiplicative 2-norm Estimator

Input: parameters N and d; query access to N-node graph G with degree bound d; starting vertex u; running time t; accuracy parameter ϵ; confidence parameter δ

Do:

1: **for** $k = 1 \dots T \in O(\log N)$ **do**
2: use Algorithm 5 to create estimate α of $\|P^t|u\rangle\|$ to error $\epsilon_k = \epsilon 2^{-k-2}$, with probability $1 - \delta'$ for $\delta' \in O(\delta \log^{-1} N)$
3: if $\alpha \geq (1+\epsilon)2^{-k}$, abort **for**-loop
4: output α

Theorem 12 (Quantum Multiplicative 2-norm Estimator). *With probability at least $1 - \delta$, Algorithm 7 outputs an estimate α such that*

$$\left| \|P^t|u\rangle\| - \alpha \right| \leq \epsilon \|P^t|u\rangle\|.$$

The algorithm requires an expected number of QW steps bounded by

$$O\left(\frac{\sqrt{t}}{\epsilon \|p\|} \log \frac{\log N}{\delta} \log^{1/2} \frac{N}{\epsilon} \right).$$

Proof. We will prove the theorem for $T = \lceil \frac{1}{2} \log N \rceil$ and $\delta' = \delta/T$. We do so by showing that with probability at least $1 - \delta$ the loop aborts such that the value of α forms an estimate of $\|p\|$ to multiplicative error ϵ, where we denote $p = P^t|u\rangle$. We first assume that every call to Algorithm 5 is successful, the probability of which we will bound afterwards. Let a_k be the value of α in the k-th iteration, so that $|\|p\| - a_k| \leq \epsilon_k$. If the loop is stopped at the k-th iteration then $a_k \geq (1+\epsilon)2^{-k}$ or equivalently $\epsilon_k \leq \frac{\epsilon}{1+\epsilon} a_k$. Combined with the fact that $a_k \leq \|p\| + \epsilon_k$ this shows that $\epsilon_k \leq \frac{\epsilon}{1+\epsilon}(\|p\| + \epsilon_k)$ or equivalently $\epsilon_k \leq \epsilon \|p\|$, so that we find an estimate with multiplicative error ϵ.

If the first $\lceil \log \|p\|^{-1} \rceil$ calls to the 2-norm estimator are successful, then the algorithm stops and outputs a correct estimate. We can bound this number of calls by $T = \lceil \frac{1}{2} \log N \rceil$ using the fact that $\|p\| \geq N^{-1/2}$. The probability that this happens, i.e., that none of the first $\lceil \log \|p\|^{-1} \rceil$ implementations of the 2-norm tester fails, is at least $1 - \lceil \log \|p\|^{-1} \rceil \delta' \geq 1 - \delta$ if we set $\delta' = \delta/T$. This proves the success probability of the algorithm.

To bound the runtime, we first note that the k-th iteration runs the 2-norm tester with error $\epsilon_k = \epsilon 2^{-k}$ and success probability $1 - \delta'$, which by Theorem 8 requires a number of QW steps

$$O\left(\frac{2^k \sqrt{t}}{\epsilon} \log \frac{\log N}{\delta} \log^{1/2} \frac{2^k N}{\epsilon} \right).$$

Now we bound the expected number of iterations. If the algorithm succeeds, then it aborts after $\lceil \log \|p\|^{-1} \rceil$ iterations. If this does not happen, then either it aborts earlier, resulting in a number of iterations smaller than $\lceil \log \|p\|^{-1} \rceil$, or it aborts later. However, after $\lceil \log \|p\|^{-1} \rceil$ iterations, any successful call to the 2-norm tester will abort the algorithm, which happens per iteration with probability at least $1 - \delta$. In such case the expected number of iterations can be bounded by $(1 - \delta)^{-1} \leq 2$

under the assumption that $\delta \leq 1/2$. In any case we see that the expected number of iterations is $O(\log \|p\|^{-1})$. Now we can use the fact that $\sum_{k=0}^{b} 2^k \log^{1/2} 2^k \in O(2^b \sqrt{b}) \in O(\|p\|^{-1} \log^{1/2} \|p\|)$ for $b \in O(\log \|p\|^{-1})$ to bound the total expected number of QW steps by

$$O\left(\frac{\sqrt{t}}{\epsilon \|p\|} \log \frac{\log N}{\delta} \log^{1/2} \frac{N}{\epsilon}\right). \qquad \square$$

Chapter 10
Other Applications and Outlook

In this chapter we discuss some further applications of QFF. First we consider the problem of speeding up the *escape time* of a Markov chain. For a Markov chain that starts inside a subset of a graph, this quantifies how long it take the Markov chain to escape this set. We will show that QFF allows to escape large sets quadratically faster. This is a new result that relates to the long line of graph search algorithms using quantum walks [3, 4, 21, 22], underlying much of the early work on quantum walk algorithms. A big open question in this line of work is accelerating the hitting time of large sets. Our result sheds new light on this issue.

Secondly we show that QFF can be a relevant tool in the field of quantum state generation [40]. Here the goal is to construct a quantum state that is typically specified only implicitly, similar to the classical Markov chain Monte Carlo setting. In the final section we provide some finalizing discussion on the optimality of the QFF runtime and on its generalization from Markov chains to Hamiltonian matrices.

10.1 Escaping Sets Quadratically Faster

We first show that QFF allows to speed up the problem of escaping large sets. Formally, consider a Markov chain P and a subset $\mathfrak{b} \subset \mathcal{V}$. We define the ϵ-escape time $\mathrm{E}_{\mathfrak{b}}(\epsilon)$ as the time at which the Markov chain escaped $\mathfrak{b}$, i.e., is located in its complement $\mathfrak{b}^c$, with probability at least ϵ, starting from the stationary distribution $\pi_{\mathfrak{b}}$ on $\mathfrak{b}$. We recall that $\pi_{\mathfrak{b}}(j) = \pi(j)/\pi(\mathfrak{b})$ if $j \in \mathfrak{b}$, and is zero elsewhere. This gives

$$\mathrm{E}_{\mathfrak{b}}(\epsilon) = \min\{t \mid (P^t \pi_{\mathfrak{b}})(\mathfrak{b}^c) \geq \epsilon\}.$$

We will generally be interested in $\epsilon \in \Theta(\pi(\mathfrak{b}^c))$, and we will loosely define the *escape time* $\mathrm{E}_{\mathfrak{b}}$ as the smallest order of $E_{\mathfrak{b}}(\epsilon)$ for $\epsilon \in \Theta(\pi(\mathfrak{b}^c))$. To get some intuition on the escape time, we prove some properties of independent interest.

Lemma 15.

- *If P is reversible then the escape time is symmetric:*

$$(P^t \pi_{\mathfrak{b}})(\mathfrak{b}^c) \in \Theta(\pi(\mathfrak{b}^c)) \Leftrightarrow (P^t \pi_{\mathfrak{b}^c})(\mathfrak{b}) \in \Theta(\pi(\mathfrak{b})).$$

• *We can bound* $E_{\mathfrak{b}} \in \Omega\left(\frac{\pi(\mathfrak{b}^c)}{\Phi(\mathfrak{b})}\right)$ *and* $E_{\mathfrak{b}} \in O(\tau(\pi(\mathfrak{b}^c))) \in O\left(\tau \log \frac{1}{\pi(\mathfrak{b}^c)}\right)$.

Proof. If P is reversible then $P = \operatorname{diag}(\sqrt{\pi})D^t \operatorname{diag}(\sqrt{\pi})^{-1}$, with D the symmetric discriminant matrix. As a consequence, $P^t(j', j)\pi(j) = D^t(j', j)\sqrt{\pi(j')}\sqrt{\pi(j)} = P^t(j, j')\pi(j')$ and

$$\begin{aligned}(P^t\pi_{\mathfrak{b}})(\mathfrak{b}^c) &= \frac{\sum_{j\in\mathfrak{b}, j'\in\mathfrak{b}^c} P^t(j', j)\pi(j)}{\pi(\mathfrak{b})} \\ &= \frac{\sum_{j\in\mathfrak{b}, j'\in\mathfrak{b}^c} P^t(j, j')\pi(j')}{\pi(\mathfrak{b})} = \frac{\pi(\mathfrak{b}^c)}{\pi(\mathfrak{b})}(P^t\pi_{\mathfrak{b}^c})(\mathfrak{b}).\end{aligned}$$

This shows that $(P^t\pi_{\mathfrak{b}})(\mathfrak{b}^c) \in \Theta(\pi(\mathfrak{b}^c))$ if and only if $(P^t\pi_{\mathfrak{b}^c})(\mathfrak{b}) \in \Theta(\pi(\mathfrak{b}))$, proving the first bullet.

To prove the second bullet, recall from Lemma 3, Section 3.2, that $(P^t\pi_{\mathfrak{b}})(\mathfrak{b}^c) \leq t\Phi(\mathfrak{b})$, so that we can easily bound $\mathrm{E}_{\mathfrak{b}}(\epsilon) \geq \epsilon/\Phi(\mathfrak{b})$ and so

$$\mathrm{E}_{\mathfrak{b}} \in \Omega(\pi(\mathfrak{b}^c)/\Phi(\mathfrak{b})).$$

To see that $\mathrm{E}_{\mathfrak{b}} \in O(\tau(\pi(\mathfrak{b}^c)))$ note that from Chapter 1 we know that the distribution $P^t\pi_{\mathfrak{b}}$ will be ϵ-close to π in TV-distance after $\tau(\epsilon)$ steps. In particular this implies that $|(P^t\pi_{\mathfrak{b}})(\mathfrak{b}^c) - \pi(\mathfrak{b}^c)| \leq \epsilon$ and so $(P^t\pi_{\mathfrak{b}})(\mathfrak{b}^c) \geq \pi(\mathfrak{b}^c) - \epsilon \in \Theta(\pi(\mathfrak{b}^c))$ if we choose $\pi(\mathfrak{b}^c) > \epsilon \in \Theta(\pi(\mathfrak{b}^c))$. □

We will use our QFF algorithm to show that *quantum walks can escape large sets quadratically faster*, starting from the superposition $|\sqrt{\pi_{\mathfrak{b}}}\rangle$. Here we call $\mathfrak{b}$ a *large set* with respect to a distribution π if $\pi(\mathfrak{b}) \in \Theta(1)$. This is in accordance to the definition in [141]. Combining the two bullets from the above lemma we see that $\mathrm{E}_{\mathfrak{b}} \in O(\tau)$ if $\mathfrak{b}$ is a large set. In Example 16 we show that in certain cases it is much smaller. Similarly to the existing quantum walk search literature, we assume that the quantum state $|\sqrt{\pi_{\mathfrak{b}}}\rangle$ is given to us as a primitive. One could also imagine that the set $\mathfrak{b}$ corresponds to a list of nodes that we have already visited, as in a tree search on a graph, from which it is feasible that we can construct $|\sqrt{\pi_{\mathfrak{b}}}\rangle$. In the end of this section we will go more into detail on these assumptions.

Proposition 19 (Escaping in $\sqrt{\mathbf{E}}$). *Consider a reversible Markov chain P with an escape time $E_{\mathfrak{b}}(\epsilon)$ for some large set $\mathfrak{b} \subset \mathcal{V}$ and $\epsilon \in \Theta(\pi(\mathfrak{b}^c))$. Then running* $\mathbf{QFF}(|\sqrt{\pi_{\mathfrak{b}}}\rangle, P, E_{\mathfrak{b}}(\epsilon), \epsilon')$ *with* $\epsilon' = \sqrt{\frac{\pi(\mathfrak{b})}{2\pi(\mathfrak{b}^c)}}\epsilon$, *and measuring the outcome, returns an element in $\mathfrak{b}^c$ with probability $\Theta(\pi(\mathfrak{b}^c))$. This scheme requires* $\Theta\left(\sqrt{E_{\mathfrak{b}}(\epsilon) \log \frac{1}{\pi(\mathfrak{b}^c)}}\right)$ *QW steps.*

Proof. The algorithm $\mathbf{QFF}(|\sqrt{\pi_{\mathfrak{b}}}\rangle, P, t, \epsilon')$ returns a quantum state $|\psi\rangle$ for which it holds that $\||\psi\rangle - |D^t\sqrt{\pi_{\mathfrak{b}}}\rangle\| \leq \epsilon'$. This implies that $\|\Pi_{\mathfrak{b}^c}(|\psi\rangle - |D^t\sqrt{\pi_{\mathfrak{b}}}\rangle)\| \leq \epsilon'$ and therefore

$$\|\Pi_{\mathfrak{b}^c}|\psi\rangle\| \geq \|\Pi_{\mathfrak{b}^c}|D^t\sqrt{\pi_{\mathfrak{b}}}\rangle\| - \epsilon'.$$

Using the below lemma, we see that

$$\|\Pi_{\mathfrak{b}^c}|D^t\sqrt{\pi_{\mathfrak{b}}}\rangle\| \geq \|\Pi_{\mathfrak{b}^c}D^t|\sqrt{\pi_{\mathfrak{b}}}\rangle\| \geq \sqrt{\frac{\pi(\mathfrak{b})}{\pi(\mathfrak{b}^c)}}(P^t\pi_{\mathfrak{b}})(\mathfrak{b}^c).$$

For $t = \mathrm{E}_{\mathfrak{b}}(\epsilon)$ we know that $(P^t\pi_{\mathfrak{b}})(\mathfrak{b}^c) \geq \epsilon$. Combined with the above inequalities shows that $\|\Pi_{\mathfrak{b}^c}|\psi\rangle\| \geq \sqrt{\frac{\pi(\mathfrak{b})}{\pi(\mathfrak{b}^c)}}\epsilon - \epsilon'$. By our choice $\epsilon' = \sqrt{\frac{\pi(\mathfrak{b})}{2\pi(\mathfrak{b}^c)}}\epsilon$, and the fact that $\epsilon \in \Theta(\pi(\mathfrak{b}^c))$, we find that

$$\|\Pi_{\mathfrak{b}^c}|\psi\rangle\|^2 \geq (1 - 1/\sqrt{2})^2 \frac{\pi(\mathfrak{b})}{\pi(\mathfrak{b}^c)}\epsilon^2 \in \Theta(\pi(\mathfrak{b})\pi(\mathfrak{b}^c)) \in \Theta(\pi(\mathfrak{b}^c)),$$

using our assumption that $\mathfrak{b}$ is a large set and hence $\pi(\mathfrak{b}) \in \Theta(1)$. As a consequence, a measurement $\{\Pi_{\mathfrak{b}^c}, I - \Pi_{\mathfrak{b}^c}\}$ on the output state $|\psi\rangle$ will return an element of $\mathfrak{b}^c$ with probability $\Theta(\pi(\mathfrak{b}^c))$. The success probability of the algorithm $\|D^t|\sqrt{\pi_{\mathfrak{b}}}\rangle\|^2 \geq \|\Pi_{\mathfrak{b}}D^t|\sqrt{\pi_{\mathfrak{b}}}\rangle\|^2 \in \Theta(1)$. Since $\epsilon' \in \Theta(\sqrt{\pi(\mathfrak{b})\pi(\mathfrak{b}^c)}) \in \Theta(\sqrt{\pi(\mathfrak{b}^c)})$, the algorithm will require an expected number of QW steps

$$O\left(\sqrt{t \log\left(\frac{1}{\epsilon'\|D^t|v\rangle\|}\right)}\right) \in O\left(\sqrt{\mathrm{E}_{\mathfrak{b}}(\epsilon) \log \frac{1}{\pi(\mathfrak{b}^c)}}\right). \qquad \square$$

The proof makes use of the below lemma, showing that if the original dynamics $P^t\pi_{\mathfrak{b}}$ are in a subset $\mathfrak{b}'$ with a certain probability, so that $(P^t\pi_{\mathfrak{b}})(\mathfrak{b}') \geq \epsilon$, then the probability of finding the quantum state $|D^t\sqrt{\pi_{\mathfrak{b}}}\rangle$ in $\mathfrak{b}'$, being $\|\Pi_{\mathfrak{b}'}|D^t\sqrt{\pi_{\mathfrak{b}}}\rangle\|^2 \geq \|\Pi_{\mathfrak{b}'}D^t|\sqrt{\pi_{\mathfrak{b}}}\rangle\|^2$, can also be lower bounded.

Lemma 16. *Let P be a reversible Markov chain, D its discriminant matrix, and $\mathfrak{b}, \mathfrak{b}' \subseteq \mathcal{V}$. If $(P^t\pi_{\mathfrak{b}})(\mathfrak{b}') \geq \epsilon$, then*

$$\|\Pi_{\mathfrak{b}'}D^t|\sqrt{\pi_{\mathfrak{b}}}\rangle\|^2 \geq \frac{\pi(\mathfrak{b})}{\pi(\mathfrak{b}')}\epsilon^2.$$

Proof. If P is reversible then $D = \mathrm{diag}(\sqrt{\pi})^{-1}P\,\mathrm{diag}(\sqrt{\pi})$, so that

$$D^t|\sqrt{\pi_{\mathfrak{b}}}\rangle = \mathrm{diag}(\sqrt{\pi})^{-1} \cdot P^t \cdot \sum_{j\in\mathfrak{b}} \frac{\pi(j)}{\sqrt{\pi(\mathfrak{b})}}|j\rangle.$$

The last term we can rewrite as $\sum_{j\in\mathfrak{b}} \frac{\pi(j)}{\sqrt{\pi(\mathfrak{b})}}|j\rangle = \sqrt{\pi(\mathfrak{b})}\sum_{j\in\mathfrak{b}} \frac{\pi(j)}{\pi(\mathfrak{b})}|j\rangle = \sqrt{\pi(\mathfrak{b})}\pi_{\mathfrak{b}}$, proving that

$$D^t|\sqrt{\pi_{\mathfrak{b}}}\rangle = \sqrt{\pi(\mathfrak{b})}\,\mathrm{diag}(\sqrt{\pi})^{-1}(P^t\pi_{\mathfrak{b}}) = \sqrt{\pi(\mathfrak{b})}(P^t\pi_{\mathfrak{b}})./\sqrt{\pi},$$

where by "./" we mean pointwise division. This shows that

$$\|\Pi_{\mathfrak{b}'}D^t|\sqrt{\pi_{\mathfrak{b}}}\rangle\|^2 = \pi(\mathfrak{b})\|\Pi_{\mathfrak{b}'}(P^t\pi_{\mathfrak{b}})./\sqrt{\pi}\|^2.$$

In order to bound this last property, we will now make use of the following fact, which we prove in the below Lemma 17: *if $p, q \in \mathbb{R}^N$, $p \geq 0$, $q > 0$, and $\|p\|_1 = \|q\|_1 = 1$, then $\|p./\sqrt{q}\| \geq 1$.* If now we set $p, q \in \mathbb{R}^{|\mathfrak{b}'|}$ by

$$\forall j \in \mathfrak{b}' : \quad p(j) = \frac{(\Pi_{\mathfrak{b}'}P^t\pi_{\mathfrak{b}})(j)}{\|\Pi_{\mathfrak{b}'}(P^t\pi_{\mathfrak{b}})\|_1}, \quad q(j) = \frac{(\Pi_{\mathfrak{b}'}\pi)(j)}{\|\Pi_{\mathfrak{b}'}\pi\|_1} = \frac{(\Pi_{\mathfrak{b}'}\pi)(j)}{\pi(\mathfrak{b}')},$$

then this implies that

$$\|\Pi_{\mathfrak{b}'}(P^t\pi_{\mathfrak{b}})./\sqrt{\pi}\| = \frac{\|\Pi_{\mathfrak{b}'}(P^t\pi_{\mathfrak{b}})\|_1}{\sqrt{\pi(\mathfrak{b}')}}\|p./\sqrt{q}\| \geq \frac{\|\Pi_{\mathfrak{b}'}(P^t\pi_{\mathfrak{b}})\|_1}{\sqrt{\pi(\mathfrak{b}')}}.$$

Combining this with the fact that

$$\|\Pi_{\mathfrak{b}'} D^t |\sqrt{\pi_{\mathfrak{b}}}\rangle\|^2 = \pi(\mathfrak{b}) \|\Pi_{\mathfrak{b}'}(P^t \pi_{\mathfrak{b}})./\sqrt{\pi}\|^2,$$

we find the claimed inequality. □

Lemma 17. *If $p, q \in \mathbb{R}^N$, $p \geq 0$, $q > 0$, and $\|p\|_1 = \|q\|_1 = 1$, then $\|p./\sqrt{q}\| \geq 1$.*

Proof. We will prove this inequality by induction. It trivially holds for $N = 1$, and so it suffices to prove that if it holds for general $N = M$ then it also holds for $N = M + 1$. We will now assume the that it holds for $N = M$. We can rewrite $p, q \in \mathbb{R}^{N+1}$ as

$$p = \begin{bmatrix} \alpha p_N \\ 1 - \alpha \end{bmatrix}, \quad q = \begin{bmatrix} \beta q_N \\ 1 - \beta \end{bmatrix},$$

with $1 \geq \alpha \geq 0$, $1 > \beta > 0$, and $p_N, q_N \in \mathbb{R}^N$ such that $p_N \geq 0$, $q_N > 0$ and $\|p_N\|_1 = \|q_N\|_1 = 1$. Now

$$\|p./\sqrt{q}\|^2 = \sum_{j=1}^{N+1} \frac{p(j)^2}{q(j)} = \frac{\alpha^2}{\beta} \sum_{j=1}^{N} \frac{p_N(j)^2}{q_N(j)} + \frac{(1-\alpha)^2}{1-\beta}.$$

By our induction hypothesis, we know that $\sum_{j=1}^{N} \frac{p_N(j)^2}{q_N(j)} = \|p_N./\sqrt{q_N}\|^2 \geq 1$, so that

$$\|p./\sqrt{q}\|^2 \geq \frac{\alpha^2}{\beta} + \frac{(1-\alpha)^2}{1-\beta}.$$

For fixed $1 > \beta > 0$, this expression reaches a minimum of 1 if $\alpha = \beta$, proving our statement. □

In Proposition 19 we use the bound in Lemma 16 for $\mathfrak{b}' = \mathfrak{b}^c$ and $\epsilon \in \Theta(\pi(\mathfrak{b}^c))$, so that $\|\Pi_{\mathfrak{b}'} D^t |\sqrt{\pi_{\mathfrak{b}}}\rangle\|^2 \in \Omega(\pi(\mathfrak{b})\pi(\mathfrak{b}^c))$. We then use the condition that $\mathfrak{b}$ is a large set, so that $\pi(\mathfrak{b}) \in \Theta(1)$, to bound $\|\Pi_{\mathfrak{b}'} D^t |\sqrt{\pi_{\mathfrak{b}}}\rangle\|^2 \in \Omega(\pi(\mathfrak{b}^c))$. If $\mathfrak{b}$ is not a large set then we cannot use this bound.

Escaping sets has algorithmic use for graph problems. One could imagine that a certain well-connected part of a graph was explored in a previous stage, such as the neighborhood of a local minimum, and we wish to escape this neighborhood to find new elements. In the following we will discuss a related line of research on using quantum walks for search problems, and we will compare our new result to the existing literature.

Quantum Search Algorithms

In a long line of research, including but not limited to [3, 19–22, 37, 106, 107, 142–145], the usage of quantum walks to speed up the *hitting time* of Markov chains on graphs was investigated. The hitting time $\mathrm{HT}_{\mathfrak{b}}$ of a subset $\mathfrak{b} \subset \mathcal{V}$ is defined as the expected number of steps before a Markov chain hits an element in $\mathfrak{b}$, starting from its stationary distribution π. One of the most important contributions is the work by Szegedy [3], in which he introduced the quantum walk formalism underlying this thesis part. In that paper he showed that, provided that we are given the superposition

$|\sqrt{\pi}\rangle$, then a quantum walk can *detect* the presence of marked elements in $\Theta(\sqrt{\mathrm{HT}_\flat})$ QW steps. Using quantum walks, it turns out that actually finding marked elements is more difficult than detecting them. A series of efforts culminated in the work by Krovi, Magniez, Ozols and Roland [22] in which they show that if there is a unique marked element, $|\flat| = 1$, then quantum walks can effectively find a marked element in $\Theta(\sqrt{\mathrm{HT}_\flat})$ QW steps. If there are more elements, then their algorithm needs $\Theta(\sqrt{\mathrm{HT}^+_\flat})$ QW steps, where $\mathrm{HT}^+_\flat$ is a property called the *extended hitting time*, and can be unboundedly larger than the hitting time (see [22, 37, 146] for discussions). This leaves the problem of quadratically accelerating the hitting time of large sets as one of the main open questions.

Our result sheds some new light on this issue. Indeed in many cases the escape time and hitting time are equal, so that our algorithm effectively solves the hitting problem. Examples of such are when the large set $\flat$ is a subtree (a node and all its descendants) of a tree graph, or a square subset of a lattice. In the below example we explicitly demonstrate such a setting, showing how our algorithm improves the algorithm by Krovi et al [22], which provides no acceleration in this case.

Example 16 (Escaping on $\mathbb{Z}^2_{\sqrt{N}} - \mathbb{Z}_N$). Consider the lazy random walk on the graph $\mathbb{Z}^2_{\sqrt{N}} - \mathbb{Z}_N$, shown below in Figure 10.1. A cut halfway through the lattice results in the bipartition $(\flat, \flat^c)$. Using standard random walk techniques it is easy to see that $\pi(\flat) = \frac{1}{3} + O(N^{-1/2}) \in \Theta(1)$ and the escape time $\mathrm{E}(\flat) \in \Theta(\mathrm{HT}(\flat^c)) \in \Theta(N)$. By Proposition 19 we know that quantum walks can escape $\flat$ in $O(\sqrt{N})$ QW steps. Contrarily, we can show that the extended hitting time $\mathrm{HT}^+(\flat^c) \in \Omega(N^2)$, since it can be bounded by the time it takes to mix from $\pi_\flat$ to π (see e.g. [145]), and it requires $\Omega(N^2)$ steps to mix on the path $\mathbb{Z}_N$. The algorithm by Krovi et al [22] therefore requires $\Omega(N)$ QW steps, providing no speedup with respect to the classical hitting time. This graph is a typical case where the escape time is much smaller than the mixing time when starting from $\pi_\flat$, so that our algorithm will outperform the existing algorithm in [22]. △

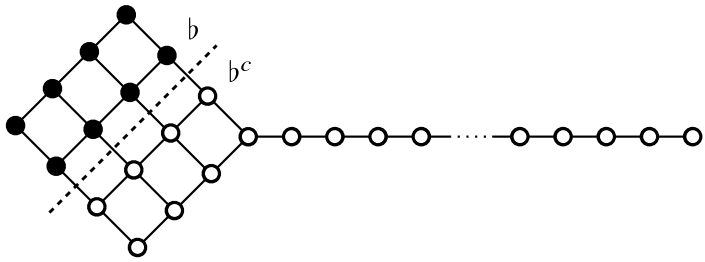


Fig. 10.1 Random walk on the graph $\mathbb{Z}^2_{\sqrt{N}} - \mathbb{Z}_N$. For the large set $\flat$ shown, the escape time and hitting time are of the same order: $\mathrm{E}_{\pi(\flat^c)/2}(\flat) \in \Theta(\mathrm{HT}(\flat^c)) \in \Theta(N)$. Our QFF algorithm allows to escape $\flat$ quadratically faster, using $\Theta(\sqrt{N})$ QW steps. Since the extended hitting time $\mathrm{HT}^+(\flat^c) \in \Omega(N^2)$, the algorithm by Krovi et al [22] requires $\Omega(N)$ QW steps, providing no speedup.

We also mention that the escape time can be preferable over the hitting time in cases where the *checking cost* is high. Given an element $j \in \mathcal{V}$, the checking cost

C is the cost associated to checking whether $j \in \flat$ or not. For example, C could be the cost of verifying whether a given element represents a valid solution of some problem. In accordance with [21], we will denote by $\mathsf{S}_\flat$ the cost of creating the initial superposition $|\sqrt{\pi_\flat}\rangle$, and by U the cost of implementing a single QW step. The total expected cost for finding an element of $\flat^c$ using our algorithm is then

$$\Theta\Big(\frac{1}{\pi(\flat^c)}\big(\mathsf{S}_\flat + \mathsf{C} + \sqrt{\mathrm{E}_\flat \log(1/\pi(\flat^c))}\mathsf{U}\big)\Big).$$

Indeed, we create $|\sqrt{\pi_\flat}\rangle$, with a cost $\mathsf{S}_\flat$, then we run our QFF algorithm requiring $\sqrt{\mathrm{E}_\flat \log \frac{1}{\pi(\flat^c)}}$ QW steps, and then we perform a measurement and check the solution. This returns a marked element with probability $\pi(\flat^c)$. Especially in the case where $\flat^c$ is also a large set, so $\pi(\flat^c) \in \Theta(1)$, this bound is interesting. In this case it becomes

$$\Theta\big(\mathsf{S}_\flat + \mathsf{C} + \sqrt{\mathrm{E}_\flat \log(1/\pi(\flat^c))}\mathsf{U}\big),$$

so that the checking cost is constant. For comparison, the algorithm by Krovi et al, provided that it also is given the initial state $|\sqrt{\pi_\flat}\rangle$, has a total cost $\Theta(\mathsf{S}_\flat + \sqrt{\mathrm{HT}^+_{\flat^c}}(\mathsf{C}+\mathsf{U}))$. This can be bad if the checking cost is high. In this setting the QW algorithm by Magniez et al [21] is more interesting. It has a cost

$$\Theta\Big(\mathsf{S}_\flat + \frac{1}{\sqrt{\pi(\flat^c)}}\big(\mathsf{C} + \frac{1}{\sqrt{\delta}}\mathsf{U}\big)\Big),$$

with δ the spectral gap of the Markov chain. If $\pi(\flat^c) \in \Theta(1)$, then $\mathrm{E}_\flat \in O(\tau)$, and by Proposition 1 we know that $\tau(\epsilon) \in O\big(\frac{1}{\delta}\log\frac{1}{\min_j \pi(j)}\big)$, so that our algorithm is at most a log-factor slower, yet it will typically be faster. Indeed, in the above example we have that $\pi(\flat^c) \in \Theta(1)$ and $\mathrm{E}_\flat \in \Theta(N)$ whereas $\delta \in \Theta\big(\frac{1}{N^2}\big)$, so that our algorithm performs quadratically better. Finally also in the work by Dohotaru and Høyer [145] a QW algorithm for the case of a *unique* marked element was proposed with a total cost $\Theta\big(\mathsf{S}_\flat + \frac{1}{\sqrt{\pi(\flat)}}\mathsf{C} + \sqrt{\mathrm{HT}_{\flat^c}}\mathsf{U}\big) \in \Theta\big(\mathsf{S}_\flat + \mathsf{C} + \sqrt{\mathrm{HT}^c_\flat}\mathsf{U}\big)$ if $\pi(\flat) \in \Theta(1)$. It seems unlikely that such cost could be generalized to larger sets as we do.

10.2 Quantum State Generation

We already mentioned that an important feature of our QFF algorithm is that it allows to simulate and accelerate the intermediate dynamics of a Markov chain. This complements the bulk of the existing results on QW acceleration of Markov chains, which focuses on accelerating the limit behavior [3, 15, 22, 41, 42]. A setting where the intermediate dynamics may be of interest is the problem of *quantum sampling* or *quantum state generation*, as proposed in [40]. This amounts to the following problem: given a classical algorithm that efficiently samples from some distribution π, create the corresponding quantum state $|\sqrt{\pi}\rangle = \sum_j \sqrt{\pi(j)}|j\rangle$. They show that if we can do so efficiently, then all problems in the complexity class

SZK called *statistical zero knowledge* have an efficient solution, and for instance the graph isomorphism problem can be solved efficiently on a quantum computer.

In many practical cases the classical algorithm for generating π consists of an efficiently implementable Markov chain P, which is run for a certain number of steps on some initial state. If the goal distribution π is the stationary distribution of P, then the corresponding quantum state $|\sqrt{\pi}\rangle$ is the stationary state of D. Phase estimation techniques can then be used to retrieve $|\sqrt{\pi}\rangle$. This is the essence of the quantum sampling algorithms in for instance [40–42]. We will discuss a situation where π is not the stationary distribution of P but an intermediate state $\pi = P^t e_j$, so that similar quantum sampling algorithms would fail. This could for example be the case when G is an infinite graph, or when we are interested in transient dynamics (e.g., frustrated states or local minima). We demonstrate how QFF can then be used to generate $|\sqrt{\pi}\rangle$. The demonstration is mainly a proof of principle, yet we mention that the generated states are for example useful resources for a recently discovered class of quantum error-correcting codes called "binomial quantum codes" [147].

Quantum Simulating a Gaussian

Let P be the transition matrix of a lazy random walk on the integer line $\mathbb{Z}$, standing still with probability $1/2$ and otherwise going left or right with probability $1/4$. By the discrete local limit theorem [148] we know that

$$(P^{2t}e_0)(j) = P^{2t}(j,0) = \frac{1}{\sqrt{2\pi t}} e^{-j^2/(2t)} + o\Big(\frac{1}{\sqrt{t}}\Big).$$

The distribution $P^{2t}e_0$ thus approaches a normal distribution $\mathcal{N}_t$ with mean 0 and variance t. Such spreading behavior is very characteristic of classical diffusive dynamics. In Figure 10.2 we show $P^{2t}e_0$ for $t = 200$. Now we wish to create the corresponding quantum state, that is, a quantum state $|\sqrt{\mathcal{N}_t}\rangle$ such that

$$|\sqrt{\mathcal{N}_t}\rangle \sim \sum_{j\in\mathbb{Z}} e^{-j^2/(4t)}|j\rangle.$$

If we simply run the quantum walk W, naturally associated to P as in Section 8.1, then we find back a very different behavior: it has a very irregular shape, with many peaks due to interference effects, and a standard deviation in $\Theta(t)$. In Figure 10.2 we show the outcome distribution of a measurement of the quantum state $W^t|0,\flat\rangle$ for $t = 100$. We can remedy this by using our QFF scheme, forcing the quantum walk into taking the classical random walk form.

Generating a quantum Gaussian $|\sqrt{\mathcal{N}_t}\rangle$ in $\widetilde{O}(t^{3/4})$ steps. Since the initial state $|0\rangle$ is a basis state around which we can reflect, we can apply the QFF scheme with Grover search **QFFg**$(|0\rangle, P, t, \epsilon)$ (note that P is symmetric so $P = D$). This scheme returns an ϵ-approximation of the state

$$\frac{P^t|0\rangle}{\|P^t|0\rangle\|} = |\sqrt{\mathcal{N}_t}\rangle + o\Big(\frac{1}{\sqrt{t}}\Big),$$

with constant probability in a number of QW steps in

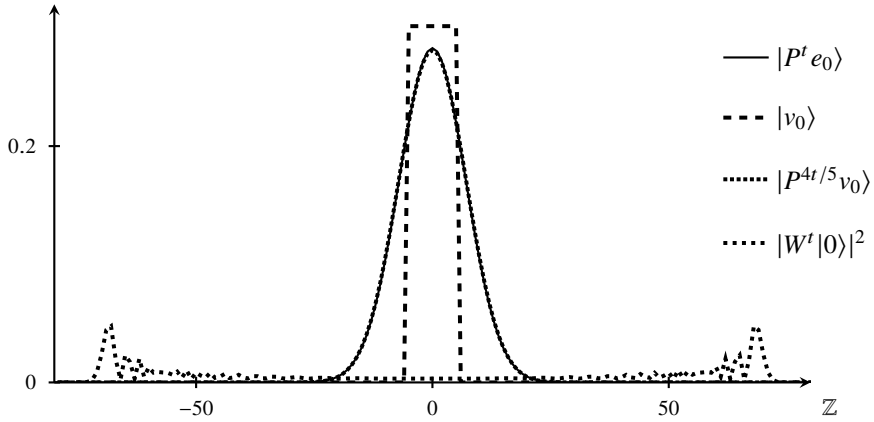


Fig. 10.2 Quantum Gaussian states for $t = 200$. QFF allows to approximately generate the quantum state $P^t|0\rangle/\|P^t|0\rangle\| = |\sqrt{\mathcal{N}_t}\rangle + o(1/\sqrt{t})$ in $\widetilde{O}(t^{3/4})$ QW steps. If we are given the initial uniform superposition $|v_0\rangle$, then we can generate the quantum state $P^{4t/5}|v_0\rangle/\|P^{4t/5}|v_0\rangle\| = |\sqrt{\mathcal{N}_t}\rangle + o(1/\sqrt{t})$ quadratically faster in $O(\sqrt{t})$ QW steps. For comparison, we also show the measurement outcome probability distribution if we would simply run the quantum walk W for t steps.

$$\Theta\left(\frac{\sqrt{t}}{\|P^t|0\rangle\|}\sqrt{\log\frac{1}{\|P^t|0\rangle\|}}\right).$$

To bound $\|P^t|0\rangle\|$, we interpret $P^t|0\rangle$ as a probability vector, which will approach a normal distribution with standard deviation $O(\sqrt{t})$. Using the fact that for any vector v it holds that $\|v\|^2 \geq \|v\|_1^2/|\mathrm{supp}(v)|$ with $|\mathrm{supp}(v)|$ the size of the support of v, we can easily see that

$$\|P^t|0\rangle\|^2 \geq \|(P^t|0\rangle)_{[-\sqrt{t},\sqrt{t}]}\|^2 \geq \frac{1}{2\sqrt{t}}\|(P^t|0\rangle)_{[-\sqrt{t},\sqrt{t}]}\|_1^2 \in \Theta(1/\sqrt{t}). \tag{10.1}$$

As a consequence, applying $\mathbf{QFFg}(|0\rangle, P, t, \epsilon)$ succeeds with constant probability in $O\left(t^{3/4}\log^{1/2} t\right)$ QW steps, returning an approximation of the state $|\sqrt{\mathcal{N}_t}\rangle$.

Generating a quantum Gaussian $|\mathcal{N}_t\rangle$ in $O(\sqrt{t})$ steps. The above algorithm is slowed down by the fact that the initial success probability, before applying the Grover routine of the **QFFg** algorithm, is $\|P^t|0\rangle\|^2 \in \Theta(1/\sqrt{t})$. This can be resolved if instead of from an initial state $|0\rangle$, we start from an initial uniform superposition $|v_0\rangle$ over the states in $[-\sqrt{t}/2, \sqrt{t}/2]$ (for clarity of exposition assume that $\sqrt{t}/2 \in \mathbb{Z}$)

$$|v_0\rangle = \frac{1}{\sqrt{\sqrt{t}+1}}\sum_{j=-\sqrt{t}/2}^{\sqrt{t}/2}|j\rangle.$$

Such a situation, where we assume that we are given an initial resource state $|v_0\rangle$, is typically considered in quantum simulated annealing schemes [40, 41, 43, 108], and in [149] in a different context. Running $\mathbf{QFF}(|v_0\rangle, P, t', \epsilon)$ now outputs an ϵ-approximation of the quantum state $|P^{t'}v_0\rangle$ with success probability $\Theta(1/\|P^{t'}|v_0\rangle\|^2)$. If we choose $t' = 4t/5$, then this state will approximate the

original goal state $|\sqrt{N_t}\rangle$[1], as is clearly shown in Figure 10.2. We can bound the success probability by noting that $\|P^{t'}|v_0\rangle\|^2 = \|P^{t'}v_0\|^2/\|v_0\|^2$, where now v_0 is the uniform distribution over $[-\sqrt{t}/2, \sqrt{t}/2]$. We can easily calculate that $\|v_0\|^2 = 1/(\sqrt{t}+1)$, and using a similar reasoning as in (10.1) we find that $\|P^{t'}v_0\|^2 \in \Theta(1/\sqrt{t})$. This shows that $\|P^{t'}|v_0\rangle\|^2 \in \Theta(1)$, so that the scheme will have a constant success probability. The total number of QW steps required then becomes

$$O\left(\frac{\sqrt{t'}}{\|P^{t'}|v_0\rangle\|}\log^{1/2}\frac{1}{\|P^{t'}|v_0\rangle\|}\right) \in O(\sqrt{t}).$$

This proves that if we are given the uniform superposition $|v_0\rangle$, we can generate an approximation of the quantum Gaussian state $|\sqrt{N_t}\rangle$ in $O(\sqrt{t})$ expected QW steps.

10.3 Discussion and Outlook

Finalizing this thesis part, we wish to provide some further observations and discussion on the QFF scheme, and present some remaining open questions and research directions.

Improving the QFF Algorithm: Parameter Dependence and Irreversible Markov Chains

QFF requires $O\big(t^{1/2}\log^{1/2}(\epsilon\|D^t|v\rangle\|)^{-1}\big)$ QW steps to create an ϵ-approximation of the state $|D^t v\rangle$ with success probability $\|D^t|v\rangle\|^2$. The success probability can be boosted to a constant using $O(1/\|D^t|v\rangle\|)$ reflections around the initial state $|v\rangle$, in total requiring a number of QW steps in

$$O\left(\frac{\sqrt{t}}{\|D^t|v\rangle\|}\sqrt{\log\frac{1}{\epsilon\|D^t|v\rangle\|}}\right).$$

We will restrict our discussion to this case. As we elaborate below, up to the $\log\|D^t|v\rangle\|^{-1}$-factor it seems that the dependence of the algorithm's performance on the parameters t, ϵ and $\|D^t|v\rangle\|$ is optimal. We leave improvement on the $\log\|D^t|v\rangle\|^{-1}$ factor as an open question.

Towards the t and ϵ dependency, we can look at the random walk on $\mathbb{Z}$. If we tolerate an ϵ error, then we can confine the probability distribution of a t-step random walk to the $\Theta\big(t^{1/2}\log^{1/2}\epsilon^{-1}\big)$ neighborhood of the initial state. Since the QW has the same locality constraints as the RW, it needs $\Omega\big(t^{1/2}\log^{1/2}\epsilon^{-1}\big)$ QW steps to spread out over this interval. Notice the parallel with the trade-off in the Chebyshev approximation, Lemma 10. A very similar argument also shows why in general QFF cannot work for irreversible Markov chains. Indeed, consider the Markov chain on $\mathbb{Z}$ which simply moves to the right every step, $P(i+1,i)=1$ and $P(i-1,i)=0$. This walk is clearly not reversible, as the direction of its motion

[1] The reduced exponent $4t/5$ is a consequence of the fact that the initial state $|v_0\rangle$ already has some width, contrary to $|0\rangle$.

reverses when running the time forward or backward. When starting in the origin, the walk will be on node t after t steps. A local QW requires $\Theta(t)$ steps to reach this point, so that no fast-forwarding is possible.

Towards the $\|D^t|v\rangle\|$-dependency, we base our intuition on a standard quantum algorithm that relies on the optimality of Grover search [150], showing that the number of reflections in Proposition 16 is optimal up to a constant. Roughly, if QFF starting from $|j\rangle$ would allow to return the superposition over the nodes of the complete graph K_N using $o(1/\|D^t|j\rangle\|) = o(\sqrt{N})$ reflections around $|j\rangle$, then we could run this quantum algorithm backwards to find the element $|j\rangle$ using $o(\sqrt{N})$ reflections around $|j\rangle$. This is however known to be impossible.

As a consequence it seems that we cannot improve the t and $\|D^t|v\rangle\|$ dependency separately, yet it is possible that we can improve their combined dependency. A similar situation occurred in quantum search algorithms, where we search for a marked node j using some Markov chain with stationary distribution π and spectral gap δ. An algorithm by Magniez et al [21] proposed a quantum algorithm for finding the node j that required $\Theta(1/\sqrt{\delta\pi(j)})$ QW steps. This is optimal in both $\pi(j)$ and δ separately. The later algorithm by Krovi et al [22] however improved this to $\Theta(\sqrt{\mathrm{HT}(j)})$, where the hitting time $\mathrm{HT}(j)$ is a parameter that combines both quantities and can be upper bounded by $1/(\delta\pi(j))$. On for example the dumbbell graph $K_N - K_N$, their bound gives $\Theta(N)$ which improves the bound $\Theta(1/\sqrt{\delta\pi(j)}) \in \Theta(N^{3/2})$. In Appendix A we show that the same situation arises for quantum state generation. For QFF we see that a similar improvement might be possible when considering for instance the random walk on $\mathbb{Z}$, as in Section 10.2. If we run QFF on the initial state $|0\rangle$ then it requires $\widetilde{\Theta}(t^{3/4})$ QW steps to generate the quantum Gaussian $|\sqrt{\mathcal{N}_t}\rangle$. If however we first create a quantum state similar to the initial superposition $|v_0\rangle$, which by the techniques in Appendix A seems feasible in $\Theta(\sqrt{t})$ QW steps, and then apply QFF, we can generate the quantum Gaussian in $\Theta(\sqrt{t})$ QW steps.

Mixing with QFF

In the light of the first thesis part, we briefly discuss how QFF can be used for mixing, i.e., classically sampling an element according to some distribution π over the node set. A very common way to tackle this problem is by solving the more difficult quantum state generation problem, i.e., creating the quantum state $|\sqrt{\pi}\rangle$. We can then simply measure this state in the node basis to retrieve a classical sample from π. This approach is taken in a range of papers, containing but not limited to [40–43, 108, 149]. At the end of Chapter 8 we discussed how QFF allows to create the quantum state $|\sqrt{\pi}\rangle$, for π a uniform distribution, in $\widetilde{\Theta}(\sqrt{\tau|\mathcal{V}|})$ QW steps. With additional techniques this can be further improved, see Appendix A. However, as discussed above, the $\Omega(\sqrt{|\mathcal{V}|})$ lower bound seems inherent in quantum algorithms for creating a superposition over the node set (exceptions can be found when there are additional resources such as access to a sequence of *slowly evolving Markov chains* [40, 108] or additional information on π and $\mathcal{V}$ [149]). For many important classical Markov chain algorithms this is overly excessive. For example,

many algorithms build on *rapidly mixing Markov chains* [70], which have a mixing time $\tau \in \text{poly}(\log|\mathcal{V}|)$ (that is, polynomial in $\log|\mathcal{V}|$).

A remedy to these considerations is the use of hybrid algorithms which combine quantum walk and Markov chain techniques. In for instance [15, 39] it is proposed to use a quantum walk for some t' steps, then measure the output, then again run the quantum walk for t' steps, etc. This is very much like the amplification procedure that we used in the previous thesis part. It is possible that combining our QFF scheme with such ideas can lead to new results on QW mixing. Indeed, QFF allows to simulate and accelerate a Markov chain for an intermediate number of steps (not simply its limit behavior) and could therefore serve as a subroutine of a quantum-classical mixing scheme. For example, mixing on very clustered graphs such as small world networks could be accelerated by using QFF merely to escape the clusters quadratically faster, rather than directly using it to mix over the entire graph.

Hamiltonian QFF

The Watrous and Ambainis-Szegedy schemes associate a quantum walk to the symmetric discriminant matrix of a Markov chain. Following for instance Childs [118], we can easily extend both schemes to associate a quantum walk to a general Hermitian matrix $H = H^\dagger$. Thereto we adapt the coin toss (7.1) in Watrous' construction so that it maps

$$|i,\flat\rangle \mapsto V|i,\flat\rangle = \sum_j \sqrt{H(j,i)}|i,j\rangle + \sqrt{1-\sum_j |H(j,i)|}|j,\sharp\rangle,$$

where $\sharp$ denotes an additional "garbage" register in which we put the bad part[2]. In analogy with Propositions 14 and 15, we define the quantum walk $W = R_\flat V^\dagger S V$, and we immediately find that

$$\Pi_\flat W|v,\flat\rangle = H|v,\flat\rangle \text{ and } \Pi_\flat W^t|v,\flat\rangle = T_t(H)|v,\flat\rangle.$$

This generalization has been used for a variety of purposes, the main application being the discrete simulation of continuous time quantum walks [118, 151] and Hamiltonian dynamics [111, 113], where H represents the Hamiltonian of a quantum system. Other applications are the evaluation of boolean formulas [117, 151], and the construction of quantum Gibbs states and quantum SDP solvers [115, 152].

Our QFF algorithm directly carries over to this generalized setting, allowing for instance to create an approximation of the state $|H^t v\rangle$ using $\widetilde{O}(\sqrt{t}/\||H^t|v\rangle\|^2)$ QW steps and queries to the matrix H [153]. We mention two possible applications of our scheme:

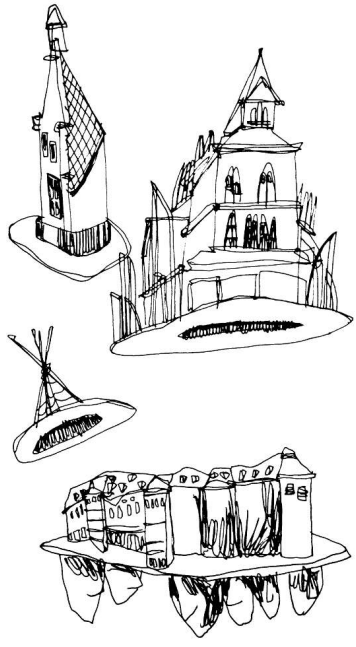
- Different quantum algorithms rely on the implementation of a function of a Hamiltonian $f(H)$ [119, 152]. In for instance [115] this problem is tackled in three steps: (i) the Taylor approximation of f is truncated, (ii) an involved

[2] We assume that H is rescaled such that $\sum_j |H(j,i)| \leq 1$ for all j. We are also neglecting the sign problem when taking the square root $\sqrt{H^\dagger(j,i)}$. See [111] for a discussion and an easy solution.

scheme approximates the truncated Taylor series by a Fourier series, (iii) Hamiltonian simulation is used to implement each of the Fourier terms. Since our QFF scheme allows to directly implement terms of the form H^k, we can use it to bypass steps (ii) and (iii) and directly implement the truncated Taylor series. Furthermore, applying their method to implement the function $f(H) = H^t$ requires $\Theta(t)$ QW steps, whereas QFF allows to do so in $O(\sqrt{t})$, so that QFF will also allow to accelerate their scheme for certain functions.

- Hamiltonians can be used to encode optimization problems, with their ground state corresponding to an optimal solution. Evidently, and this is where these ideas come from, this is the case in physical quantum systems such as spin systems, where the Hamiltonian represents the physical Hamiltonian of the system. However, such ideas have also proven very fruitful in an algorithmic context, where problems are artificially encoded in a Hamiltonian. See for instance the fields of adiabatic quantum computation [154] and Hamiltonian complexity theory [60, 155]. A common way of accessing or gaining information on the ground state is by implementing the operator e^{-Ht}, a technique that is called *imaginary time evolution* (see e.g. [156, 157] in the context of matrix product states, or [115] for quantum Gibbs sampling). Since e^{-H} is also a Hermitian operator, we can use QFF to quadratically fast-forward the implementation of e^{-Ht}. Moreover, we can not only approximate the limit case $\lim_{t\to\infty} e^{-Ht}$, as for instance quantum phase estimation techniques might also allow, but we can also access the intermediate dynamics, allowing for the study of frustrated states and local minima. This is similar to the way that the intermediate random walk dynamics in the clusterability tester of Section 9.3 reveal information about the local clusters.

Back Matter



Appendix A
Quantum State Generation in $O(\sqrt{\mathrm{HT}})$

In this appendix we present a new result that is relevant to this thesis, yet does not explicitly require the new techniques introduced in the thesis. We do make use of the QFF scheme from Chapter 8, but the algorithm can be analyzed making use only of the limit behavior of the quantum simulation, which can also be achieved by existing techniques such as quantum phase estimation.

The result that we will present is a new quantum walk algorithm for quantum state generation from graphs. Informally stated, the result comes down to the following theorem, where we let HT_j denote the hitting time of element j.

Theorem 13. *Assume that we are given an element $j \in \mathcal{V}$ and a reversible Markov chain P with stationary distribution π. Then we can create a quantum state ϵ-close to $|\sqrt{\pi}\rangle$ in an expected number of QW steps in*

$$O\left(\sqrt{\frac{HT_j}{\epsilon}} \log \frac{1}{\epsilon}\right).$$

The algorithm builds on insights and techniques from Krovi et al [22], proposed in the context of quantum search.

Most relevant to this thesis is the fact that the algorithm also leads to a quantum speedup for mixing on certain graphs: evidently, if we can create $|\sqrt{\pi}\rangle$, then we can measure the node basis and obtain a classical sample. Our algorithm provides a quadratic speedup for all graphs for which the hitting time and classical mixing time are similar. Examples of such are the dumbbell graph and complete binary trees.

A.1 Folklore: QSG in $\widetilde{O}\left(1/\sqrt{\delta\pi(j)}\right)$

To better place our result into context, and give some feeling on the topic, we will present the folklore result that quantum state generation, starting from node j, is possible in $\widetilde{O}(1/\sqrt{\delta\pi(j)})$ QW steps, with δ the spectral gap of the Markov chain. See e.g. [20, 39, 108, 149] for some discussion. In line with our framework, we can achieve this bound by building on the below lemma, which shows that the state $|D^t j\rangle$ for sufficiently large t will approximate $|\sqrt{\pi}\rangle$:

Lemma 18. *If* $t \geq \frac{1}{\delta} \ln \frac{2}{\epsilon\sqrt{\pi(j)}}$, *then*

$$\left\| |D^t j\rangle - |\sqrt{\pi}\rangle \right\| \leq \epsilon.$$

Proof. By the reversibility of P we know that $D = \mathrm{diag}(\sqrt{\pi})^{-1} P \,\mathrm{diag}(\sqrt{\pi})$ and so P and D share the same spectrum. Therefore also D has a unique eigenvector with eigenvalue, which we can show is $|\sqrt{\pi}\rangle$:

$$D|\sqrt{\pi}\rangle = \mathrm{diag}(\sqrt{\pi})^{-1} P\pi = |\sqrt{\pi}\rangle.$$

Moreover, the spectral gap of D equals the spectral gap δ of P. Now let $\{(\lambda_k, |v_k\rangle), 1 \leq k \leq |\mathcal{V}|\}$ be a complete orthonormal set of eigenpairs of D, where we set $|v_0\rangle = |\sqrt{\pi}\rangle$. Then we can expand

$$|j\rangle = \sum \langle v_k|j\rangle |v_k\rangle = \sqrt{\pi(j)}|\sqrt{\pi}\rangle + \sum_{k>0} \langle v_k|j\rangle |v_k\rangle.$$

Since D has a spectral gap δ, we know that $|\lambda_k| \leq 1 - \delta$ and so we can bound

$$\begin{aligned} \left\| D^t|j\rangle - \sqrt{\pi(j)}|\sqrt{\pi}\rangle \right\| &= \left\| \sum_{k>0} \lambda_k^t \langle v_k|j\rangle |v_k\rangle \right\| \\ &\leq (1-\delta)^t \leq e^{-\delta t} \leq \epsilon', \end{aligned}$$

if $t \geq \frac{1}{\delta} \ln \frac{1}{\epsilon'}$. Using the inequality in (7.6), Example 12, this implies that $\left\| |D^t j\rangle - |\sqrt{\pi}\rangle \right\| \leq 2\epsilon'/\sqrt{\pi(j)}$. Choosing $\epsilon' = \sqrt{\pi(j)}\epsilon/2$ finishes the proof. □

Using this lemma we can use our QFF scheme to create $|\sqrt{\pi}\rangle$. This extends the scheme discussed in Example 15 to non-symmetric reversible Markov chains, having non-uniform superposition $|\sqrt{\pi}\rangle$.

Proposition 20. *Let P be an ergodic reversible Markov chain with stationary distribution π and spectral gap δ, and let j be a node. Running* $\mathbf{QFFg}(|j\rangle, P, t, \epsilon')$ *for $\epsilon' = \epsilon/2$ and $t = \frac{1}{\delta} \ln \frac{2}{\epsilon'\sqrt{\pi(j)}}$ returns a state ϵ-close to $|\sqrt{\pi}\rangle$. The scheme has success probability at least $1/2$, and requires $O(1/\sqrt{\pi(j)})$ reflections around $|j\rangle$ and a number of QW steps in*

$$O\left(\frac{1}{\sqrt{\delta\pi(j)}} \log \frac{1}{\epsilon\sqrt{\pi(j)}} \right).$$

Proof. The QFF scheme returns a quantum state $|\psi\rangle$ which is ϵ'-close to $D^t|j\rangle / \|D^t|j\rangle\|$. By the above lemma, and our choice of t, this implies that

$$\| |\psi\rangle - |\sqrt{\pi}\rangle \| \leq \left\| |\psi\rangle - \frac{D^t|j\rangle}{\|D^t|j\rangle\|} \right\| + \left\| \frac{D^t|j\rangle}{\|D^t|j\rangle\|} - |\sqrt{\pi}\rangle \right\| \leq \epsilon.$$

By Theorem 6 the scheme requires a number of QW steps in

$$O\left(\frac{\sqrt{t}}{\|D^t|j\rangle\|} \sqrt{\log\left(\frac{1}{\epsilon \|D^t|j\rangle\|} \right)} \right) \in O\left(\frac{1}{\sqrt{\delta}\langle\sqrt{\pi}|j\rangle} \log \frac{1}{\epsilon\langle\sqrt{\pi}|j\rangle} \right). \qquad \square$$

A.2 Novelty: QSG in $\widetilde{O}(1/\sqrt{\text{HT}_j})$

Our contribution is to improve this scheme by using the idea of an *interpolated Markov chain*, as introduced in the paper by Krovi et al [22] and formerly discussed in Section 5.3. We recall that an interpolated Markov chain P_s, with respect to an original Markov chain P and a node j, follows from adding a self loop with some strength $1 > s > 0$ to the node j:

$$P_s(j,j) = s + (1-s)P(j,j), \quad P_s(k \neq j, j) = (1-s)P(k,j),$$

and $P_s(k,l) = P(k,l)$ elsewhere. If P is ergodic and reversible then also P_s will be ergodic and reversible. A crucial feature of these Markov chains is that the stationary distribution π_s of P_s also interpolates between π for $s = 0$ and e_j for $s = 1$. Most relevant to us is the fact that if we choose $s = 1 - \pi(j)/(1-\pi(j))$, then

$$\pi_s(j) = \frac{1}{2}, \qquad \pi_s(k \neq j) = \frac{1}{2}\pi^c(k),$$

with $\pi^c c(j) = 0$, and $\pi_c(k) = \pi(k)/(1-\pi(j))$ the stationary distribution restricted to the complement of j. This is proven in [22, Proposition 4]. It follows that the initial node j has a large overlap with π_s. The key now is that we can apply the following two-stage routine, building on the fact that $|D_s^t j\rangle \overset{t\to\infty}{\to} |\sqrt{\pi_s}\rangle$ and $|D^t\psi\rangle \overset{t\to\infty}{\to} |\sqrt{\pi}\rangle$:

1. Apply $\mathbf{QFF}(|j\rangle, P_s, t_1, \epsilon)$ for t_1 such that $|D_s^{t_1} j\rangle$ is close to $|\sqrt{\pi_s}\rangle$. Since $\langle\sqrt{\pi_s}|j\rangle = 1/\sqrt{2}$, the scheme has a constant success probability $\|D_s^{t_1}|j\rangle\| \geq |\langle\sqrt{\pi_s}|j\rangle|^2 = 1/2$.
2. Apply $\mathbf{QFF}(|\psi\rangle, P, t, \epsilon)$, on the output state $|\psi\rangle$, for t_2 such that $|D^{t_2}\psi\rangle \approx |D^{t_2}\sqrt{\pi_s}\rangle$ is close to $|\sqrt{\pi}\rangle$. Since

$$\langle\sqrt{\pi}|\sqrt{\pi_s}\rangle = \sqrt{\frac{\pi(j)}{2}} + \sqrt{\frac{1-\pi(j)}{2}} \geq \frac{1}{2},$$

the scheme has a constant success probability $\|D^{t_2}|\psi\rangle\| \geq |\langle\sqrt{\pi}|\psi\rangle|^2$, which is approximately $|\langle\sqrt{\pi}|\sqrt{\pi_s}\rangle|^2 \geq 1/2$.

We detail this scheme in Algorithm A.2 and Theorem 14. The total amount of QW steps is $\widetilde{O}(\sqrt{t_1} + \sqrt{t_2})$, which we can bound using the below lemma. We define HT_j as the expected hitting time of node j, starting from the stationary distribution on its complement π^c.

Lemma 19. *If $s = 1 - \pi(j)/(1-\pi(j))$ and $t \geq HT_j/(\sqrt{2}\epsilon)$ then*

$$\||D^t\sqrt{\pi_s}\rangle - |\sqrt{\pi}\rangle\|^2, \ \||D_s^t j\rangle - |\sqrt{\pi_s}\rangle\|^2 \leq \epsilon.$$

Proof. From (7.6) in Example 12 we know that for any two nonzero vectors it holds that $\left\|v/\|v\| - w/\|w\|\right\| \leq 2\|v-w\|/\|w\|$, so that we can bound

$$\begin{aligned}\||D_s^t j\rangle - |\sqrt{\pi_s}\rangle\| &\leq \frac{2}{|\langle\sqrt{\pi_s}|j\rangle|}\|D_s^t|j\rangle - \langle\sqrt{\pi_s}|j\rangle|\sqrt{\pi_s}\rangle\| \\ &= 2\sqrt{2}\|D_s^t|j\rangle - \langle\sqrt{\pi_s}|j\rangle|\sqrt{\pi_s}\rangle\|.\end{aligned}$$

Similarly, $\||D^t\sqrt{\pi_s}\rangle - |\sqrt{\pi}\rangle\|^2 \leq 2\sqrt{2}\|D^t|\sqrt{\pi_s}\rangle - \langle\sqrt{\pi}|\sqrt{\pi_s}\rangle|\sqrt{\pi}\rangle\|$. Therefore it will suffice to bound the right hand sides, for which we can use techniques very similar to [22]. Let $\{(\lambda_k(s), |v_k(s)\rangle), 1 \leq k \leq |\mathcal{V}|\}$ be a complete orthonormal set of eigenpairs of D_s, where we set $|v_0\rangle = |\sqrt{\pi_s}\rangle$. This allows us to expand

$$\|D_s^t|j\rangle - \langle\sqrt{\pi_s}|j\rangle|\pi_s\rangle\|^2 = \sum_{k>0} |\langle j|v_k(s)\rangle|^2 \lambda_k^{2t}(s).$$

We can bound

$$\lambda_k^{2t}(s) \leq \frac{1}{2t}\sum_{l=0}^{2t-1} \lambda_k^l(s) \leq \frac{1}{2t}\frac{1}{1-\lambda_k(s)},$$

so that $\|D_s^t|j\rangle - \langle\sqrt{\pi_s}|j\rangle|\sqrt{\pi_s}\rangle\|^2 \leq (2t)^{-1}\sum_{k>0}|\langle v_k(s)|j\rangle|^2/(1-\lambda_k(s))$. Using the fact that $|\sqrt{\pi_s}\rangle = |j\rangle/\sqrt{2} + |\sqrt{\pi^c}\rangle/\sqrt{(2)}$, and $\langle v_k(s)|\sqrt{\pi_s}\rangle = 0$ for $k > 0$ by the orthogonality of the basis, we can rewrite the sum

$$\sum_{k>0}\frac{|\langle v_k(s)|j\rangle|^2}{1-\lambda_k(s)} = \sum_{k>0}\frac{|\langle v_k(s)|\sqrt{\pi^c}\rangle|^2}{1-\lambda_k(s)}.$$

It is proven in [22] that the quantity $\sum_{k>0}|\langle v_k(s)|\sqrt{\pi^c}\rangle|^2)/(1-\lambda_k(s)) = \mathrm{HT}_j/2$, so that we indeed find that $\|D_s^t|j\rangle - \langle\sqrt{\pi_s}|j\rangle|\sqrt{\pi_s}\rangle\|^2 \leq \mathrm{HT}_j/(4t)$. We can prove the bound on $\|D^t|\sqrt{\pi_s}\rangle - \langle\sqrt{\pi}|\sqrt{\pi_s}\rangle|\pi\rangle\|$ in a very similar way:

$$\|D^t|\sqrt{\pi_s}\rangle - \langle\sqrt{\pi}|\sqrt{\pi_s}\rangle|\pi\rangle\| \leq \frac{1}{2t}\sum_{k>0}\frac{|\langle v_k|\sqrt{\pi_s}\rangle|^2}{1-\lambda_k}.$$

We can bound the sum by rewriting $|\sqrt{\pi_s}\rangle = \frac{1}{\sqrt{2\pi(j)}}|\sqrt{\pi}\rangle + \big(1/\sqrt{1-s}-1\big)|\sqrt{\pi_c}\rangle/\sqrt{2}$, and using the fact that $\langle v_k|\sqrt{\pi}\rangle = 0$ for all $k > 0$, so that

$$\sum_{k>0}\frac{|\langle v_k|\sqrt{\pi_s}\rangle|^2}{1-\lambda_k} = \frac{1}{2}\Big(\frac{1}{\sqrt{1-s}}-1\Big)^2\sum_{k>0}\frac{|\langle v_k|\sqrt{\pi^c}\rangle|^2}{1-\lambda_k}.$$

Again from [22] we know that $\sum_{k>0}|\langle v_k|\sqrt{\pi^c}\rangle|^2/(1-\lambda_k) = \pi(j)\mathrm{HT}_j$. By our choice of $s = 1-\pi(j)/(1-\pi(j))$ we see that

$$\sum_{k>0}\frac{|\langle v_k|\sqrt{\pi_s}\rangle|^2}{1-\lambda_k} \leq \frac{1}{2}\Big(\frac{1}{\sqrt{1-s}}-1\Big)^2\pi(j)\mathrm{HT}_j \leq \frac{\pi(j)}{2(1-s)}\mathrm{HT}_j.$$

Since $\pi(j)/(2(1-s)) = (1-\pi(j))/2$ we finally find that $\|D^t|\sqrt{\pi_s}\rangle - \langle\sqrt{\pi}|\sqrt{\pi_s}\rangle|\pi\rangle\| \leq (1-\pi(j))\mathrm{HT}_j/(4t) \leq \mathrm{HT}_j/(4t)$. □

These bounds show that we can choose $t_1, t_2 \in O(\mathrm{HT}_j)$ in the QFF scheme, leading to the below proposition.

Theorem 14. *The QSG algorithm* $\mathbf{QSG}(|j\rangle, P, \epsilon, \pi(j), HT_j)$ *outputs a state* ϵ*-close to* $|\sqrt{\pi}\rangle$ *with success probability at least* $(1-\epsilon)/(2\sqrt{2})$*. Otherwise, it outputs "*`Fail`*". The algorithm requires a number of QW steps in*

$$O\Big(\sqrt{\frac{HT_j}{\epsilon}}\log\frac{1}{\epsilon}\Big).$$

Algorithm 9 Quantum State Generation $\mathbf{QSG}(|j\rangle, P, \epsilon, \pi(j), \mathrm{HT}_j)$

Input:
node state $|j\rangle$, Markov chain P, $\epsilon > 0$, parameters $\pi(j)$ and HT_j
Do:
1: set $\epsilon_1 = \epsilon/16$, $\epsilon_2 = \epsilon/4$, $t_1 = \left\lceil \frac{\mathrm{HT}_j}{\sqrt{2}\epsilon_1} \right\rceil$, and $t_2 = \left\lceil \frac{\mathrm{HT}_j}{\sqrt{2}\epsilon_2} \right\rceil$
2: set $s = 1 - \frac{\pi(j)}{1-\pi(j)}$
3: run $\mathbf{QFF}(|j\rangle, P_s, t_1, \epsilon_1)$
If successful, call the output state $|\psi\rangle$. Otherwise output "`Fail`" and stop
4: run $\mathbf{QFF}(|\psi\rangle, P, t_2, \epsilon_2)$
If unsuccessful, output "`Fail`" and stop

Complexity: $\left\lceil \sqrt{2t_1 \ln \frac{2}{\epsilon_1}} \right\rceil + \left\lceil \sqrt{2t_2 \ln \frac{2}{\epsilon_2}} \right\rceil$ QW steps **Success Prob.:** $\geq (1-\epsilon)\frac{1}{2\sqrt{2}}$

Proof. For clarity we will analyze the scheme for parameters $(\epsilon_1, \epsilon_2, \delta_1, \delta_2)$, with

$$t_1 = \left\lceil \frac{\mathrm{HT}_j}{\sqrt{2}\delta_1} \right\rceil, \quad t_2 = \left\lceil \frac{\mathrm{HT}_j}{\sqrt{2}\delta_2} \right\rceil.$$

We will later fix the parameters. The two QFF stages then become

$$\mathbf{QFF}(|j\rangle, P_s, t_1, \epsilon_1, \pi(j), \mathrm{HT}_j) \quad \text{and} \quad \mathbf{QFF}(|\psi\rangle, P_s, t_2, \epsilon_2, \pi(j), \mathrm{HT}_j).$$

We can use Theorem 5 to describe both stages:

1. $\mathbf{QFF}(|j\rangle, P_s, t_1, \epsilon_1, \pi(j), \mathrm{HT}_j)$: outputs $|\psi_1\rangle$ such that

$$\||\psi_1\rangle - |D_s^{t_1} j\rangle\| \leq \epsilon_1,$$

with success probability $p_1 \geq (1-\epsilon_1)\|D_s^{t_1}|j\rangle\|$. By our choice of t_1 we know from Lemma 19 that

$$\||D_s^{t_1} j\rangle - |\sqrt{\pi_s}\rangle\| \leq \delta_1.$$

2. $\mathbf{QFF}(|\psi\rangle, P, t_2, \epsilon_2, \pi(j), \mathrm{HT}_j)$: outputs $|\psi_2\rangle$ such that

$$\||\psi_2\rangle - |D^{t_2}\psi_1\rangle\| \leq \epsilon_2,$$

with success probability $p_2 \geq (1-\epsilon_2)\|D^{t_2}|\psi_1\rangle\|$. By our choice of t_2 we know from Lemma 19 that

$$\||D^{t_2}\sqrt{\pi_s}\rangle - |\sqrt{\pi}\rangle\| \leq \delta_2.$$

We can bound the success probabilities by noting that both $\|D_s^{t_1}|j\rangle\|$ and $\|D^{t_2}|\psi_1\rangle\|$ are nonincreasing functions of t, with

$$\lim_{t\to\infty} \|D_s^{t_1}|j\rangle\| = |\langle\sqrt{\pi_s}|j\rangle| \quad \text{and} \quad \lim_{t\to\infty} \|D^{t_2}|\psi_1\rangle\| = |\langle\sqrt{\pi}|\psi_1\rangle|.$$

As a consequence, $p_1 \geq (1-\epsilon_1)|\langle\sqrt{\pi_s}|j\rangle| = (1-\epsilon_1)/\sqrt{2}$ and $p_2 \geq (1-\epsilon_2)|\langle\sqrt{\pi}|\psi_1\rangle|$. To bound $|\langle\sqrt{\pi}|\psi_1\rangle|$ we use that

$$\begin{aligned} &\||\psi_1\rangle - |\sqrt{\pi_s}\rangle\| \\ &\quad \leq \||\psi_1\rangle - |D_s^{t_1} j\rangle\| + \||D_s^{t_1} j\rangle - |\sqrt{\pi_s}\rangle\| \leq \epsilon_1 + \delta_1, \end{aligned} \tag{A.1}$$

and therefore $|\langle\sqrt{\pi}|\psi_1\rangle - \langle\sqrt{\pi}|\sqrt{\pi_s}\rangle| \leq \epsilon_1 + \delta_1$. Using the reverse triangle inequality this proves that $|\langle\sqrt{\pi}|\psi_1\rangle| \geq |\langle\sqrt{\pi}|\sqrt{\pi_s}\rangle| - \epsilon_1 - \delta_1$. Explicit calculation shows that $|\langle\sqrt{\pi}|\sqrt{\pi_s}\rangle| = \sqrt{\pi(j)/2} + \sqrt{(1-\pi(j))/2} \geq 1/2$, so that finally

$$p_1 \geq (1-\epsilon_1)\frac{1}{\sqrt{2}}, \qquad p_2 \geq (1-\epsilon_2)\Big(\frac{1}{2} - \epsilon_1 - \delta_1\Big).$$

Now we will bound $\||\sqrt{\pi}\rangle - |\psi_2\rangle\|$. Using the triangle inequality we directly see that

$$\begin{aligned}
&\||\sqrt{\pi}\rangle - |\psi_2\rangle\| \\
&\leq \||\sqrt{\pi}\rangle - |D^{t_2}\sqrt{\pi_s}\rangle\| + \||D^{t_2}\sqrt{\pi_s}\rangle - |D^{t_2}\psi_1\rangle\| + \||D^{t_2}\psi_1\rangle - |\psi_2\rangle\| \\
&\leq \epsilon_2 + \delta_2 + \||D^{t_2}\sqrt{\pi_s}\rangle - |D^{t_2}\psi_1\rangle\|.
\end{aligned}$$

Now we can again use the fact that for any two nonzero vectors it holds that $\big\|v/\|v\| - w/\|w\|\big\| \leq 2\|v-w\|/\|w\|$, so that

$$\begin{aligned}
\||D^{t_2}\sqrt{\pi_s}\rangle - |D^{t_2}\psi_1\rangle\| &\leq \frac{2}{\|D^{t_2}|\sqrt{\pi_s}\rangle\|}\|D^{t_2}(|\sqrt{\pi_s}\rangle - |\psi_1\rangle)\| \\
&\leq \frac{2}{|\langle\sqrt{\pi_s}|\sqrt{\pi}\rangle|}\||\sqrt{\pi_s}\rangle - |\psi_1\rangle\|.
\end{aligned}$$

By (A.1) and the fact that $|\langle\sqrt{\pi}|\sqrt{\pi_s}\rangle| \geq \frac{1}{2}$ this shows that $\||D^{t_2}\sqrt{\pi_s}\rangle - |D^{t_2}\psi_1\rangle\| \leq 4(\epsilon_1+\delta_1)$. Combined with the above inequality this gives

$$\||\sqrt{\pi}\rangle - |\psi_2\rangle\| \leq 4(\epsilon_1+\delta_1) + \epsilon_2 + \delta_2.$$

So finally we can choose $\epsilon_1 = \delta_1 = \epsilon/16$ and $\epsilon_2 = \delta_2 = \epsilon/4$ to yield $\||\sqrt{\pi}\rangle - |\psi_2\rangle\| \leq \epsilon$. The total success probability $p_1 p_2$ is then bounded by

$$\begin{aligned}
p_1 p_2 &\geq \frac{\sqrt{3}}{2}(1-\epsilon_1)(1-\epsilon_2)\Big(\frac{1}{2} - \epsilon_1 - \delta_1\Big) \\
&= \frac{1}{2\sqrt{2}}\Big(1 - \frac{\epsilon}{16}\Big)\Big(1 - \frac{\epsilon}{4}\Big)^2 \geq \frac{1}{2\sqrt{2}}(1-\epsilon).
\end{aligned}$$

□

A.3 Discussion

We proposed a quantum walk algorithm for quantum state generation, building on the work of Krovi et al [22]. Starting from a node j, the algorithm creates an ϵ-approximation of the goal state $|\sqrt{\pi}\rangle$ in $O\big(\sqrt{\mathrm{HT}_j/\epsilon}\log\epsilon^{-1}\big)$ expected QW steps. It is a well known result, see e.g. [63, Lemma 3.17], that $\mathrm{HT}_j \leq 1/(\delta\pi(j))$, so that the algorithm improves on the folklore bound. Indeed, in many cases HT_j is significantly lower than $1/(\delta\pi(j))$, some examples of which we will discuss below.

A nice application of the algorithm is for quantum search. Krovi et al [22] have shown that quantum walks allow to hit a marked element k in $O(\sqrt{\mathrm{HT}_k})$ QW steps, quadratically faster than the classical Markov chain. A caveat to their algorithm, and in fact to most quantum search algorithms [3, 21, 106], is that it requires

to start from the state $|\sqrt{\pi}\rangle$, rather than starting from a single node. A notable exception is the work by Belovs [144, 158], who constructed an algorithm that allows to *detect* the presence of a marked element k, starting from an element j, in $O(\sqrt{\mathrm{HT}(j,k) + \mathrm{HT}(k,j)})$ QW steps, with $\mathrm{HT}(k,j)$ the hitting time of k starting in element j. To the best of our knowledge, this is the only result on quantum search when starting from a single element. We provide a new result: starting from an element j, we can find an element k in a number of expected QW steps in

$$O\big(\sqrt{\mathrm{HT}_j} + \sqrt{\mathrm{HT}_k}\big).$$

The total cost of the algorithm is

$$O\Big(\big(\sqrt{\mathrm{HT}_j} + \sqrt{\mathrm{HT}_k}\big)(\mathtt{U} + \mathtt{C})\Big),$$

where $\mathtt{U}$ is the update cost of implementing a step of the quantum walk, and $\mathtt{C}$ is the checking cost (see Section 10.1 for definitions). The algorithm in [22] for finding a marked element k is $O(\mathtt{S} + \sqrt{\mathrm{HT}_k}(\mathtt{U} + \mathtt{C}))$, with $\mathtt{S}$ the setup cost of creating the superposition $|\sqrt{\pi}\rangle$. It seems reasonable that our algorithm is not optimal for this task, and that along the line of Belovs' work a quantum algorithm should exist which finds an element k, starting from j, in $\Theta(\sqrt{\mathrm{HT}(j,k) + \mathrm{HT}(k,j)})$ expected QW steps.

The algorithm also has relevance for the task of accelerating classical mixing using quantum walks, the main topic of the first thesis part. Here we wish to return an element of $\mathcal{V}$ distributed according to π, which is clearly easier than creating $|\sqrt{\pi}\rangle$. A classical Markov chain approximately performs this task in $\tau \in \Omega\big(\delta^{-1}\big)$ Markov chain steps, so that we find a quadratic speedup if $\mathrm{HT}_j \in O(\delta^{-1})$. This is the case for our running example, the dumbbell graph $K_N - K_N$, where $\mathrm{HT}, \delta^{-1} \in \Theta(1/N^2)$ [63]. Our algorithm allows to approximately create the superposition $|\sqrt{\pi}\rangle$ in $O(N)$ QW steps. Measuring this state returns a classical sample quadratically faster than a random walk. Note that in this case the folklore bound $(\delta\pi(j))^{-1/2} \in O(N^{3/2})$.

Another relevant example is the complete binary tree $T_{2,k}$ of depth k, having $N = 2^{k+1} - 1$ nodes and spectral gap $\delta \in \Theta(1/N)$. If j is a leaf, then $\mathrm{HT}_j \in \Theta(N \log N)$ [63] and so we approximately create the superposition $|\sqrt{\pi}\rangle$ in $O(\sqrt{N \log N})$ QW steps. If j is the root node, then $\mathrm{HT}_j \in \Theta(N)$ [63], so that our algorithm only requires $O(\sqrt{N})$ QW steps. The folklore bound in this case is $(\delta\pi(j))^{-1/2} \in O(N)$, which we quadratically improve. This case is also interesting with respect to the results from the first thesis part, where we showed how any invariant quantum walk necessarily requires $\Omega(1/\Phi_{G,\pi})$ steps. For the binary tree this gives a lower bound of $\Omega(N)$ QW steps, as we discussed in Example 11. This is clearly broken by our algorithm. Since the algorithm is nevertheless local, building on local quantum walks, this implies that it is not invariant, as defined in the first part of this thesis. This suggests that the algorithm should be interpreted as a preparation scheme rather than a mixing scheme, missing the stabilization properties of Markov chain and quantum walk mixing schemes as considered in [39, 51, 53].

Publications

1. Simon Apers and Alain Sarlette. Accelerating Consensus by Spectral Clustering and Polynomial Filters. *IEEE Transactions on Control of Network Systems*, 4(3):544–554, IEEE, 2017.

2. Simon Apers, Francesco Ticozzi, and Alain Sarlette. Lifting Markov Chains to Mix Faster: Limits and opportunities. Submitted to *IEEE Transactions on Information Theory*, arXiv:1705.08253, 2017.

3. Simon Apers, Alain Sarlette, and Francesco Ticozzi. When Does Memory Speed-up Mixing? In *Proceedings of the 56th IEEE Conference on Decision and Control*, 4940-4945, IEEE, 2017.

4. Wei-Wei Zhang, Barry C Sanders, Simon Apers, Sandeep K Goyal, and David L Feder. Detecting Topological Transitions in Two Dimensions by Hamiltonian Evolution. *Physical Review Letters*, 119(19):197401, 2017.

5. Simon Apers, Alain Sarlette, and Francesco Ticozzi. Bounding the Convergence Time of Local Probabilistic Evolution. In *Proceedings of the 2nd International Conference on Geometric Science of Information*, 754-762, Springer, 2017.

6. Simon Apers, Alain Sarlette, and Francesco Ticozzi. Simulation of Quantum Walks and Fast Mixing with Classical Processes. *Physical Review A*, 98(3):032115, 2018.

7. Simon Apers and Alain Sarlette. Quantum Fast-Forwarding: Markov Chains and Graph Property Testing. Submitted to *Quantum Information and Computation*, arXiv:1804.02321, 2018.

References

[1] John Watrous. Quantum simulations of classical random walks and undirected graph connectivity. *Journal of Computer and System Sciences*, 62(2): 376–391, 2001.

[2] Andris Ambainis. Quantum walks and their algorithmic applications. *International Journal of Quantum Information*, 1(04):507–518, 2003.

[3] Mario Szegedy. Quantum speed-up of Markov chain based algorithms. In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, pages 32–41. IEEE, 2004.

[4] Andris Ambainis. Quantum walk algorithm for element distinctness. *SIAM Journal on Computing*, 37(1):210–239, 2007.

[5] David A Levin and Yuval Peres. *Markov chains and mixing times*. American Mathematical Society, 2017.

[6] Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953.

[7] Balázs Gerencsér. Markov chain mixing time on cycles. *Stochastic Processes and their Applications*, 121(11):2553–2570, 2011.

[8] Persi Diaconis, Susan Holmes, and Radford M Neal. Analysis of a nonreversible Markov chain sampler. *Annals of Applied Probability*, 10(3): 726–752, 2000.

[9] Fang Chen, László Lovász, and Igor Pak. Lifting Markov chains to speed up mixing. In *Proceedings of the 31st Annual ACM Symposium on Theory of Computing*, pages 275–281. ACM, 1999.

[10] Persi Diaconis and Laurent Miclo. On the spectral analysis of second-order Markov chains. *Annales de la Faculté des Sciences de Toulouse. Mathématiques*, 22(3):573–621, 2013.

[11] Simon Apers, Francesco Ticozzi, and Alain Sarlette. Lifting Markov chains to mix faster: Limits and opportunities. *arXiv:1705.08253*, 2017.

[12] Yakir Aharonov, Luiz Davidovich, and Nicim Zagury. Quantum random walks. *Physical Review A*, 48(2):1687, 1993.

[13] Edward Farhi and Sam Gutmann. Quantum computation and decision trees. *Physical Review A*, 58(2):915, 1998.

[14] Andris Ambainis, Eric Bach, Ashwin Nayak, Ashvin Vishwanath, and John Watrous. One-dimensional quantum walks. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing*, pages 37–49. ACM, 2001.

[15] Dorit Aharonov, Andris Ambainis, Julia Kempe, and Umesh Vazirani. Quantum walks on graphs. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing*, pages 50–59. ACM, 2001.

[16] Andrew M Childs. Universal computation by quantum walk. *Physical Review Letters*, 102(18):180501, 2009.

[17] Neil B Lovett, Sally Cooper, Matthew Everitt, Matthew Trevers, and Viv Kendon. Universal quantum computation using the discrete-time quantum walk. *Physical Review A*, 81(4):042330, 2010.

[18] Frédéric Magniez, Miklos Santha, and Mario Szegedy. Quantum algorithms for the triangle problem. *SIAM Journal on Computing*, 37(2):413–424, 2007.

[19] Neil Shenvi, Julia Kempe, and K Birgitta Whaley. Quantum random-walk search algorithm. *Physical Review A*, 67(5):052307, 2003.

[20] Andrew M Childs and Jeffrey Goldstone. Spatial search by quantum walk. *Physical Review A*, 70(2):022314, 2004.

[21] Frédéric Magniez, Ashwin Nayak, Jérémie Roland, and Miklos Santha. Search via quantum walk. *SIAM Journal on Computing*, 40(1):142–164, 2011.

[22] Hari Krovi, Frédéric Magniez, Maris Ozols, and Jérémie Roland. Quantum walks can find a marked element on any graph. *Algorithmica*, 74(2):851–907, 2016.

[23] Julia Kempe. Quantum random walks: an introductory overview. *Contemporary Physics*, 44(4):307–327, 2003.

[24] Gregory S Engel, Tessa R Calhoun, Elizabeth L Read, Tae-Kyu Ahn, Tomáš Mančal, Yuan-Chung Cheng, Robert E Blankenship, and Graham R Fleming. Evidence for wavelike energy transfer through quantum coherence in photosynthetic systems. *Nature*, 446(7137):782–786, 2007.

[25] Masoud Mohseni, Patrick Rebentrost, Seth Lloyd, and Alan Aspuru-Guzik. Environment-assisted quantum walks in photosynthetic energy transfer. *The Journal of Chemical Physics*, 129(17):11B603, 2008.

[26] Alejandro Romanelli, Raul Donangelo, Renato Portugal, and Franklin de Lima Marquezino. Thermodynamics of n-dimensional quantum walks. *Physical Review A*, 90(2):022329, 2014.

[27] Takashi Oka, Norio Konno, Ryotaro Arita, and Hideo Aoki. Breakdown of an electric-field driven system: a mapping to a quantum walk. *Physical Review Letters*, 94(10):100602, 2005.

[28] Takuya Kitagawa, Matthew A Broome, Alessandro Fedrizzi, Mark S Rudner, Erez Berg, Ivan Kassal, Alán Aspuru-Guzik, Eugene Demler, and Andrew G White. Observation of topologically protected bound states in photonic quantum walks. *Nature Communications*, 3(882), 2012.

[29] Wei-Wei Zhang, Barry C Sanders, Simon Apers, Sandeep K Goyal, and David L Feder. Detecting topological transitions in two dimensions by hamiltonian evolution. *Physical Review Letters*, 119(19):197401, 2017.

[30] Michal Karski, Leonid Förster, Jai-Min Choi, Andreas Steffen, Wolfgang Alt, Dieter Meschede, and Artur Widera. Quantum walk in position space with single optically trapped atoms. *Science*, 325(5937):174–177, 2009.

[31] Alberto Peruzzo, Mirko Lobino, Jonathan CF Matthews, Nobuyuki Matsuda, Alberto Politi, Konstantinos Poulios, Xiao-Qi Zhou, Yoav Lahini, Nur Ismail, Kerstin Wörhoff, et al. Quantum walks of correlated photons. *Science*, 329(5998):1500–1503, 2010.

[32] Maximilian Genske, Wolfgang Alt, Andreas Steffen, Albert H Werner, Reinhard F Werner, Dieter Meschede, and Andrea Alberti. Electric quantum walks with individual atoms. *Physical Review Letters*, 110(19):190601, 2013.

[33] Philipp M Preiss, Ruichao Ma, M Eric Tai, Alexander Lukin, Matthew Rispoli, Philip Zupancic, Yoav Lahini, Rajibul Islam, and Markus Greiner. Strongly correlated quantum walks in optical lattices. *Science*, 347(6227): 1229–1233, 2015.

[34] Emmanuel Flurin, Vinay V Ramasesh, Shay Hacohen-Gourgy, Leigh S Martin, Norman Y Yao, and Irfan Siddiqi. Observing topological invariants using quantum walks in superconducting circuits. *Physical Review X*, 7(3): 031023, 2017.

[35] Andrew M Childs, Richard Cleve, Enrico Deotto, Edward Farhi, Sam Gutmann, and Daniel A Spielman. Exponential algorithmic speedup by a quantum walk. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing*, pages 59–68. ACM, 2003.

[36] Julia Kempe. Discrete quantum walks hit exponentially faster. *Probability Theory and Related Fields*, 133(2):215–235, 2005.

[37] Peter Høyer and Mojtaba Komeili. Efficient quantum walk on the grid with multiple marked elements. In *Proceedings of the 34th International Symposium on Theoretical Aspects of Computer Science*, pages 42:1–42:14. LIPICs, 2016.

[38] Cristopher Moore and Alexander Russell. Quantum walks on the hypercube. In *Proceedings of the 6th International Workshop on Randomization and Approximation Techniques in Computer Science*, pages 164–178. Springer, 2002.

[39] Peter C Richter. Quantum speedup of classical mixing processes. *Physical Review A*, 76(4):042306, 2007.

[40] Dorit Aharonov and Amnon Ta-Shma. Adiabatic quantum state generation and statistical zero knowledge. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing*, pages 20–29. ACM, 2003.

[41] Rolando D Somma, Sergio Boixo, Howard Barnum, and Emanuel Knill. Quantum simulations of classical annealing processes. *Physical Review Letters*, 101(13):130504, 2008.

[42] David Poulin and Pawel Wocjan. Sampling from the thermal quantum gibbs state and evaluating partition functions with a quantum computer. *Physical Review Letters*, 103(22):220502, 2009.

[43] Vedran Dunjko and Hans J Briegel. Sequential quantum mixing for slowly evolving sequences of Markov chains. *arXiv:1503.01334*, 2015.

[44] Simon Apers, Alain Sarlette, and Peter Høyer. Quantum sampling in square root of the search time. Poster at *Conference on Quantum Information Processing*, 2018.

[45] Peter C Richter. Almost uniform sampling via quantum walks. *New Journal of Physics*, 9(3):72, 2007.

[46] Michael A Nielsen and Isaac Chuang. *Quantum computation and quantum information*. Cambridge University Press, 2002.

[47] David A Meyer. From quantum cellular automata to quantum lattice gases. *Journal of Statistical Physics*, 85(5-6):551–574, 1996.

[48] Viv Kendon. Decoherence in quantum walks - a review. *Mathematical Structures in Computer Science*, 17(6):1169–1220, 2007.

[49] Franklin L Marquezino, Renato Portugal, and Gonzalo Abal. Mixing times in quantum walks on two-dimensional grids. *Physical Review A*, 82(4): 042341, 2010.

[50] Franklin L Marquezino, Renato Portugal, Gonzalo Abal, and Raul Donangelo. Mixing times in quantum walks on the hypercube. *Physical Review A*, 77(4):042312, 2008.

[51] Kristan Temme, Michael James Kastoryano, Mary Beth Ruskai, Michael Marc Wolf, and Frank Verstraete. The χ^2-divergence and mixing times of quantum Markov processes. *Journal of Mathematical Physics*, 51(12):122201, 2010.

[52] Terry Farrelly, Fernando GSL Brandão, and Marcus Cramer. Thermalization and return to equilibrium on finite quantum lattice systems. *Physical Review Letters*, 118(14):140601, 2017.

[53] Emilio Onorati, Oliver Buerschaper, Martin Kliesch, Winton Brown, Albert H Werner, and Jens Eisert. Mixing properties of stochastic quantum hamiltonians. *Communications in Mathematical Physics*, 355(3):905–947, 2017.

[54] Jeff Cheeger. A lower bound for the smallest eigenvalue of the laplacian. In *Proceedings of the Princeton conference in honor of Professor S. Bochner*, pages 195–199, 1969.

[55] Peter Buser. A note on the isoperimetric constant. In *Annales scientifiques de l'École Normale Supérieure*, volume 15, pages 213–230. Elsevier, 1982.

[56] David Aldous. On the Markov chain simulation method for uniform combinatorial distributions and simulated annealing. *Probability in the Engineering and Informational Sciences*, 1(1):33–46, 1987.

[57] Noga Alon. Eigenvalues and expanders. *Combinatorica*, 6(2):83–96, 1986.

[58] Scott Aaronson. Quantum computing and hidden variables. *Physical Review A*, 71(3):032325, 2005.

[59] Richard P Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6):467–488, 1982.

[60] Alexei Yu Kitaev, Alexander Shen, and Mikhail N Vyalyi. *Classical and quantum computation*, volume 47. American Mathematical Society, 2002.

[61] Mark Jerrum and Alistair Sinclair. Approximating the permanent. *SIAM journal on computing*, 18(6):1149–1178, 1989.

[62] Martin Dyer, Alan Frieze, and Ravi Kannan. A random polynomial-time algorithm for approximating the volume of convex bodies. *Journal of the ACM*, 38(1):1–17, 1991.

[63] David Aldous and Jim Fill. Reversible Markov chains and random walks on graphs. Unfinished monograph, 2002.

[64] Miroslav Fiedler. Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal*, 23(2):298–305, 1973.

[65] Jozef Dodziuk. Difference equations, isoperimetric inequality and transience of certain random walks. *Transactions of the American Mathematical Society*, 284(2):787–794, 1984.

[66] Gregory F Lawler and Alan D Sokal. Bounds on the l^2 spectrum for Markov chains and Markov processes: a generalization of cheeger's inequality. *Transactions of the American Mathematical Society*, 309(2):557–580, 1988.

[67] Milena Mihail. Conductance and convergence of Markov chains-a combinatorial treatment of expanders. In *Proceedings of the 30th Annual IEEE Symposium on Foundations of Computer Science*, pages 526–531. IEEE, 1989.

[68] Mark Jerrum and Alistair Sinclair. Conductance and the rapid mixing property for Markov chains: the approximation of permanent resolved. In *Proceedings of the 21th Annual ACM Symposium on Theory of Computing*, pages 235–244. ACM, 1988.

[69] Ravi Montenegro and Prasad Tetali. Mathematical aspects of mixing times in Markov chains. *Foundations and Trends in Theoretical Computer Science*, 1(3):237–354, 2006.

[70] Alistair Sinclair. *Algorithms for random generation and counting: a Markov chain approach*. Springer Science & Business Media, 2012.

[71] Stephen Jordan. Quantum algorithms zoo. Published electronically at `https://math.nist.gov/quantum/zoo/`, 2011.

[72] Simon Apers, Alain Sarlette, and Francesco Ticozzi. Simulation of quantum walks and fast mixing with classical processes. *Physical Review A*, 98(3): 032115, 2018.

[73] Scott Kirkpatrick, C Daniel Gelatt, and Mario P Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[74] Simon Apers and Alain Sarlette. Accelerating consensus by spectral clustering and polynomial filters. *IEEE Transactions on Control of Network Systems*, 4(3):544–554, 2017.

[75] Neal Madras and Gordon Slade. *The self-avoiding walk*. Springer Science & Business Media, 2013.

[76] László Lovász and Peter Winkler. Mixing times. *Microsurveys in discrete probability*, 41:85–134, 1998.

[77] Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. *Introduction to algorithms*. MIT, 2001.

[78] Lester R Ford and Delbert R Fulkerson. Maximal flow through a network. *Canadian Journal of Mathematics*, 8(3):399–404, 1956.

[79] Danial Dervovic. For every quantum walk there is a (classical) lifted Markov chain with the same mixing time. *arXiv:1712.02318*, 2017.

[80] Andrew M Childs, Edward Farhi, and Sam Gutmann. An example of the difference between quantum and classical random walks. *Quantum Information Processing*, 1(1-2):35–43, 2002.

[81] Laurent Saloff-Coste and Jessica Zúniga. Convergence of some time inhomogeneous Markov chains via spectral techniques. *Stochastic Processes and their Applications*, 117(8):961–979, 2007.

[82] Behrouz Touri and Angelia Nedić. Alternative characterization of ergodicity for doubly stochastic chains. In *Proceedings of the 50th Annual Conference on Decision and Control and European Control Conference*, pages 5371–5376. IEEE, 2011.

[83] Elchanan Mossel, Yuval Peres, and Alistair Sinclair. Shuffling by semirandom transpositions. In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, pages 572–581. IEEE, 2004.

[84] Persi Diaconis and Arun Ram. Analysis of systematic scan metropolis algorithms using iwahori-hecke algebra techniques. *Michigan Mathematical Journal*, 48(1), 2000.

[85] Gert De Cooman, Filip Hermans, and Erik Quaeghebeur. Imprecise Markov chains and their limit behavior. *Probability in the Engineering and Informational Sciences*, 23(4):597–635, 2009.

[86] Julien M Hendrickx, Raphaël M Jungers, Alexander Olshevsky, and Guillaume Vankeerberghen. Graph diameter, eigenvalues, and minimum-time consensus. *Automatica*, 50(2):635–640, 2014.

[87] Jacob Wolfowitz. Products of indecomposable, aperiodic, stochastic matrices. *Proceedings of the American Mathematical Society*, 14(5):733–737, 1963.

[88] Leslie Hogben. Spectral graph theory and the inverse eigenvalue problem of a graph. *Electronic Journal of Linear Algebra*, 14(1):3, 2005.

[89] Peter L Knight, Eugenio Roldán, and John E Sipe. Quantum walk on the line as an interference phenomenon. *Physical Review A*, 68(2):020301, 2003.

[90] Hyunseok Jeong, Mauro Paternostro, and Myungshik S Kim. Simulation of quantum random walks using the interference of a classical field. *Physical Review A*, 69(1):012310, 2004.

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

[92] Sandeep K Goyal, Filippus S Roux, Andrew Forbes, and Thomas Konrad. Implementation of multidimensional quantum walks using linear optics and classical light. *Physical Review A*, 92(4):040302, 2015.

[93] Kavita Ramanan and Aaron Smith. Bounds on lifting continuous-state Markov chains to speed up mixing. *Journal of Theoretical Probability*, 31 (3):1–32, 2017.

[94] Sidney C Port. *Theoretical probability for applications*, volume 206. Wiley-Interscience, 1994.

[95] Samis Trevezas and Nikolaos Limnios. Variance estimation in the central limit theorem for Markov chains. *Journal of Statistical Planning and Inference*, 139(7):2242–2253, 2009.

[96] Michele Pavon and Francesco Ticozzi. Discrete-time classical and quantum Markovian evolutions: Maximum entropy problems on path space. *Journal of Mathematical Physics*, 51(4):042104, 2010.

[97] Tryphon T Georgiou and Michele Pavon. Positive contraction mappings for classical and quantum Schrödinger systems. *Journal of Mathematical Physics*, 56(3):033301, 2015.

[98] Elizabeth Crosson and John Bowen. Quantum ground state isoperimetric inequalities for the energy spectrum of local hamiltonians. *arXiv:1703.10133*, 2017.

[99] Michael Jarret. Hamiltonian surgery: Cheeger-type gap inequalities for non-positive (stoquastic), real, and hermitian matrices. *arXiv:1804.06857*, 2018.

[100] Lov K Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the 28th Annual ACM Symposium on Theory of Computing*, pages 212–219. ACM, 1996.

[101] Esma Aïmeur, Gilles Brassard, and Sébastien Gambs. Quantum speed-up for unsupervised learning. *Machine Learning*, 90(2):261–287, 2013.

[102] Oded Goldreich, Shari Goldwasser, and Dana Ron. Property testing and its connection to learning and approximation. *Journal of the ACM*, 45(4): 653–750, 1998.

[103] Oded Goldreich and Dana Ron. On testing expansion in bounded-degree graphs. In *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation*, pages 68–75. Springer, 2011.

[104] Artur Czumaj, Pan Peng, and Christian Sohler. Testing cluster structure of graphs. In *Proceedings of the 47th Annual ACM Symposium on Theory of Computing*, pages 723–732. ACM, 2015.

[105] Ashish Chiplunkar, Michael Kapralov, Sanjeev Khanna, Aida Mousavifar, and Yuval Peres. Testing graph clusterability: Algorithms and lower bounds. In *Proceedings of the 59th Annual IEEE Symposium on Foundations of Computer Science*. IEEE, 2018.

[106] Scott Aaronson and Andris Ambainis. Quantum search of spatial regions. In *Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science*, pages 200–209. IEEE, 2003.

[107] Avatar Tulsi. Faster quantum-walk algorithm for the two-dimensional spatial search. *Physical Review A*, 78(1):012310, 2008.

[108] Pawel Wocjan and Anura Abeyesinghe. Speedup via quantum sampling. *Physical Review A*, 78(4):042336, 2008.

[109] Harry Buhrman, Christoph Durr, Mark Heiligman, Peter Høyer, Frédéric Magniez, Miklos Santha, and Ronald De Wolf. Quantum algorithms for element distinctness. In *Proceedings of the 16th Annual IEEE Conference on Computational Complexity*, pages 131–137. IEEE, 2001.

[110] Harry Buhrman and Robert Špalek. Quantum verification of matrix products. In *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 880–889. SIAM, 2006.

[111] Dominic W Berry and Andrew M Childs. Black-box hamiltonian simulation and unitary implementation. *Quantum Information and Computation*, 12(1& 2):29–62, 2012.

[112] Camille Jordan. Essai sur la géométrie à n dimensions. *Bulletin de la Société Mathématique de France*, 3:103–174, 1875.

[113] Andrew M Childs and Nathan Wiebe. Hamiltonian simulation using linear combinations of unitary operations. *Quantum Information and Computation*, 12(11& 12):901–924, 2012.

[114] Dominic W Berry, Andrew M Childs, Richard Cleve, Robin Kothari, and Rolando D Somma. Simulating hamiltonian dynamics with a truncated taylor series. *Physical Review Letters*, 114(9):090502, 2015.

[115] Joran Van Apeldoorn, András Gilyén, Sander Gribling, and Ronald de Wolf. Quantum SDP-solvers: better upper and lower bounds. In *Proceedings of the 58th Annual IEEE Symposium on Foundations of Computer Science*, pages 403–414. IEEE, 2017.

[116] Amparo Gil, Javier Segura, and Nico M Temme. *Numerical methods for special functions*, volume 99. SIAM, 2007.

[117] Andris Ambainis, Andrew M Childs, Ben W Reichardt, Robert Špalek, and Shengyu Zhang. Any and-or formula of size n can be evaluated in time $n^{1/2} + o(1)$ on a quantum computer. *SIAM Journal on Computing*, 39(6): 2513–2530, 2010.

[118] Andrew M Childs. On the relationship between continuous-and discrete-time quantum walk. *Communications in Mathematical Physics*, 294(2): 581–603, 2010.

[119] Andrew M Childs, Robin Kothari, and Rolando D Somma. Quantum algorithm for systems of linear equations with exponentially improved dependence on precision. *SIAM Journal on Computing*, 46(6):1920–1950, 2017.

[120] Gilles Brassard and Peter Høyer. An exact quantum polynomial-time algorithm for Simon's problem. In *Proceedings of the 5th Israeli Symposium on Theory of Computing and Systems*, pages 12–23. IEEE, 1997.

[121] Lov K Grover. Quantum computers can search rapidly by using almost any transformation. *Physical Review Letters*, 80(19):4329, 1998.

[122] Gilles Brassard, Peter Høyer, Michele Mosca, and Alain Tapp. Quantum amplitude amplification and estimation. *Contemporary Mathematics*, 305: 53–74, 2002.

[123] Andris Ambainis, Andrew M Childs, and Yi-Kai Liu. Quantum property testing for bounded-degree graphs. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 365–376. Springer, 2011.

[124] Paul Valiant. Testing symmetric properties of distributions. *SIAM Journal on Computing*, 40(6):1927–1968, 2011.

[125] Artur Czumaj and Christian Sohler. Testing expansion in bounded-degree graphs. *Combinatorics, Probability and Computing*, 19(5-6):693–709, 2010.

[126] Satyen Kale and Comandur Seshadhri. An expansion tester for bounded degree graphs. *SIAM Journal on Computing*, 40(3):709–720, 2011.

[127] Asaf Nachmias and Asaf Shapira. Testing the expansion of a graph. *Information and Computation*, 208(4):309, 2010.

[128] Oded Goldreich and Dana Ron. Property testing in bounded degree graphs. In *Proceedings of the 29th Annual ACM Symposium on Theory of Computing*, pages 406–415. ACM, 1997.

[129] Ashley Montanaro and Ronald de Wolf. A survey of quantum property testing. *Theory of Computing Library Graduate Surveys*, (7):1–81, 2016.

[130] Tuğkan Batu, Lance Fortnow, Ronitt Rubinfeld, Warren D Smith, and Patrick White. Testing closeness of discrete distributions. *Journal of the ACM (JACM)*, 60(1):4, 2013.

[131] Siu-On Chan, Ilias Diakonikolas, Paul Valiant, and Gregory Valiant. Optimal algorithms for testing closeness of discrete distributions. In *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1193–1203. SIAM, 2014.

[132] Shayan Oveis Gharan and Luca Trevisan. Partitioning into expanders. In *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1256–1266. SIAM, 2014.

[133] Daniel A Spielman and Shang-Hua Teng. Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems. In *Proceedings of the 36th Annual ACM symposium on Theory of Computing*, pages 81–90. ACM, 2004.

[134] Daniel A Spielman and Shang-Hua Teng. A local clustering algorithm for massive graphs and its application to nearly linear time graph partitioning. *SIAM Journal on Computing*, 42(1):1–26, 2013.

[135] Joshua Batson, Daniel A Spielman, Nikhil Srivastava, and Shang-Hua Teng. Spectral sparsification of graphs: theory and algorithms. *Communications of the ACM*, 56(8):87–94, 2013.

[136] Andris Ambainis, Loïck Magnin, Martin Roetteler, and Jérémie Roland. Symmetry-assisted adversaries for quantum state generation. In *Proceedings of the 26th Annual Conference on Computational Complexity*, pages 167–177. IEEE, 2011.

[137] Reid Andersen, Fan Chung, and Kevin Lang. Local graph partitioning using pagerank vectors. In *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science*, pages 475–486. IEEE, 2006.

[138] Reid Andersen and Yuval Peres. Finding sparse cuts locally using evolving sets. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing*, pages 235–244. ACM, 2009.

[139] Shayan Oveis Gharan and Luca Trevisan. Approximating the expansion profile and almost optimal local graph clustering. In *Proceedings of the 53rd Annual IEEE Symposium on Foundations of Computer Science*, pages 187–196. IEEE, 2012.

[140] Theodore J Yoder, Guang Hao Low, and Isaac L Chuang. Fixed-point quantum search with an optimal number of queries. *Physical Review Letters*, 113(21):210501, 2014.

[141] Yuval Peres and Perla Sousi. Mixing times are hitting times of large sets. *Journal of Theoretical Probability*, 28(2):488–519, 2015.

[142] Andris Ambainis, Julia Kempe, and Alexander Rivosh. Coins make quantum walks faster. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1099–1108. SIAM, 2005.

[143] Frédéric Magniez, Ashwin Nayak, Peter C Richter, and Miklos Santha. On the hitting times of quantum versus random walks. *Algorithmica*, 63(1-2): 91–116, 2012.

[144] Aleksandrs Belovs, Andrew M Childs, Stacey Jeffery, Robin Kothari, and Frédéric Magniez. Time-efficient quantum walks for 3-distinctness. In *International Colloquium on Automata, Languages, and Programming*, pages 105–122. Springer, 2013.

[145] Catalin Dohotaru and Peter Høyer. Controlled quantum amplification. In *LIPIcs-Leibniz International Proceedings in Informatics*, volume 80. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2017.

[146] Andris Ambainis and Martinš Kokainis. Analysis of the extended hitting time and its properties. Poster presented at *Conference on Quantum Information Processing*, 2015.

[147] Marios H Michael, Matti Silveri, RT Brierley, Victor V Albert, Juha Salmilehto, Liang Jiang, and Steven M Girvin. New class of quantum error-correcting codes for a bosonic mode. *Physical Review X*, 6(3):031006, 2016.

[148] Valentin Vladimirovich Petrov. *Sums of independent random variables*, volume 82. Springer Science & Business Media, 2012.

[149] Vedran Dunjko and Hans J Briegel. Quantum mixing of Markov chains for special distributions. *New Journal of Physics*, 17(7):073004, 2015.

[150] Charles H Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani. Strengths and weaknesses of quantum computing. *SIAM Journal on Computing*, 26(5):1510–1523, 1997.

[151] Ben W Reichardt and Robert Spalek. Span-program-based quantum algorithm for evaluating formulas. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing*, pages 103–112. ACM, 2008.

[152] Fernando GSL Brandão and Krysta M Svore. Quantum speed-ups for solving semidefinite programs. In *Proceedings of the 58th Annual IEEE Symposium on Foundations of Computer Science*, pages 415–426. IEEE, 2017.

[153] Andrew M Childs. Lecture notes on quantum algorithms. 2017.

[154] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. Quantum computation by adiabatic evolution. *arXiv:quant-ph/0001106*, 2000.

[155] Julia Kempe, Alexei Kitaev, and Oded Regev. The complexity of the local hamiltonian problem. *SIAM Journal on Computing*, 35(5):1070–1097, 2006.

[156] Guifre Vidal. Classical simulation of infinite-size quantum lattice systems in one spatial dimension. *Physical Review Letters*, 98(7):070201, 2007.

[157] Frank Verstraete, Valentin Murg, and Ignacio J Cirac. Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems. *Advances in Physics*, 57(2):143–224, 2008.

[158] Aleksandrs Belovs. Quantum walks and electric networks. *arXiv:1302.3143*, 2013.