View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Dagstuhl Research Online Publication Server

Efficient Quantum Walk on the Grid with Multiple Marked Elements*

Peter Høyer[1] and Mojtaba Komeili[2]

1 **Department of Computer Science, University of Calgary, Calgary, Canada**
hoyer@ucalgary.ca

2 **Department of Computer Science, University of Calgary, Calgary, Canada**
mojtaba.komeili@ucalgary.ca

Abstract

We give a quantum algorithm for finding a marked element on the grid when there are multiple marked elements. Our algorithm uses quadratically fewer steps than a random walk on the grid, ignoring logarithmic factors. This is the first known quantum walk that finds a marked element in a number of steps less than the square-root of the extended hitting time. We also give a new tighter upper bound on the extended hitting time of a marked subset, expressed in terms of the hitting times of its members.

1998 ACM Subject Classification F.1.2 Modes of Computation, F.2.2 Nonnumerical Algorithms and Problems, G.2.2 Graph Theory

Keywords and phrases Quantum walks, random walks, query complexity, spatial search

Digital Object Identifier 10.4230/LIPIcs.STACS.2017.42

1 Introduction

Searching structured and unstructured data is one of the most fundamental tasks in computer science. In many search problems in quantum computing, we are given a set of N elements of which M elements are marked, and our task is to find and output a marked element.

Search problems have been studied intensively and found many applications, both classically and quantumly. The first result on search within quantum computing was given by Bennett et al. [9], who showed in 1994 that any quantum algorithm requires $\Omega(\sqrt{N/M})$ steps to find a marked element. Grover [18] showed next that a quantum computer can find such a marked element in $O(\sqrt{N})$ steps, compared to $\Omega(N)$ for a classical computer. This quadratic speed-up was then generalized to arbitrary unstructured search problems by a generic amplitude amplification process by Brassard et al. [10].

Grover's algorithm and amplitude amplification are directly applicable to unstructured global search problems, but not to search problems relying on a local realization. Consider we have just inspected one of the N elements and found that it is not marked, and we want next to inspect another of the N elements. Many search problems have the localized property that it is less costly to inspect an element that is close to the most recently inspected element, as opposed to inspecting an arbitrary element. Many probabilistic algorithms for such problems use random walks, and the quantum analogue of such are called quantum walks.

Quantum walks have proven very successful in quantum computing, with applications in diverse settings such as communication complexity [1], element distinctness problems [3, 8],

* This work was partially supported by NSERC, the Natural Sciences and Engineering Research Council of Canada and CIFAR, the Canadian Institute for Advanced Research.

© Peter Høyer and Mojtaba Komeili;
licensed under Creative Commons License CC-BY

testing group commutativity [24], and triangle finding [27, 16]. Excellent surveys on quantum walks, their history and applications, include [2, 19, 32, 36, 31].

The expected number of steps H required to find a marked element by a random walk is called the *hitting time*. The hitting time depends on the structure being searched as well as the number and locations of the marked elements.

Quantum walks have been studied for many structures, and in particular for the torus. A torus is a graph containing N vertices laid out in a two-dimensional square structure. It is also called a grid or a two-dimensional lattice. The first work on the torus was by Aaronson and Ambainis [1], who showed that a torus can be searched in $O(\sqrt{N}\log^2 N)$ steps. Their breakthrough result is remarkably close to the quadratic speed-up that is attainable for unstructured search problems, and it raised the question of determining the limitations of quantum walks in general and on a torus in particular.

For the torus, Ambainis et al. [5] next gave a quantum walk using $O(\sqrt{N}\log N)$ steps. The question of whether one could find a marked element any faster was solved Tulsi [34] who found a quantum algorithm using $O(\sqrt{N\log N})$ steps, obtained by attaching an ancilla qubit and thereby modifying the search space. The above results assume the torus contains a single marked element. If there are multiple marked elements, one can probabilistically reduce the number of marked elements, potentially incurring an increased cost, and not what we would naturally expect and desire, a decreased cost.

For general walks, Szegedy [33] showed in an influential paper how to construct a quantum walk from any given symmetric random walk. Szegedy's algorithm detects the presence of a marked element in a number of steps of order $\sqrt{H}$ which is quadratically smaller than classical hitting time H. Szegedy's algorithm applies to any number M of marked elements, but does not necessarily find a marked element. In some cases, it outputs a marked element with success probability no better than if we simply sampled from the stationary distribution.

Magniez et al. [26] next showed how phase estimation can be applied to the larger class of reversible random walk, and gave an algorithm that both detects and finds a marked element. Their algorithm applies to any number of marked elements, but does not guarantee a quadratic speed-up in the hitting time. Magniez et al. [25] gave a quantum algorithm that detects the presence of a marked element for any reversible random walk in $O(\sqrt{H})$ steps. As Szegedy's algorithm, it applies to any number of marked elements, but it does not necessarily find a marked element. Magniez et al. [25] also gave a quantum algorithm that finds a unique marked element in $O(\sqrt{H})$ steps for any state-transitive random walk.

Krovi et al. [23] next introduced the novel idea of interpolating walks. Krovi et al. [22] show that interpolated walks can find a marked element for any reversible random walk, even with multiple marked elements. The algorithm does not guarantee a quadratic speed-up when there are multiple marked elements. Dohotaru and Høyer [15] introduced controlled quantum walks and showed that such walks also find a marked element for any reversible random walk, even with multiple marked elements, but again, not quadratically faster when there are multiple marked elements. The quantum algorithms given in both papers [22] and [15] use a number of steps in the order of a quantity called the *extended hitting time*.

The question of finding a marked element in quadratically fewer steps than by a random walk when there are multiple marked elements, has thus remained the main open question.

The torus has continued to be a canonical graph of study. Ambainis and Kokainis [6] show that for the torus, the extended hitting time can be $\Theta(N)$ while the hitting time is $O(1)$ when there are multiple marked elements. On the torus, we can find a unique marked element in $O(\sqrt{N\log N})$ steps with success probability of order $1/\log N$ by a continuous-time quantum walk [13] and by a coin-less quantum walk [7]. Ambainis et al. [4] show that the algorithm for

the torus in [5] can be modified, yielding a quantum algorithm that uses $O(\sqrt{N \log N})$ steps and finds a unique marked element with constant probability. Nahimovs and Santos [30] show that the probability the algorithm of [5] finds a marked element can be as small as $O(1/N)$ when there are two marked elements. Nahimovs and Rivosh [29] show that the locations of multiple marked elements on the torus can significantly impact the hitting time.

In this work, we give a quantum algorithm that finds a marked element quadratically faster than classically, up to a logarithmic factor, on the torus, no matter the number of marked elements. This is the first known quantum algorithm that finds a marked element faster than the square-root of the extended hitting time. For some instances, the extended hitting time is a factor of N larger than the hitting time.

We also analyze the extended hitting time. We give a new upper bound on the extended hitting time and prove that it is convex in the marked subset, with respect to the stationary distribution. These results are stated as Theorem 2 and Corollary 3 in Section 2. These two results yield in themselves a simplification of known quantum walks that are based on pruning the number of marked elements.

We next define and discuss the torus graph in Section 3. A major obstacle in finding a better quantum algorithm for the torus has been its locality properties. In Section 4, we investigate the locality properties of a random walk on the torus, and we turn these into our advantage, instead of being a disadvantage. We are sculpturing the connectivity. As argued by Meyer and Wong in [28], connectivity in itself is a poor indicator of fast quantum search. The idea of using properties of the underlying graph to direct the quantum walk to specific parts of the search space has been used elsewhere, e.g. by Le Gall in [16] to obtain the best known quantum algorithm for triangle finding.

In Section 5, we give our new quantum algorithm for finding a marked element on the torus when there are multiple marked elements. Our algorithm uses quadratically fewer steps than a random walk, ignoring logarithmic factors.

2 Bounds on the extended hitting time

Consider a Markov chain on a discrete finite state space X of size N. We represent its transition function as an $N \times N$ matrix P. The entries of P are real and non-negative. Entry P_{yx} denotes the probability of transitioning from state x to state y in one step. The entries in each column sum to one, implying that P is column-stochastic. We can consider the matrix P as the adjacency matrix of an underlying directed weighted graph.

We assume that the chain P is ergodic, which implies that it has a unique stationary distribution π satisfying that $\mathsf{P}\pi = \pi$. It follows from the Perron–Frobenius theorem that the stationary distribution π has real and positive entries. A Markov chain is *ergodic* if its underlying graph is strongly connected and acyclic.

We also assume that P is reversible. A Markov chain is *reversible* if $\mathsf{P}_{yx}\pi_x = \mathsf{P}_{xy}\pi_y$ for all states $x, y \in X$ in the state space. This condition expresses that the same amount of probability transition in either direction between any two states x and y in the stationary distribution. From now on, we will only consider Markov chains that are both ergodic and reversible, and we will also refer to such chains as random walks. Reversibility permits us to apply spectral analysis, following the seminal work of Szegedy [33].

Let $\mathcal{M} \subset X$ be the subset of marked states, and let $U = X \setminus \mathcal{M}$ be the remaining states which are unmarked. We form the *absorbing* walk P' from P by modifying all outgoing edges from marked states into self-loops. That is, if $x \in \mathcal{M}$ is marked, we set $\mathsf{P}'_{xx} = 1$ and $\mathsf{P}'_{yx} = 0$ for all other states $y \in X \setminus \{x\}$. We set $P'_{yx} = P_{yx}$ for all unmarked states $x \in U$ and all states $y \in X$. We will interchangeably refer to states as "elements."

The main goal of the random walk P is to find a marked state. The walk starts in a state drawn from the stationary distribution π. We keep applying the transition function until we reach a marked state, at which point the walk halts. The *hitting time* is the expected number of steps it takes for the random walk to find a marked state, and it is denoted by $\mathsf{HT}(\mathsf{P}, \mathcal{M})$.

We use spectral analysis to study the hitting time of random walks, as in Szegedy [33]. The *discriminant* of any given random walk P is the matrix $\mathsf{D}(P) = \sqrt{\mathsf{P} \circ \mathsf{P}^T}$, where T denotes matrix transposition, and where the Hadamard product $\circ$ denotes entry-wise product and the square-root is taken entry-wise. The discriminant is a symmetric real matrix by definition and thus has real eigenvalues.

We use both the discriminant of the walk P and its absorbing walk P'. The discriminant $\mathsf{D}(\mathsf{P})$ of P has real eigenvalues $1 = \lambda_1 > \lambda_2 \geq \cdots \geq \lambda_N > -1$ with corresponding eigenvectors $|\lambda_1\rangle, |\lambda_2\rangle, \ldots, |\lambda_N\rangle$. The *spectral gap* of P is $\delta = 1 - \lambda_2$.

The discriminant $\mathsf{D}(\mathsf{P}')$ of the absorbing walk P' has $|\mathcal{M}|$ eigenvectors $|x\rangle$ with eigenvalue $+1$, one for each marked state $x \in \mathcal{M}$. The remaining $N - |\mathcal{M}|$ eigenvectors $|\lambda'_1\rangle, |\lambda'_2\rangle, \ldots, |\lambda'_{N-|\mathcal{M}|}\rangle$ have eigenvalues $1 > \lambda'_1 \geq \lambda'_2 \geq \cdots \geq \lambda'_{N-|\mathcal{M}|} > -1$ that are strictly less than one in absolute value and they span the unmarked subspace.

The hitting time of P can then be expressed in terms of the spectra of the discriminant $\mathsf{D}(\mathsf{P}')$ for the absorbing walk. Let $|\pi\rangle = \sum_{x \in X} \sqrt{\pi_x}|x\rangle$ denote the column vector corresponding to the stationary distribution, normalized entry-wise. For any subset $S \subseteq X$ of elements, let $\epsilon_S = \sum_{x \in S} \pi_x$ denote the probability that the stationary distribution is in a state in S. Let $|S_\pi\rangle = \frac{1}{\sqrt{\epsilon_S}} \sum_{x \in S} \sqrt{\pi_x}|x\rangle$ denote the normalized projection of the stationary distribution $|\pi\rangle$ onto the subspace spanned by elements in a non-empty subset S. Let $|S_\pi\rangle$ be the vector of length zero if subset S is empty. In particular, we use $\epsilon_{\mathcal{M}}$ and ϵ_U, as well as $|\mathcal{M}_\pi\rangle$ and $|U_\pi\rangle$, to denote the quantities for the marked and unmarked subsets, respectively.

▶ **Lemma 1** (see e.g. [33, 31, 22]). *The hitting time of a reversible ergodic Markov chain* P *with marked elements* $\mathcal{M}$ *is*

$$\mathsf{HT}(\mathsf{P}, \mathcal{M}) = \sum_{k=1}^{N-|\mathcal{M}|} \frac{|\langle \lambda'_k | U_\pi \rangle|^2}{1 - \lambda'_k}. \tag{1}$$

The hitting time is the expected number of steps of P' required to reach a marked vertex, starting from a random unmarked vertex, picked according to the stationary distribution π. We define the *effective hitting time* $\mathsf{HT}_{\text{eff}}(\mathsf{P}, \mathcal{M})$ as the number of steps of P' required to reach a marked vertex with probability at least $2/3$, again starting from a random unmarked vertex, picked according to the stationary distribution π. By Markov's inequality, the effective hitting time is at most three times larger than the hitting time.

In his seminal paper, Szegedy [33] proved that we can *detect* whether a marked element exists or not quadratically faster by a quantum algorithm. If there are $|\mathcal{M}| > 0$ marked elements $\mathcal{M}$, it suffices to run Szegedy's quantum algorithm for $O\big(\sqrt{\mathsf{HT}_{\text{eff}}(\mathsf{P}, \mathcal{M})}\big)$ steps to determine the existence of a marked element with bounded one-sided error [33, 25].

A breakthrough for quantum walks with multiple marked elements was achieved by Krovi et al. [23, 21, 22]. They introduced a walk $\mathsf{P}(s) = (1 - s)\mathsf{P} + s\mathsf{P}'$ which is an interpolation between the non-absorbing walk P and the absorbing walk P'. The walk is parameterized by a quantity $0 < s < 1$, which is chosen to be very close to 1, implying that the walk is almost absorbing. They prove that their algorithm both detects and finds a marked element, even when there are multiple marked elements. The limitation is that their quantum walk does not necessarily guarantee a quadratic speedup over a classical walk. To measure the number of steps of their algorithm, they introduce a quantity $\mathsf{HT}^+(\mathsf{P}, \mathcal{M})$ called the *extended hitting*

time. Their algorithm takes a number of steps that is of order $\sqrt{\mathsf{HT}^+(\mathsf{P},\mathcal{M})}$, the square-root of the extended hitting time, ignoring logarithmic factors.

Their work raises two main questions. The first question is to determine the extent to which the extended hitting time can exceed the hitting time. The second question is to continue the quest for the discovery of a quantum algorithm that finds a marked element quadratically faster than a random walk when there are multiple marked elements.

Ambainis and Kokainis [6] considered the question of determining the largest possible ratio between the extended hitting time and the hitting time for a natural search space. They show that especially for the torus, the ratio can be exceptionally large by providing an example of a set of marked elements $\mathcal{M}$ on the torus for which $\mathsf{HT}(\mathsf{P},\mathcal{M}) \in O(1)$, yet $\mathsf{HT}^+(\mathsf{P},\mathcal{M}) \in \Theta(N)$. That is, the hitting time is a constant, yet the extended hitting time is linear in the size of the torus. Searching with multiple marked elements on the torus in the square-root of the extended hitting time can be remarkably slow.

In the case there is a single marked element ($\mathcal{M} = \{m\}$), the extended hitting time is identical to the hitting time, and thus $\mathsf{HT}^+(\mathsf{P},\{m\}) = \mathsf{HT}(\mathsf{P},\{m\})$. For multiple marked elements, Ambainis and Kokainis [6] proved a general upper bound on the extended hitting time of $\mathsf{HT}^+(\mathsf{P},\mathcal{M}) \leq \frac{1}{\epsilon}\frac{1}{\delta}$, which implies that the extended hitting time can be at most a factor of $\frac{1}{\delta}$ larger than the hitting time $\mathsf{HT}(\mathsf{P},\mathcal{M})$. For the torus, the spectral gap δ is of order $\frac{1}{N}$, which is so small that it permits the above ratio of order N of the extended hitting time over the hitting time.

To derive an efficient quantum algorithm for the torus for multiple marked elements, we first provide a new upper bound on the extended hitting time. We show that the extended hitting time on a marked set $\mathcal{M}$ is never more than the weighted average of the hitting times of any its constituents.

▶ **Theorem 2.** *Let* P *be a reversible ergodic random walk with stationary distribution* π*. Let* $\mathcal{M} = \cup_i \mathcal{M}_i$ *be the disjoint union of non-empty subsets* $\mathcal{M}_i$ *of marked elements. The extended hitting time on* $\mathcal{M}$ *is at most the weighted average of the extended hitting times of its subsets,*

$$\mathsf{HT}^+(\mathsf{P},\mathcal{M}) \in O\left(\sum_i \frac{\epsilon_i}{\epsilon_{\mathcal{M}}} \mathsf{HT}^+(\mathsf{P},\mathcal{M}_i)\right).$$

Here $\epsilon_i = \sum_{x\in\mathcal{M}_i} \pi_x$ *is the probability that* π *is in marked subset* $\mathcal{M}_i$*, and* $\epsilon_{\mathcal{M}} = \sum_{i\in\mathcal{M}} \epsilon_i$ *is the total probability that* π *is in a marked state.*

As a corollary, by letting each subset $\mathcal{M}_i$ be a singleton set, we obtain that the extended hitting time of a set is never more than the worst-case hitting time of its members.

▶ **Corollary 3.** *The extended hitting time* $\mathsf{HT}^+(\mathsf{P},\mathcal{M})$ *on a marked subset* $\mathcal{M}$ *is in the order of the maximum of the hitting times* $\mathsf{HT}(\mathsf{P},\{m\})$ *of its members* $m \in \mathcal{M}$*.*

There are two technical obstacles in analyzing and understanding the extended hitting time. Firstly, it is defined as a limit of the hitting time of the interpolated walk $\mathsf{P}(s)$ as the parameter s approaches 1. Secondly, it is expressed in terms of the spectra of the absorbing walk P', and that spectra changes as we change the set of marked elements. Fortunately, we can circumvent both obstacles by applying the following theorem from [15].

▶ **Theorem 4** ([15])**.** *For any reversible ergodic random walk* P *with marked elements* $\mathcal{M}$*,*

$$\mathsf{HT}^+(\mathsf{P},\mathcal{M}) \in \Theta\left(\frac{1}{\epsilon_{\mathcal{M}}}\mathsf{E}(\mathsf{P},\mathcal{M}_\pi)\right). \tag{2}$$

The theorem expresses the extended hitting time as a product of two factors. The first factor is $\frac{1}{\epsilon_{\mathcal{M}}}$, the inverse of the probability that the stationary distribution is in a marked state. The second factor is defined below and is a quantity that is expressed in terms of the spectra of P, the original walk and not the absorbing walk P'. Theorem 4 permits us to analyze the extended hitting time by analyzing the original walk P, a task that is often simpler than analyzing the absorbing walk P'.

▶ **Definition 5.** For any normalized vector $|g\rangle$ over the state space of the walk P, the *escape time* of $|g\rangle$ is

$$\mathsf{E}(\mathsf{P}, |g\rangle) = \sum_{k=2}^{N} \frac{|\langle \lambda_k | g \rangle|^2}{1 - \lambda_k}. \tag{3}$$

For any non-trivial subset $S \subseteq X$ of elements, define the escape time of S with respect to π as $\mathsf{E}(\mathsf{P}, S_\pi) = \mathsf{E}(\mathsf{P}, |S_\pi\rangle)$.

We will often omit the subscript π and simply write $\mathsf{E}(\mathsf{P}, S)$ for $\mathsf{E}(\mathsf{P}, S_\pi)$. By definition, the escape time is a weighted average over the reciprocals of all of the spectral gaps $1 - \lambda_k$ of the original walk P. It follows the escape time is at most the inverse of the (smallest) gap $\delta = 1 - \lambda_2$. Equation 2 then permits us to re-derive that the extended hitting time is at most $\frac{1}{\epsilon_{\mathcal{M}}} \frac{1}{\delta}$, as shown by Ambainis and Kokainis [6]. The escape time is at least $1/2$ for any normalized vector $|g\rangle$ orthogonal to the principal eigenvector $|\lambda_1\rangle$, since the denominator in Eq. 3 is upper bounded by 2. We next show that the escape time is at most additive.

▶ **Lemma 6.** *For any two disjoint subsets of elements S_1 and S_2,*

$$\mathsf{E}(\mathsf{P}, S_1 \cup S_2) \leq \mathsf{E}(\mathsf{P}, S_1) + \mathsf{E}(\mathsf{P}, S_2).$$

Proof. Fix a distribution over the vertex set V, e.g. the stationary distribution π. The lemma holds trivially if S_1 or S_2 is the empty set, and thus assume that both sets are non-empty. Write the normalized state $|S_1 \cup S_2\rangle = a|S_1\rangle + b|S_2\rangle$ as a linear combination of the normalized and orthogonal elements $|S_1\rangle$ and $|S_2\rangle$. Noting that $a^2 + b^2 = 1$, by the Cauchy–Schwarz inequality, $|\langle \lambda_k | S_1 \cup S_2 \rangle|^2$ is then at most the sum of $|\langle \lambda_k | S_1 \rangle|^2$ and $|\langle \lambda_k | S_2 \rangle|^2$. We can thus upper bound the sum of the terms $\frac{|\langle \lambda_k | S_1 \cup S_2 \rangle|^2}{1-\lambda_k}$ by one sum over terms of the form $\frac{|\langle \lambda_k | S_1 \rangle|^2}{1-\lambda_k}$, plus another sum over terms of the form $\frac{|\langle \lambda_k | S_2 \rangle|^2}{1-\lambda_k}$. The former sum is the escape time of S_1, the latter the escape time of S_2. ◀

We next use the sub-additivity of the escape time to prove that the extended hitting time on a marked subset $\mathcal{M}$ is never more than the extended hitting time of any of constituents.

Proof of Theorem 2. Let $\mathcal{M} = \mathcal{M}_1 \cup \mathcal{M}_2$ be a disjoint union of two non-empty subsets of marked elements. Let $\epsilon_1 = \epsilon_{\mathcal{M}_1}$ be the probability that the stationary distribution π is in a marked state in $\mathcal{M}_1$, and let $\epsilon_2 = \epsilon_{\mathcal{M}_2}$ be defined similarly. Let $\epsilon_{\mathcal{M}} = \epsilon_1 + \epsilon_2$.

Using Equation 2 and Lemma 6, and omitting asymptotic tight factors, write

$$\begin{aligned}
\mathsf{HT}^+(\mathsf{P}, \mathcal{M}) &= \frac{1}{\epsilon_{\mathcal{M}}} \mathsf{E}(\mathsf{P}, \mathcal{M}) \\
&\leq \frac{1}{\epsilon_{\mathcal{M}}} \big(\mathsf{E}(\mathsf{P}, \mathcal{M}_1) + \mathsf{E}(\mathsf{P}, \mathcal{M}_2) \big) \\
&= \frac{\epsilon_1}{\epsilon_{\mathcal{M}}} \frac{1}{\epsilon_1} \mathsf{E}(\mathsf{P}, \mathcal{M}_1) + \frac{\epsilon_2}{\epsilon_{\mathcal{M}}} \frac{1}{\epsilon_2} \mathsf{E}(\mathsf{P}, \mathcal{M}_2) \\
&= \frac{\epsilon_1}{\epsilon_{\mathcal{M}}} \mathsf{HT}^+(\mathsf{P}, \mathcal{M}_1) + \frac{\epsilon_2}{\epsilon_{\mathcal{M}}} \mathsf{HT}^+(\mathsf{P}, \mathcal{M}_2).
\end{aligned}$$

Theorem 2 follows by linearity. ◀

Corollary 3 has an important and previously unrecognized consequence. Consider we are given some computational problem that has multiple solutions, and assume that we know how to solve the problem when there is a unique solution. Then we may be able to device a randomized polynomial-time reduction that probabilistically makes all solutions but one into non-solutions, and then find the only remaining solution. Such pruning ideas have been used in e.g. reducing SAT to unique-SAT [35] and finding a marked element on the torus by Aaronson and Ambainis [1]. As stated in e.g. [25], randomized reductions of multiple marked elements to a unique marked element may increase the cost by a poly-logarithmic factor.

Theorem 2 yields an alternative to such reductions. We simply just run either the controlled quantum walk of [15] or the interpolated quantum walk of [22]. Both algorithms take a number of steps in the order of the square-root of the extended hitting time, ignoring logarithmic factors. By Theorem 2, the extended hitting time of a subset $\mathcal{M}$ is upper bounded by the average of the hitting times of its members, where the average is with respect to the stationary distribution π. Provided we are given an estimate $\tilde{\epsilon}$ of $\epsilon_{\mathcal{M}}$ satisfying that $\frac{2}{3} \leq \frac{\tilde{\epsilon}}{\epsilon_{\mathcal{M}}} \leq \frac{4}{3}$, then we find a marked element with probability at least $1/5$. (Apply e.g. Theorem 7 in [22] with the value $\epsilon_2 = \frac{1}{100}$ in their proof.)

▶ **Corollary 7.** *Given a reversible ergodic Markov chain* P *with marked elements* $\mathcal{M}$*, and an estimate* $\tilde{\epsilon}$ *of* $\epsilon_{\mathcal{M}}$ *satisfying that* $\frac{2}{3} \leq \frac{\tilde{\epsilon}}{\epsilon_{\mathcal{M}}} \leq \frac{4}{3}$*, we can find a marked state by a quantum walk with probability at least* $1/5$ *using in the order of* $\sqrt{\mathsf{HT}(\mathsf{P}, \{m\})}$ *steps, where* $m \in \mathcal{M}$ *is chosen to maximize the upper bound.*

One advantage of applying an algorithm that runs in the square-root of the extended hitting time, is that no direct pruning is necessary. We do not need to turn marked elements into non-marked elements. It suffices to guess an estimate $\tilde{\epsilon}$ of the probability $\epsilon_{\mathcal{M}}$ of measuring a marked state in the stationary distribution. A second advantage is that the extended hitting time of a subset can be significantly less than the average of the hitting times of its members. The bounds in Theorem 2 and Lemma 6 are only upper bounds, not tight bounds. The bounds can not be improved in general as they are tight for some instances. One such example is the case considered by Nahimovs and Rivosh [29] of the torus with multiple marked elements packed as densely as possible into a sub-square.

3 The torus graph

We consider walks on two-dimensional square torus graphs. The graph contains $N = n^2$ vertices organized into n rows and n columns. There is one vertex at location (r, c) for each row $0 \leq r < n$ and column $0 \leq c < n$. The graph is directed and every vertex has in-degree 4 and out-degree 4.

We consider two types of boundary conditions. We define the *torus* graph in the usual way, with vertices along the boundary connecting to vertices on the opposite boundary. A vertex at location (r, c) is connected to its four neighbors at locations $(r-1, c)$, $(r+1, c)$, $(r, c-1)$, and $(r, c+1)$, where the addition is modulo n.

We define the *grid* graph to have self-loops on vertices on the boundary. A vertex at location (r, c) has four out-going edges pointing to the locations $(\max\{r-1, 0\}, c)$, $(\min\{r+1, n-1\}, c)$, $(r, \max\{c-1, 0\})$, and $(r, \min\{c+1, n-1\})$. Every vertex has in-degree 4 and out-degree 4, as for the torus.

Prior to this Section, all of our discussions have been for the torus, not the grid. Our algorithm given in Section 5 uses both the torus and grid graphs. Since one cannot replace a walk on a torus that crosses the boundary by a walk on a grid without potentially incurring

a cost, we need to clearly distinguish between the two graphs. Our algorithm in Section 5 works for both the torus and grid.

We form a random walk P_G on a graph G by taking the adjacency matrix $\mathrm{Adj}(G)$ of G and normalize its *columns*. For every directed edge $(u, v) \in G$, we set entry (v, u) in P_G to be the inverse of the out-degree of vertex u. All other entries of P_G are zero. Since both the torus and grid are regular graphs G with out-degree 4, their random walk operators $\mathsf{P}_G = \frac{1}{4}\mathrm{Adj}(G)$ are scaled versions of their adjacency matrices.

Our proposed quantum algorithm for finding a marked state on the torus uses both the torus and grid graphs. Since the torus and grid are so closely related, it seems intuitively obvious that walking on either graphs should have little influence on the complexity of the algorithm. Indeed, the escape times on the torus and grid with $N = |V|$ vertices of an element $m \in V$ are both of order $\log N$.

▶ **Fact 8.** *The escape times of an element $m \in V$ on the torus and grid are both of order $\log N$,*

$$\mathsf{E}(\mathsf{P}_{\mathrm{torus}}, \{m\}) \in \Theta(\log N) \qquad \text{and} \qquad \mathsf{E}(\mathsf{P}_{\mathrm{grid}}, \{m\}) \in \Theta(\log N).$$

The above fact can be derived from known facts that the hitting time of a unique element on the torus and grid are of order $N \log N$ and then applying Theorem 4. The fact can also be shown directly by first computing the spectra for the torus and grid, as done in e.g. [5] for the torus, and then applying Definition 5.

4 Locality of random walks on the torus

To obtain a faster quantum algorithm, we first need to resolve the main obstacle that a random walk on a torus is localized.

▶ **Lemma 9.** *The probability that a random walk of T steps on an infinite line stays within distance $\lceil 4\sqrt{T} \rceil$ from the initial position is at least $1 - 1/745$.*

Proof. Consider a walk on a doubly-infinite line. The walk starts in some fixed initial position, and we measure distances from this initial position. The probability that the walk is at distance (strictly) more than k from the initial position after ℓ steps is at most $2\exp(-\frac{k^2}{2\ell})$ by the Azuma–Hoeffding inequality.

The conditional probability that the walk ends at a distance larger than k, conditioned on that the walk ever reaches distance $k+1$, is at least $1/2$ by the reflection principle: once the walk reaches distance $k+1$, the walk is equally likely to end on either side of that location.

By Bayes's rule, the probability that the walk reaches distance $k+1$ is then at most $4\exp(-\frac{k^2}{2\ell})$ which is at most $4e^{-8}$ when $k = \lceil 4\sqrt{T} \rceil$. Finally, $4e^{-8}$ is less than $1/745$. ◀

Since the grid is the cartesian graph product of two line graphs, we immediately get that a walk on the grid is also locally contained.

▶ **Lemma 10.** *The probability that a random walk of T steps on an infinite grid stays within distance $\lceil 4\sqrt{T} \rceil$ in all four directions from the initial position is at least $1 - 2/745$.*

Let us say that a walk is *localized* if it stays within distance $\lceil 4\sqrt{T} \rceil$ in all four directions from its initial position u.

This locality property implies that we can substitute the global walk by disjoint local walks. Consider a torus of size $n \times n$, and fix an integer $1 \leq d < n$. We cut the torus into $\Theta((\frac{n}{d})^2)$ disjoint graph components by removing edges from the torus. We remove edges that

cross graph cuts so that each resulting component is a sub-grid (without self-loops on the boundary vertices) and so that all components have length and width that are at least D and at most $D+1$, for some $d \leq D < 2d$. We next add self-loops to every vertex on the boundaries of the resulting graph components. The overall effect is that we have modified the $n \times n$ torus into $\Theta((\frac{n}{d})^2)$ disjoint grid graphs, each of size roughly $D \times D$, by turning edges between adjacent components into self-loops.

Now, consider a random walk of T steps on the torus of size $N = n^2$ starting from the stationary distribution π. We set $d = 2\lceil 4\sqrt{T} \rceil$ and modify the torus into $\Theta(\frac{N}{T})$ disjoint sub-grids as described above. We next sample one of these $\Theta(\frac{N}{T})$ sub-grids according to the stationary distribution π for the torus. That is, we sample the sub-grid G with probability $\epsilon_G = \sum_{v \in G} \pi_v$. The next lemma shows that the probability that a random sub-grid G contains at least one marked vertex is high.

▶ **Lemma 11.** *If a random walk of T steps on the torus of size N finds a marked vertex with probability at least p, for $p \geq \frac{1}{74}$, then the probability p_G that a random sub-grid, sampled from the $\Theta(\frac{N}{T})$ sub-grids as described above, contains at least one marked vertex is at least $\frac{1}{5}p$.*

Proof. We define the probabilities p_{ml} and p_{Gl} below and prove the following three inequalities,

$$p - \frac{2}{745} \leq p_{ml} \leq p_{Gl} \leq 4p_G.$$

By these three inequalities, when $p \geq \frac{1}{74}$ then $p_G \geq \frac{1}{5}p$, and the lemma follows.

Sample a random walk ω of length T as follows: Pick a vertex u on the torus according to the stationary distribution π, and apply the stochastic matrix $\mathsf{P}_{\text{torus}}$ a number of T times, starting at u.

Let p_{ml} be the probability that a sampled walk is localized and visits a marked vertex. Let p_{Gl} be the probability that a sampled walk is localized and visits a vertex that is in a sub-grid that contains a marked vertex. Clearly, $p_{ml} \leq p_{Gl}$, proving the second inequality.

The unconditional probability that the walk ω visits a marked vertex is at least p. The unconditional probability that ω is *not* localized is at most $\frac{2}{745}$ by Lemma 10. The joint probability that ω visits a marked vertex and is localized, is then at least $p - \frac{2}{745}$, proving the first inequality.

Fix any sub-grid G and consider the joint probability that the sampled walk ω is localized and visits G. For a localized walk ω to visit G, its starting vertex must be within distance $\lceil 4\sqrt{T} \rceil$ of G. Since G has width and length at least $2\lceil 4\sqrt{T} \rceil$, the number of such vertices is at most four times the number of vertices in G. The probability of sampling any of these as the starting vertex u from the stationary distribution π, which is uniform, is then at most four times the probability of sampling the starting vertex u from G itself. This proves the third inequality, and thus also the lemma. ◀

In the next Section, we use this locality property in the design of our quantum algorithm and give an efficient algorithm for the torus with multiple marked elements.

5 An efficient quantum algorithm for the torus

An implementation of a random walk is done as follows. We first pick a starting node $v \in V$ for the random walk. This node is picked according to the stationary distribution π. The cost of generating v is called the *setup cost* and is denoted by S. We next apply the absorbing walk P' for some number T steps. Each step consists of two parts: We first check whether

the node v we are currently located at is marked or not. The cost of checking whether a node is marked or not is called the *checking cost* and is denoted by C. If v is marked, we halt the algorithm and output v. If v is not marked, we next apply the operator P once. The cost of applying P is called the *update cost* and is denoted by U. After repeating these two parts for T steps, we check whether the final node v is marked or not. If so, we output v, and if not, we output "failed search." A random walk with T steps has total cost $\mathsf{S} + (T+1)\mathsf{C} + T\mathsf{U}$, which we write as $\mathsf{S} + T(\mathsf{U} + \mathsf{C})$ by letting the setup cost include the cost of the first checking cost. The walk outputs a marked element with constant probability when $T \in \Omega(\mathsf{HT}_{\text{eff}}(\mathsf{P}_{\text{torus}}, \mathcal{M}))$.

A quantum walk is implemented similarly [33, 32, 31]. We first create some initial state $|\text{init}\rangle$ from the state $|\pi\rangle = \sum_{v \in V} \sqrt{\pi_v}|v\rangle$, where π is the stationary distribution. We next apply some number T_q steps of the quantum walk, where each step consists of one application of each of two quantum operators, one denoted $\mathsf{Ref}(\mathcal{M})$ and one denoted $\mathsf{W}(\mathsf{P})$, corresponding to the checking and update operators applied in a random walk. The algorithm stops after T_q steps in some final state. The costs of these three operators are also denoted S, C, and U, and are in general comparable in cost to the corresponding operators for the random walk. The quantum algorithm has total cost $\mathsf{S} + T_q(\mathsf{U} + \mathsf{C})$.

A measurement of the final state of the quantum walk will not necessarily produce a marked state with constant probability. Had that been the case, then we would have had a quantum algorithm that *finds* a marked element in cost $\mathsf{S} + T_q(\mathsf{U} + \mathsf{C})$. Instead, the quantum walk evolves the initial state $|\text{init}\rangle$ away from the initial state, and this evolution away from the initial state is sufficient to *detect* that a marked state exists. Szegedy [33] show that after T_q steps, for some $T_q \in \Theta\big(\sqrt{\mathsf{HT}_{\text{eff}}(\mathsf{P}, \mathcal{M})}\big)$, the final state has overlap bounded away from 1 with the initial state. A change by a constant in overlap can be detected by standard techniques such as the swap test [12]. If the swap test shows the final state is different from the initial state, we deduce there is a marked state. We learn that there exists a marked element, but we do *not* necessarily find one. It is possible to efficiently estimate the speed of the change in overlap by applying eigenvalue estimation [20] similar to its uses in e.g. quantum counting [11], phase estimation [14], and quantum walks [26, 25, 22].

▶ **Theorem 12.** *There exists a quantum algorithm that given a reversible ergodic random walk* P *with marked elements* $\mathcal{M} \subset X$*, with probability at least* $2/3$ *performs as follows: (1) it outputs an estimate* $\tilde{h} \in O(\mathsf{HT}_{\text{eff}}(\mathsf{P}, \mathcal{M}))$ *satisfying that if we apply* P' *for* $\tilde{h}$ *steps starting from the initial distribution* π*, we find a marked state with probability at least* $3/4$ *and (2) it has cost in the order of* $\mathsf{S} + \sqrt{\tilde{h}}(\mathsf{U} + \mathsf{C})$.

Theorem 12 states that there is a quantum algorithm that, with probability at least $2/3$, computes an accurate estimate of the effective hitting time efficiently. With complementary probability at most $1/3$, this does not happen. We can prevent that the algorithm in Theorem 12 never terminates or terminates after a significant cost. Let $H_{\text{unique}} = \mathsf{HT}_{\text{eff}}(\mathsf{P}_{\text{torus}}, \{m\}) \in \Theta(N \log N)$ be the effective hitting time for the torus when there is a unique marked element. If the algorithm in Theorem 12 has not halted after $\sqrt{H_{\text{unique}}}$ steps, we halt the algorithm and output H_{unique} as our estimate $\tilde{h}$.

With this, we can give our quantum algorithm for finding a marked element on the torus. Our algorithm works for multiple marked elements, finds a marked element with probability at least $1/\log N$, and has cost in the order of

$$\mathsf{S} + \min\left\{\sqrt{H \log H}, \sqrt{N \log N}\right\}(\mathsf{U} + \mathsf{C}),$$

where $H = \mathsf{HT}_{\text{eff}}(\mathsf{P}_{\text{torus}}, \mathcal{M})$ is the *effective* hitting time on marked subset $\mathcal{M}$. This is within a poly-logarithmic factor of being a quadratic speed-up over the cost of a random walk, which has cost the effective hitting time.

▶ **Theorem 13** (Main). *There is a quantum algorithm that, given a torus* $\mathsf{P}_{\text{torus}}$ *with marked vertices* $\mathcal{M} \subset V$*, with probability at least* $2/3$*, outputs a marked element* $m \in \mathcal{M}$ *with success probability at least* $\frac{1}{\log N}$ *in cost in the order of* $\mathsf{S} + \min\left\{\sqrt{H \log H}, \sqrt{N \log N}\right\}(\mathsf{U} + \mathsf{C})$*, where* $H = \mathsf{HT}_{\text{eff}}(\mathsf{P}_{\text{torus}}, \mathcal{M})$ *is the effective hitting time.*

The input to the algorithm in Theorem 13 is a torus $\mathsf{P}_{\text{torus}}$ of size $n \times n$ with some subset $\mathcal{M} \subset V$ of vertices being marked. The algorithm is as follows.

1. Compute an estimate $\tilde{h} \in O(\mathsf{HT}_{\text{eff}}(\mathsf{P}_{\text{torus}}, \mathcal{M}))$ using Theorem 12. If the algorithm in Theorem 12 has not halted after $\sqrt{H_{\text{unique}}}$ steps, where $H_{\text{unique}} = \mathsf{HT}_{\text{eff}}(\mathsf{P}_{\text{torus}}, \{m\})$, halt it and use H_{unique} as our estimate $\tilde{h}$.
2. Set $d = 2\lceil 4\sqrt{\tilde{h}} \rceil$. If $d > n$, then set $d = n$.
3. Divide the $n \times n$ torus into disjoint sub-grids so that each sub-grid has length and width between D and $D + 1$, for some $d \leq D < 2d$.
4. Create the initial state $|\pi\rangle = \sum_{v \in \mathsf{P}_{\text{torus}}} \sqrt{\pi_v}|v\rangle$ over all vertices in the torus corresponding to the stationary distribution π for the torus.
5. For each vertex $v \in \mathsf{P}_{\text{torus}}$, assign the name of the sub-grid that v belongs to in an ancilla register, $\sum_{v \in \mathsf{P}_{\text{torus}}} \sqrt{\pi_v}|v\rangle|\text{subgrid}(v)\rangle$, in superposition.
6. Set $\tilde{\epsilon} = \frac{1}{2^k}$, where integer k is picked uniformly at random satisfying $1 \leq 2^k < N$.
7. Run a controlled quantum walk with estimate $\tilde{\epsilon}$ on each sub-grid for $\Theta(D\sqrt{\log D})$ steps in superposition over all sub-grids, by conditioning the walk on the name of the sub-grid in the ancilla register.
8. Measure the final state, producing a vertex v of the torus. Check if v is marked. If so, output v. If not, output "unsuccessful search."

We first prove the correctness of the algorithm. Assume that the first step of the algorithm outputs a suitable estimate for the effective hitting time as given in Theorem 12. This event happens with probability at least $\frac{2}{3}$. Then the probability that a random sub-grid contains at least one marked vertex is at least $\frac{1}{5} \times \frac{3}{4} = \frac{3}{20}$, by Lemma 11. For each of those sub-grids, by Corollary 7, the controlled quantum walk in step seven finds a marked element with probability at least $\frac{1}{5}$, for at least one of the $\log N$ possible values for k. Note that in step seven, each of the conditional walks on the sub-grids start the walk on the state corresponding to the stationary distribution for that sub-grid. The entire algorithm thus outputs a marked element with probability at least $\frac{2}{3} \times \frac{3}{20} \times \frac{1}{5} = \frac{1}{50}$ for at least one of the $\log N$ possible values for k.

The cost of the algorithm is easily deduced. With probability at least $2/3$, two properties hold: (1) the estimate $\tilde{h}$ computed in the first step is in the order of the effective hitting time $\mathsf{HT}_{\text{eff}}(\mathsf{P}_{\text{torus}}, \mathcal{M})$, and (2) the first and seventh steps of the algorithm each uses a number of steps that is in the order of $\mathsf{S} + \sqrt{\tilde{h} \log \tilde{h}}(\mathsf{U} + \mathsf{C})$. Further, in all events, each of the seven steps of the algorithm never uses more than in the order of $\sqrt{N \log N}(\mathsf{U} + \mathsf{C})$ steps of a quantum walk. Theorem 13 follows.

We remark that we can test all $\log N$ possible values for k by testing each of them in turn. This will increase the overall cost by a factor of $\log N$ and lead to an algorithm with constant success probability. By applying amplitude amplification [10], the increase in cost can be improved to being a factor of order $\sqrt{\log N}$. By testing each value of k in increasing order in turn, our algorithm can be made to have the same cost as the best known quantum algorithms when there is a unique marked element.

We remark that in step seven, we need to run a quantum walk that finds a marked element in the sub-grid, even if the sub-grid contains multiple marked elements. The controlled quantum walk of [15] and the interpolated walk of [22] both do so in $O(D\sqrt{\log D})$ steps,

when provided an estimate of the probability $\epsilon_{\mathcal{M}}$. We also remark that one may omit the conditioning of the quantum walk in step seven by measuring the ancilla register containing the name of a sub-grid immediately after step five. Conducting measurements as early as possible in a quantum algorithm is frequently used when no further computations are required. An early example of such is the semi-classical quantum Fourier transform by Griffiths and Niu [17].

6 Concluding remarks

We have given an efficient quantum algorithm for a finding a marked element on the torus with probability at least $1/\log N$ when there are multiple marked elements. Our algorithm has cost in the order of $\mathsf{S} + \sqrt{H_{\text{eff}} \log H_{\text{eff}}}(\mathsf{U} + \mathsf{C})$, where $H_{\text{eff}} = \mathsf{HT}_{\text{eff}}(\mathsf{P}_{\text{torus}}, \mathcal{M})$ is the number of steps used by the random walk $\mathsf{P}_{\text{torus}}$ to find any one of the marked elements $\mathcal{M}$. This is a quadratic speed-up, up to a poly-logarithmic factor. It is the first known quantum walk that has cost less than the square-root of the extended hitting time. It is, for the torus, an affirmative answer to the main open question in quantum walks whether it is possible to find a marked element efficiently when there are multiple marked elements.

The study of the torus has proven influential for at least two reasons. Firstly, much progress in quantum walks has been initiated by work on the cycle and the torus, and then later generalized to arbitrary graphs. Secondly, the torus is a hard test-case because the ratio between its hitting time and the reciprocal of its $\epsilon\delta$ bound [33] is large. For a unique marked element, the hitting time is of order $N \log N$, and the reciprocal of the $\epsilon\delta$ bound is of order N^2, and thus almost quadratically bigger.

In this work, we have proposed to use localization to our advantage in quantum search, and not as an obstacle to be overcome. We show that localization makes quantum search efficient when there are multiple marked elements on the torus.

References

1 Scott Aaronson and Andris Ambainis. Quantum search of spatial regions. *Theory of Computing*, 1(4):47–79, 2005. doi:10.4086/toc.2005.v001a004.

2 Andris Ambainis. Quantum walks and their algorithmic applications. *International Journal of Quantum Information*, 1(4):507–518, 2003. doi:10.1142/S0219749903000383.

3 Andris Ambainis. Quantum walk algorithm for element distinctness. *SIAM Journal on Computing*, 37(1):210–239, 2007. doi:10.1137/S0097539705447311.

4 Andris Ambainis, Artūrs Bačkurs, Nikolajs Nahimovs, Raitis Ozols, and Alexander Rivosh. Search by quantum walks on two-dimensional grid without amplitude amplification. In *Conference on the Theory of Quantum Computation, Communication and Cryptography*, TQC'12, pages 87–97, 2012. doi:10.1007/978-3-642-35656-8_7.

5 Andris Ambainis, Julia Kempe, and Alexander Rivosh. Coins make quantum walks faster. In *16th Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA'05, pages 1099–1108, 2005. arXiv:quant-ph/0402107.

6 Andris Ambainis and Mārtiņš Kokainis. Analysis of the extended hitting time and its properties, 2015. Poster presented at the 18th Annual Conference on Quantum Information Processing, QIP'15, Sydney, Australia.

7 Andris Ambainis, Renato Portugal, and Nikolajs Nahimovs. Spatial search on grids with minimum memory. *Quantum Information & Computation*, 15(13-14):1233–1247, 2015. arXiv:1312.0172.

8 Aleksandrs Belovs, Andrew M. Childs, Stacey Jeffery, Robin Kothari, and Frédéric Magniez. Time-efficient quantum walks for 3-distinctness. In *40th International Colloquium on Automata, Languages, and Programming*, ICALP'13, pages 105–122, 2013. doi:10.1007/978-3-642-39206-1_10.

9 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. doi:10.1137/S0097539796300933.

10 Gilles Brassard, Peter Høyer, Michele Mosca, and Alain Tapp. Quantum amplitude amplification and estimation. In Samuel J. Lomonaco, Jr. and Howard E. Brandt, editors, *Quantum Computation and Quantum Information*, volume 305 of *AMS Contemporary Mathematics*, pages 53–74. American Mathematical Society, 2002. arXiv:quant-ph/0005055.

11 Gilles Brassard, Peter Høyer, and Alain Tapp. Quantum counting. In *25th International Colloquium on Automata, Languages, and Programming*, ICALP'98, pages 820–831, 1998. doi:10.1007/BFb0055105.

12 Harry Buhrman, Richard Cleve, John Watrous, and Ronald de Wolf. Quantum fingerprinting. *Physical Review Letters*, 87(16):167902, 2001. doi:10.1103/PhysRevLett.87.167902.

13 Andrew M. Childs and Yimin Ge. Spatial search by continuous-time quantum walks on crystal lattices. *Physical Review A: General Physics*, 89:052337, 2014. doi:10.1103/PhysRevA.89.052337.

14 Richard Cleve, Artur Ekert, Chiara Macchiavello, and Michele Mosca. Quantum algorithms revisited. *Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 454(1969):339–354, 1998. doi:10.1098/rspa.1998.0164.

15 Cătălin Dohotaru and Peter Høyer. Controlled quantum amplification, 2016. Manuscript. To be presented at the 20th Annual Conference on Quantum Information Processing, QIP'17, Seattle, USA.

16 François Le Gall. Improved quantum algorithm for triangle finding via combinatorial arguments. In *55th IEEE Symposium on Foundations of Computer Science*, FOCS'14, pages 216–225, 2014. doi:10.1109/FOCS.2014.31.

17 Robert B. Griffiths and Chi-Sheng Niu. Semiclassical Fourier transform for quantum computation. *Physical Review Letters*, 76:3228–3231, 1996. doi:10.1103/PhysRevLett.76.3228.

18 Lov K. Grover. A fast quantum mechanical algorithm for database search. In *28th Annual ACM Symposium on Theory of Computing*, STOC'96, pages 212–219, 1996. doi:10.1145/237814.237866.

19 Julia Kempe. Quantum random walks: An introductory overview. *Contemporary Physics*, 44(4):307–327, 2003. doi:10.1080/0010715103100011077 6.

20 Alexei Yu. Kitaev. Quantum measurements and the abelian stabilizer problem, 1995. arXiv:quant-ph/9511026.

21 Hari Krovi, Frédéric Magniez, Māris Ozols, and Jérémie Roland. Finding is as easy as detecting for quantum walks. In *37th International Colloquium on Automata, Languages, and Programming*, ICALP'10, pages 540–551, 2010. doi:10.1007/978-3-642-14165-2_46.

22 Hari Krovi, Frédéric Magniez, Māris Ozols, and Jérémie Roland. Quantum walks can find a marked element on any graph. *Algorithmica*, 74(2):851–907, 2016. doi:10.1007/s00453-015-9979-8.

23 Hari Krovi, Māris Ozols, and Jérémie Roland. Adiabatic condition and the quantum hitting time of Markov chains. *Physical Review A: General Physics*, 82(2):022333, 2010. doi:10.1103/PhysRevA.82.022333.

24 Frédéric Magniez and Ashwin Nayak. Quantum complexity of testing group commutativity. *Algorithmica*, 48(3):221–232, 2007. doi:10.1007/s00453-007-0057-8.

25 Frédéric Magniez, Ashwin Nayak, Peter C. Richter, and Miklos Santha. On the hitting times of quantum versus random walks. *Algorithmica*, 63(1):91–116, 2012. doi:10.1007/s00453-011-9521-6.

26 Frédéric Magniez, Ashwin Nayak, Jérémie Roland, and Miklos Santha. Search via quantum walk. *SIAM Journal on Computing*, 40(1):142–164, 2011. doi:10.1137/090745854.

27 Frédéric Magniez, Miklos Santha, and Mario Szegedy. Quantum algorithms for the triangle problem. *SIAM Journal on Computing*, 37(2):413–424, 2007. doi:10.1137/050643684.

28 David A. Meyer and Thomas G. Wong. Connectivity is a poor indicator of fast quantum search. *Physical Review Letters*, 114:110503, 2015. doi:10.1103/PhysRevLett.114.110503.

29 Nikolajs Nahimovs and Alexander Rivosh. Quantum walks on two-dimensional grids with multiple marked locations. In *42nd International Conference on Current Trends in Theory and Practice of Computer Science*, pages 381–391, 2016. doi:10.1007/978-3-662-49192-8_31.

30 Nikolajs Nahimovs and Raqueline A. M. Santos. Adjacent vertices can be hard to find by quantum walks, May 2016. arXiv:1605.05598.

31 Ashwin Nayak, Peter C. Richter, and Mario Szegedy. Quantum analogues of Markov chains. In *Encyclopedia of Algorithms*. Springer Berlin Heidelberg, 2014. doi:10.1007/978-3-642-27848-8_302-2.

32 Miklos Santha. Quantum walk based search algorithms. In *International Conference on Theory and Applications of Models of Computation*, TAMC'08, pages 31–46, 2008. doi:10.1007/978-3-540-79228-4_3.

33 Mario Szegedy. Quantum speed-up of Markov chain based algorithms. In *45th IEEE Symposium on Foundations of Computer Science*, FOCS'04, pages 32–41, 2004. doi:10.1109/FOCS.2004.53.

34 Avatar Tulsi. Faster quantum walk algorithm for the two dimensional spatial search. *Physical Review A: General Physics*, 78(4):012310, 2008. doi:10.1103/PhysRevA.78.012310.

35 Leslie G. Valiant and Vijay V. Vazirani. NP is as easy as detecting unique solutions. *Theoretical Computer Science*, 47:85–93, 1986. doi:10.1016/0304-3975(86)90135-0.

36 Salvador E. Venegas-Andraca. Quantum walks: A comprehensive review. *Quantum Information Processing*, 11(5):1015–1106, 2012. doi:10.1007/s11128-012-0432-5.