Quantum-walk-based search algorithms with multiple marked vertices

G. A. Bezerra ⬤, P. H. G. Lugão ⬤, and R. Portugal ⬤

National Laboratory of Scientific Computing (LNCC) Petrópolis, Rio de Janeiro 25651-075, Brazil

The quantum walk is a powerful tool to develop quantum algorithms, which usually are based on searching for a vertex in a graph with multiple marked vertices, with Ambainis's quantum algorithm for solving the element distinctness problem being the most shining example. In this work, we address the problem of calculating analytical expressions of the time complexity of finding a marked vertex using quantum-walk-based search algorithms with multiple marked vertices on arbitrary graphs, extending previous analytical methods based on Szegedy's quantum walk, which can be applied only to bipartite graphs. Two examples based on the coined quantum walk on two-dimensional lattices and hypercubes show the details of our method.

I. INTRODUCTION

The discrete-time quantum walk is the quantum counterpart of the classical random walk. In the classical case, the state of a random walker is a probability distribution, and its dynamics is described by a stochastic matrix (acting upon the state), which is obtained from the adjacency matrix of the graph. The adjacency matrix ensures that the random walk obeys locality constraints, which means that if the walker is on a vertex v at time t, the walker will hop to some vertex in the neighborhood of v at time $t + 1$ [1]. In the quantum case, the state of a quantum walker is a L_2 norm-1 vector in a Hilbert space, and its dynamics is described by unitary matrices, as demanded by the laws of quantum mechanics, but not only that, the unitary matrices must be *local*. The locality is defined by some discrete structure, which characterizes a neighborhood for each allowed location for the quantum walker on that discrete structure [2]. Most papers in literature employ graphs and the allowed locations are vertices, edges, arcs, or faces, depending on the quantum walk model [3]. The model is a recipe that provides local unitary operators, and the product of those operators is the evolution operator of the model. The quantum walk is not only an important tool to build quantum algorithms that outperform their classical counterparts [4], but also a versatile toy model useful to simulate and analyze complex physical systems [5,6].

In 2002, Benioff [7] came up with the idea of quantum robots searching a two-dimensional lattice for a specific site, which inspired many researchers to use quantum walks for searching algorithms. In the same year, Shenvi *et al.* described a coined quantum-walk-based search algorithm on hypercubes with a quadratic improvement over a classical random-walk-based algorithm. Ambainis *et al.* described a similar search algorithm on two-dimensional lattices [8], which was improved by Tulsi in 2008 by adding a qubit to the model [9]. Tulsi's modification proved useful for other graphs [10].

The first quantum-walk-based search algorithm (on a specific bipartite graph) with multiple marked vertices was designed by Ambainis [11] in 2003, and, in this case, the searching has a practical application for solving the element distinctness problem. The algorithm time complexity is calculated via a reduction method, which converts the marked set into only one marked vertex in a reduced graph. Ambainis's quantum walk was extended to arbitrary symmetric bipartite graphs with multiple marked vertices by Szegedy [12], who was able to obtain a quadratic improvement for the detection problem, which aims to determine whether there is at least one marked vertex in the graph. The searching problem, which aims to determine the location of at least one marked vertex in the graph, cannot be solved with a quadratic speedup on arbitrary graphs, but can be solved with a quadratic speedup on *bipartite graphs* by using a combination of Szegedy's quantum walk, the phase-estimation algorithm, and interpolated quantum walks [13,14]. In technical terms, Szegedy showed that the quantum hitting time of a quantum walk on a bipartite graph is $\sqrt{h}$, where h is the hitting time of a classical Markov chain on the underlying graph. In Szegedy's model, the underlying graph can be any graph, but the graph on which the quantum walk takes place must be bipartite.

The coined model [15] is a recipe to define quantum walks on graphs by extending the position space with an internal coin space. The dimension of the coin-position Hilbert space is strictly larger than the number of vertices. This coin extension can be understood via graph theory as a modification of the graph itself by inserting for each vertex a clique whose size is equal to the degree of the vertex [16], so that the number of vertices of the extended clique-inserted graph [17] is equal to the dimension of the original coin-position Hilbert space. By inserting cliques, the extended graph is nonbipartite if there is at least one degree-3 vertex in the original graph, for instance, the n-dimensional hypercube with $n > 2$. This means that the results about finding at least one marked vertex on bipartite graphs help little for the coined model.

Many papers have addressed the searching problem with multiple marked vertices using the coined model [18–21], and most of them show negative results by displaying "exceptional configurations" of marked vertices that hinder the coined quantum-walk-based searching algorithms. Reference [19] shows how to eliminate exceptional configuration by using Tulsi's modification on regular graphs [9]. It is not known whether those unwelcome configurations are really exceptional or in fact common in the standard coined model. To answer this kind of question we need analytical methods to address this problem. In this work, we describe an analytical framework for determining the time complexity of discrete-time quantum-walk-based searching algorithms on arbitrary graphs with multiple marked vertices, which can be applied not only to the coined model but also to any discrete-time quantum walk. The standard dynamics of searching algorithms is based on a modification of the underlying evolution operator U of the quantum walk by multiplying U by a unitary operator R, usually called *oracle*, that knows the locations of the marked vertices, so that the new evolution operator U' is $U \cdot R$. Our framework uses two eigenvectors of U' whose eigenvalues are closest to 1 but different from 1, extending a similar method that was successfully used to analyze quantum-walk-based search algorithms on many graphs with *only* one marked vertex [2,8,22].

We provide two examples of our method. We calculate analytically the time complexity of quantum-walk-based search algorithms on two-dimensional lattices and hypercubes with two marked vertices each, using the coined model with the Grover coin. We show that the asymptotic optimal running time and the success probability depend on the relative distance of the marked vertices for the two-dimensional lattice. The speed of the algorithm is slower when the marked vertices are neighbors. For the n-dimensional hypercube, the asymptotic optimal running time is $\pi\sqrt{2^n}/4$ and the success probability is $1/2$, regardless of the locations of the marked vertices. The calculations can be extended to more marked vertices, and numerical experiments show that the quantum-walk-based search algorithm on the hypercube is similar to Grover's algorithm [23] in the sense that the optimal running time t_m for m marked vertices is $t_m = t_1/\sqrt{m}$ with success probability $1/2$ (no dependence on m).

The structure of the paper is as follows. Section II describes the method to determine the time complexity of quantum-walk-based search algorithms on arbitrary graphs with multiple marked vertices, and gives all the details when there are two marked vertices. Section III applies the method to two-dimensional lattices with two marked vertices. Section IV applies the method to hypercubes with two marked vertices. Section V shows how numerical methods can be improved. Section VI presents our conclusions.

II. TIME COMPLEXITY OF SEARCH ALGORITHMS WITH MULTIPLE MARKED VERTICES

Consider a graph Γ, where $V(\Gamma)$ is the set of vertices and $|V(\Gamma)| = N$. Let $\mathcal{H}^N$ be the N-dimensional Hilbert space associated with the graph; that is, the computational basis of $\mathcal{H}^N$ is $\{|v\rangle : v \in V(\Gamma)\}$. Although the dimension of Hilbert space is equal to the number of vertices, the results of this

section can be applied to the coined model, as we show in concrete examples.

Let M be the set of marked vertices. Then, the unitary operator that marks a vertex v is

$$R = I - 2\sum_{v \in M} |v\rangle\langle v|. \qquad (1)$$

The evolution operator U' of a *quantum-walk-based search algorithm* is

$$U' = UR. \qquad (2)$$

The walker starts at an initial state $|\psi(0)\rangle$ and evolves driven by U'; that is, the walker's state after t steps is $|\psi(t)\rangle = (U')^t|\psi(0)\rangle$.

The probability of finding a marked vertex $m \in M$ after t steps is

$$p(t) = \sum_{m \in M} |\langle m|(U')^t|\psi(0)\rangle|^2. \qquad (3)$$

The goal now is to determine the optimal number of steps t_{opt}, which is the one that maximizes $p(t)$. The running time is t_{opt} and the success probability is $p(t_{\text{opt}})$.

Let $|\lambda\rangle$ and $|\lambda'\rangle$ be the eigenvectors of U' that have the eigenvalues as close as possible to 1, but different from 1. The eigenspace spanned by the other eigenvectors will be disregarded, which causes some supposedly small error. Then

$$p(t) = \sum_{m \in M} |e^{i\lambda t}\langle m|\lambda\rangle\langle\lambda|\psi(0)\rangle$$
$$+ e^{i\lambda' t}\langle m|\lambda'\rangle\langle\lambda'|\psi(0)\rangle + \epsilon_m|^2, \qquad (4)$$

where $\epsilon_m = \langle m|U_{\text{tiny}}^t|\psi(0)\rangle$ and U_{tiny} acts nontrivially only on the subspace orthogonal to the plane spanned by $\{|\lambda\rangle, |\lambda'\rangle\}$. Our approach can be applied when $|\epsilon_m|$ is much smaller than the absolute value of the remaining terms in the asymptotic limit (large N). We disregard ϵ_m for now and show applications for which $\lim_{N\to\infty} |\epsilon_m| = 0$.

Let us start by calculating λ and λ'. Suppose that $\{|\psi_k\rangle\}$ is an orthonormal eigenbasis of U and $\exp(i\phi_k)$ the corresponding eigenvalues. Then,

$$\langle m|\lambda\rangle = \sum_k \langle m|\psi_k\rangle\langle\psi_k|\lambda\rangle, \qquad (5)$$

where $m \in M$. Using $\langle\psi_k|U'|\lambda\rangle = \langle\psi_k|UR|\lambda\rangle$ and supposing that $\lambda \neq \phi_k, \forall k$, we obtain

$$\langle\psi_k|\lambda\rangle = (1 + ib_k^\lambda)\sum_{m \in M} \langle\psi_k|m\rangle\langle m|\lambda\rangle, \qquad (6)$$

where

$$b_k^\lambda = \frac{\sin(\lambda - \phi_k)}{1 - \cos(\lambda - \phi_k)}. \qquad (7)$$

Replacing Eq. (6) into Eq. (5), we learn that the $|M|$ vector with entries $\langle m|\lambda\rangle$ is a zero eigenvector of the $|M|$-dimensional Hermitian matrix Λ^λ, where

$$\Lambda_{mm'}^\lambda = \sum_k b_k^\lambda \langle m|\psi_k\rangle\langle\psi_k|m'\rangle. \qquad (8)$$

Then,

$$\det(\Lambda^\lambda) = 0. \qquad (9)$$

We use this equation to find λ and the analog equation $\det(\Lambda^{\lambda'}) = 0$ to find λ'. Terms $\langle m|\lambda\rangle$ and $\langle m|\lambda'\rangle$ are calculated up to their norms using the fact that they are zero eigenvectors of Λ^λ and $\Lambda^{\lambda'}$, respectively. The norms of these zero eigenvectors are calculated using Eq. (6) and the constraint $\sum_k |\langle\psi_k|\lambda\rangle|^2 = 1$. The last missing terms, $\langle\lambda|\psi(0)\rangle$ and $\langle\lambda'|\psi(0)\rangle$, can be calculated using Eq. (6) by assuming that $|\psi(0)\rangle$ is a uniform superposition of the $(+1)$ eigenvectors of U that have nonzero overlap with the target states.

There are cases so that $\lambda = -\lambda'$ and $\langle m|\lambda\rangle\langle\lambda|\psi(0)\rangle = -\langle m|\lambda'\rangle\langle\lambda'|\psi(0)\rangle$ for all m, and Eq. (4) simplifies in the asymptotic limit to

$$p(t) = p_{\mathrm{succ}} \sin^2 \lambda t, \tag{10}$$

where

$$p_{\mathrm{succ}} = 4 \sum_{m \in M} |\langle m|\lambda\rangle|^2 |\langle\lambda|\psi(0)\rangle|^2. \tag{11}$$

In those cases, we know that the optimal running time is $t_{\mathrm{opt}} = \pi/(2\lambda)$ and the success probability is p_{succ}. The time complexity is determined by the asymptotic behavior of $t_{\mathrm{run}} = t_{\mathrm{opt}}/\sqrt{p_{\mathrm{succ}}}$ as a function of the number of vertices because, in the quantum case after using the amplitude amplification method [24], t_{run} is the total running time with success probability $O(1)$.

The results described above can be used not only for analytical calculations but also to speed up numerical methods. We can use Eq. (9) to find λ numerically for a specific configuration of marked vertices, and then, by plotting $t_{\mathrm{opt}} = \pi/(2\lambda)$ as a function of N, we can estimate the complexity of the running time. On the other hand, Eq. (11) can be used to generate a plot of the success probability as a function of N. The combination of those plots are used to determine the time complexity of the search algorithm. An example of this numerical approach is shown in Sec. V.

Case $|M| = 2$

Suppose that $M = \{m_0, m_1\}$. Equation (9) implies that

$$\sum_{kk'} b_k^\lambda b_{k'}^\lambda c_{kk'} = 0, \tag{12}$$

where

$$c_{kk'} = \psi_k(m_0)\psi_{k'}(m_1)[\psi_k^*(m_0)\psi_{k'}^*(m_1) - \psi_k^*(m_1)\psi_{k'}^*(m_0)]. \tag{13}$$

Suppose that $\lambda \ll \phi_{\min}$ when $N \gg 1$, where $\phi_{\min}$ is the smallest positive value of ϕ_k. We will check the validity of those assumptions in specific applications. Let us split the sum (12) into four parts

$$\sum_{kk'} = \sum_{\substack{\phi_k = 0 \\ \phi_{k'} = 0}} + \sum_{\substack{\phi_k = 0 \\ \phi_{k'} \neq 0}} + \sum_{\substack{\phi_k \neq 0 \\ \phi_{k'} = 0}} + \sum_{\substack{\phi_k \neq 0 \\ \phi_{k'} \neq 0}}, \tag{14}$$

corresponding to the sum of terms such that $\phi_k = 0$ or $\phi_k \neq 0$. Using

$$\frac{\sin\lambda}{1 - \cos\lambda} = \frac{2}{\lambda} + O(\lambda) \tag{15}$$

and, if $\phi_k \neq 0$,

$$\frac{\sin(\lambda - \phi_k)}{1 - \cos(\lambda - \phi_k)} = a_k \sin\phi_k + a_k\lambda + O(\lambda^2), \tag{16}$$

where

$$a_k = \frac{1}{\cos\phi_k - 1}, \tag{17}$$

we obtain

$$\frac{A}{\lambda^2} + \frac{B}{\lambda} + C + D\lambda + E\lambda^2 = O(\lambda^3), \tag{18}$$

where

$$A = 4 \sum_{\substack{\phi_k = 0 \\ \phi_{k'} = 0}} c_{kk'}, \tag{19}$$

$$B = 2 \sum_{\substack{\phi_k = 0 \\ \phi_{k'} \neq 0}} a_{k'} c_{kk'} \sin\phi_{k'} + 2 \sum_{\substack{\phi_k \neq 0 \\ \phi_{k'} = 0}} a_k c_{kk'} \sin\phi_k, \tag{20}$$

$$C = 2 \sum_{\substack{\phi_k = 0 \\ \phi_{k'} \neq 0}} a_{k'} c_{kk'} + 2 \sum_{\substack{\phi_k \neq 0 \\ \phi_{k'} = 0}} a_k c_{kk'}$$
$$+ \sum_{\substack{\phi_k \neq 0 \\ \phi_{k'} \neq 0}} a_k a_{k'} c_{kk'} \sin\phi_k \sin\phi_{k'}, \tag{21}$$

$$D = \sum_{\substack{\phi_k \neq 0 \\ \phi_{k'} \neq 0}} a_k a_{k'} c_{kk'} (\sin\phi_k + \sin\phi_{k'}), \tag{22}$$

$$E = \sum_{\substack{\phi_k \neq 0 \\ \phi_{k'} \neq 0}} a_k a_{k'} c_{kk'}. \tag{23}$$

Using that $\langle m|\lambda\rangle$ is a zero eigenvector of Λ^λ and re-scaling $|\lambda\rangle$ by a global phase, we obtain

$$\langle m_0|\lambda\rangle = \alpha \Lambda_{m_0 m_1}^\lambda, \tag{24}$$

$$\langle m_1|\lambda\rangle = -\alpha \Lambda_{m_0 m_0}^\lambda, \tag{25}$$

where α is a positive constant, which can be calculated using $1 = \sum_k |\langle\psi_k|\lambda\rangle|^2$ and Eq. (6):

$$\frac{1}{\alpha^2} = \left(1 + \Lambda'_{m_1 m_1}\right)\left(\Lambda_{m_0 m_0}^\lambda\right)^2 - 2\Lambda_{m_0 m_0}^\lambda \mathrm{Re}\left\{\Lambda'_{m_0 m_1} \Lambda_{m_0 m_1}^\lambda\right\}$$
$$+ \left(1 + \Lambda'_{m_0 m_0}\right)\left|\Lambda_{m_0 m_1}^\lambda\right|^2, \tag{26}$$

where Re is the real part operator,

$$\Lambda_{mm'}^\lambda = \sum_k b_k^\lambda \psi_k(m)\psi_k^*(m'), \tag{27}$$

and

$$\Lambda'_{mm'} = \sum_k \left(b_k^\lambda\right)^2 \psi_k(m)\psi_k^*(m'). \tag{28}$$

Note that, because we have already calculated λ, we can now calculate explicitly $\Lambda_{mm'}^\lambda$ and $\Lambda'_{mm'}$ if we know the spectral decomposition of U.

III. SEARCHING TWO-DIMENSIONAL LATTICES WITH TWO MARKED VERTICES

As an application, we calculate the time complexity of a quantum-walk-based search algorithm on a $\sqrt{N} \times \sqrt{N}$ square lattice assuming that the lattice has cyclic boundary conditions with exactly two marked vertices. The evolution operator of a coined quantum walk with no marked vertex is

$$U = S(G \otimes I), \tag{29}$$

where G is the Grover coin and S is the flip-flop shift operator given by

$$S|i, j\rangle|x, y\rangle = |1 - i, 1 - j\rangle|x + (-1)^i, y + (-1)^j\rangle, \tag{30}$$

where i and j are bits and the arithmetic in the second register is performed modulo $\sqrt{N}$. To search the lattice, the modified evolution operator is $U' = UR'$, where

$$R' = I - 2|0'\rangle\langle 0'| - 2|1'\rangle\langle 1'| \tag{31}$$

and

$$|0'\rangle = |d_c\rangle|0, 0\rangle, \tag{32}$$

$$|1'\rangle = |d_c\rangle|x_0, y_0\rangle, \tag{33}$$

where one marked vertex is chosen at $m_0 = (0, 0)$ without loss of generality and the second marked vertex is chosen at position $m_1 = (x_0, y_0)$ so that $m_0 \neq m_1$. Vector $|d_c\rangle$ is the normalized uniform superposition of the computational basis of the coined space. Note that here the Hilbert space is larger because it has been augmented by the coin space, and then all formulas of the previous section must be extended by substituting $|d_c\rangle|m\rangle$ for $|m\rangle$. The initial state $|\psi(0)\rangle$ is the uniform superposition of all states of the computational basis, that is,

$$|\psi(0)\rangle = |d_c\rangle|d_p\rangle, \tag{34}$$

where $|d_p\rangle$ is the normalized uniform superposition of computational basis of the position space. State $|\psi(0)\rangle$ can be generated in $O(\sqrt{N})$ steps.

The eigenvectors of U that have nonzero overlap with the marked vertices are $|\psi(0)\rangle$ with eigenvalue 1 and $|v_{k\ell}^{\pm\theta}\rangle|\tilde{k}, \tilde{\ell}\rangle$ for $(k, \ell) \neq (0, 0)$ with eigenvalues $e^{\pm i\theta_{kl}}$, where [2,25]

$$|\tilde{k}, \tilde{\ell}\rangle = \frac{1}{\sqrt{N}} \sum_{x,y=0}^{\sqrt{N}-1} \omega^{xk+y\ell}|x, y\rangle \tag{35}$$

and

$$|v_{k\ell}^{\pm\theta}\rangle = \frac{\pm i}{2\sqrt{2}\sin\theta_{k\ell}} \begin{bmatrix} e^{\mp i\theta_{k\ell}} - \omega^k \\ e^{\mp i\theta_{k\ell}} - \omega^{-k} \\ e^{\mp i\theta_{k\ell}} - \omega^\ell \\ e^{\mp i\theta_{k\ell}} - \omega^{-\ell} \end{bmatrix}, \tag{36}$$

and

$$\cos\theta_{k\ell} = \frac{1}{2}\left(\cos\frac{2\pi k}{\sqrt{N}} + \cos\frac{2\pi\ell}{\sqrt{N}}\right), \tag{37}$$

where $\omega = \exp(2\pi i/\sqrt{N})$.

Using this list of eigenvectors and $\langle d_c|v_{k\ell}^{\pm\theta}\rangle = 1/\sqrt{2}$, Eq. (13) [with the modification $\psi(m) \to (\langle d_c|\langle m|)|\psi\rangle$ and

$k \to j, k, \ell$, where j represents the coin value] reduces to

$$c_{k\ell;k'\ell'} = \frac{1}{\epsilon_{k\ell}\epsilon_{k'\ell'}N^2}(1 - \omega^{(k'-k)x_0+(\ell'-\ell)y_0}),$$

where $\epsilon_{k\ell} = 1$ if $k = \ell = 0$ and $\epsilon_{k\ell} = 2$ otherwise. Index j runs from zero to 3 in the coin space, and can be readily simplified. Then, we obtain $A = B = D = 0$ and

$$C = -\frac{4}{N^2} \sum_{\substack{k, \ell = 0 \\ (k, \ell) \neq (0, 0)}}^{\sqrt{N}-1} \frac{1 - \cos\frac{2\pi(kx_0+\ell y_0)}{\sqrt{N}}}{1 - \cos\theta_{k\ell}}, \tag{38}$$

$$E = \frac{1}{N^2} \sum_{\substack{k, \ell = 0 \\ (k, \ell) \neq (0, 0)}}^{\sqrt{N}-1} \sum_{\substack{k', \ell' = 0 \\ (k', \ell') \neq (0, 0)}}^{\sqrt{N}-1} \frac{1 - \cos\frac{2\pi(k'-k)x_0+2\pi(\ell'-\ell)y_0}{\sqrt{N}}}{(1 - \cos\theta_{k\ell})(1 - \cos\theta_{k'\ell'})}. \tag{39}$$

Note that the imaginary part of E is zero because of symmetry properties. The expression for λ reduces to

$$\lambda = \sqrt{\frac{-C}{E}}. \tag{40}$$

In order to proceed with the calculations, we introduce the constant $c \in \mathbb{R}$ such that

$$\sum_{\substack{k, \ell = 0 \\ (k, \ell) \neq (0, 0)}}^{\sqrt{N}-1} \frac{1}{1 - \cos\theta_{k\ell}} = cN \ln N + O(N), \tag{41}$$

where c is bounded by $2/\pi^2 \leqslant c \leqslant 1$. Numerical calculations show that $c \approx 0.32$. The asymptotic behavior of λ using two representative pairs of marked elements, namely $[(0, 0), (1, 0)]$ and $[(0, 0), (\sqrt{N}/2, \sqrt{N}/2)]$, is obtained in Appendix A and is given by

$$\lambda = \begin{cases} \frac{\sqrt{2}}{\sqrt{c}\sqrt{N \ln N}}, & (x_0, y_0) = (1, 0), \\ \frac{2}{\sqrt{c}\sqrt{N \ln N}}, & (x_0, y_0) = (\sqrt{N}/2, \sqrt{N}/2). \end{cases} \tag{42}$$

Note that $\lambda \ll \theta_{k\ell}$ when $N \gg 1$ because $\theta_{k\ell} = O(1/\sqrt{N})$.

Case $(x_0, y_0) = (1, 0)$

Using Eqs. (15), (16), (27), and lower order terms in the asymptotic expansion of λ, we obtain

$$\Lambda_{m_0 m_0}^\lambda = -\Lambda_{m_0 m_1}^\lambda = -\frac{1}{\sqrt{2c}\sqrt{N \ln N}} + O\left(\frac{1}{\sqrt{N} \ln^{\frac{3}{2}} N}\right). \tag{43}$$

Using Eq. (28), we obtain

$$\Lambda_{m_0 m_0}' = \Lambda_{m_0 m_1}' = \Lambda_{m_1 m_1}' = 4c \ln N + O(1). \tag{44}$$

Replacing those results in Eq. (26), we have

$$\alpha = \frac{\sqrt{N}}{2\sqrt{2}} \tag{45}$$

and

$$\langle 0'|\lambda\rangle = \langle 1'|\lambda\rangle = \frac{1}{4\sqrt{c}\sqrt{\ln N}}. \tag{46}$$

Taking the complex conjugate of Eq. (6) and replacing the above results, we obtain

$$\langle \lambda | \psi(0) \rangle = -\frac{i}{\sqrt{2}} + O\left(\frac{1}{\sqrt{N \ln N}}\right). \quad (47)$$

The equation above can be used to show that the terms ϵ_m in Eq. (4) tend to zero when $N \rightarrow \infty$. Using Eqs. (10) and (11), we have asymptotically

$$p(t) = \frac{1}{4c \ln N} \sin^2\left(\frac{\sqrt{2}\, t}{\sqrt{c}\sqrt{N \ln N}}\right). \quad (48)$$

Taking the running time as the optimal t, we get

$$t_{\text{opt}} = \frac{\pi \sqrt{c}\sqrt{N \ln N}}{2\sqrt{2}} \quad (49)$$

and

$$p_{\text{succ}} = \frac{1}{4c \ln N}. \quad (50)$$

Case $(x_0, y_0) = (\sqrt{N}/2, \sqrt{N}/2)$

In this subsection, we assume that $\sqrt{N}$ is even. Using Eqs. (15), (16), and (27), in the asymptotic limit we obtain

$$\Lambda^{\lambda}_{m_0 m_0} = -\Lambda^{\lambda}_{m_0 m_0} = \sqrt{c}\,\frac{\sqrt{\ln N}}{\sqrt{N}} + O\left(\frac{1}{\sqrt{N \ln N}}\right). \quad (51)$$

Using Eq. (28), we obtain

$$\Lambda'_{m_0 m_0} = \Lambda'_{m_1 m_1} = 3c \ln N + O\left(\frac{1}{\ln N}\right), \quad (52)$$

$$\Lambda'_{m_0 m_1} = c \ln N + O(1). \quad (53)$$

Replacing those results in Eq. (26), we have

$$\alpha = \frac{\sqrt{N}}{2c\sqrt{2}\ln N} \quad (54)$$

and

$$\langle 0' | \lambda \rangle = \langle 1' | \lambda \rangle = \frac{1}{2\sqrt{2c}\sqrt{\ln N}}. \quad (55)$$

Taking the complex conjugate of Eq. (6) and replacing the above results, we obtain

$$\langle \lambda | \psi(0) \rangle = -\frac{i}{\sqrt{2}} + O\left(\frac{1}{\sqrt{N \ln N}}\right). \quad (56)$$

Using Eqs. (10) and (11), we have asymptotically

$$p(t) = \frac{1}{2c \ln N} \sin^2\left(\frac{2\, t}{\sqrt{c}\sqrt{N \ln N}}\right). \quad (57)$$

Taking the running time as the optimal t, we get

$$t_{\text{opt}} = \frac{\pi \sqrt{c}\sqrt{N \ln N}}{4} \quad (58)$$

and

$$p_{\text{succ}} = \frac{1}{2c \ln N}. \quad (59)$$

IV. SEARCHING HYPERCUBES WITH TWO MARKED VERTICES

As a second application, we calculate the time complexity of a quantum-walk-based search algorithm on a hypercube with exactly two marked vertices. A hypercube has $N = 2^n$ vertices whose labels are binary vectors $\vec{v}$. The decimal representation of $\vec{v}$ is in the range $0 \leqslant \vec{v} < N$. The evolution operator of a coined quantum walk with no marked vertex is

$$U = S(G \otimes I_N), \quad (60)$$

where $G \in \mathcal{H}^n$ is the Grover coin and $S \in \mathcal{H}^n \otimes \mathcal{H}^N$ is the flip-flop shift operator given by

$$S|a\rangle |\vec{v}\rangle = |a\rangle |\vec{v} \oplus \vec{e}_a\rangle, \quad (61)$$

where $1 \leqslant a \leqslant n$ and $\vec{e}_a$ is the binary n-tuple with all entries zero except the ath entry, whose value is 1. To search the n-dimensional hypercube, the modified evolution operator is $U' = UR'$, where

$$R' = I - 2|0'\rangle\langle 0'| - 2|1'\rangle\langle 1'| \quad (62)$$

and

$$|0'\rangle = |d_c\rangle |\vec{0}\rangle, \quad (63)$$

$$|1'\rangle = |d_c\rangle |\vec{v}_0\rangle, \quad (64)$$

where one marked vertex is chosen at $(0, \ldots, 0)$ without loss of generality and the second marked vertex is chosen at an arbitrary position $\vec{v}_0$ so that $\vec{v}_0 \neq (0, \ldots, 0)$. Vector $|d_c\rangle$ is the normalized uniform superposition of the computational basis of the coined space. Note that the Hilbert space has been augmented by the coin space, whose basis is $\{|1\rangle, \ldots, |n\rangle\}$. The initial state $|\psi(0)\rangle$ is the uniform superposition of all states of the computational basis, that is,

$$|\psi(0)\rangle = |d_c\rangle \otimes |d_p\rangle = \frac{1}{\sqrt{n}} \sum_{a=1}^{n} |a\rangle \otimes \frac{1}{\sqrt{N}} \sum_{\vec{v}=0}^{N-1} |\vec{v}\rangle. \quad (65)$$

State $|\psi(0)\rangle$ can be generated in $O(\sqrt{N})$ steps using local operators.

The results of Sec. II can be readily employed as soon as we calculate A to E given by Eqs. (19) to (23). The relevant eigenvectors of U are $|\psi(0)\rangle$, $|d_c\rangle|\beta_{(1,\ldots,1)}\rangle$, and $|\alpha^{\vec{k}}_{\pm}\rangle|\beta_{\vec{k}}\rangle$ (see [2,26]), where $|\alpha^{\vec{k}}_{-}\rangle = |\alpha^{\vec{k}}_{+}\rangle^*$,

$$|\alpha^{\vec{k}}_{+}\rangle = \frac{e^{i\theta}}{\sqrt{2}} \sum_{a=1}^{n} \left(\frac{k_a}{\sqrt{k}} - i\frac{1-k_a}{\sqrt{n-k}}\right)|a\rangle, \quad (66)$$

for $0 < k < n$, where k is the Hamming weight of $\vec{k}$, k_a is the ath entry of $\vec{k}$, and $\cos\theta = \sqrt{k/n}$; and

$$|\beta_{\vec{k}}\rangle \equiv \frac{1}{\sqrt{N}} \sum_{\vec{v}=0}^{2^n-1} (-1)^{\vec{k} \cdot \vec{v}} |\vec{v}\rangle, \quad (67)$$

where $\vec{k} \cdot \vec{v} = \sum_j \vec{k}_j \vec{v}_j \mod 2$. The corresponding eigenvalues are 1, -1, and $e^{\pm i\omega_k}$. A (nonorthogonal) eigenbasis of U has been summarized in Table I, but we use only the eigenvectors that have a nonzero overlap with the marked elements.

TABLE I. Eigenvalues $\exp(\pm i\omega_k)$ and eigenvectors $|\alpha_a^{\vec{k}}\rangle \otimes |\beta_{\vec{k}}\rangle$ of U, where k is the Hamming weight of $\vec{k}$, $\cos\omega_k = 1 - 2k/n$, $q(x) = \min\{a \mid k_a = x\}$, and $|\alpha_+^{\vec{k}}\rangle$ is given by Eq. (66).

Hamming wgt.	Index a	Eigenval.	$\vert\alpha_a^{\vec{k}}\rangle$	Multiplicity
$k = 0$	$a = 1$	1	$\sum_{b=1}^{n} \vert b\rangle/\sqrt{n}$	1
	$a \in [2, n]$	-1	$(\vert 1\rangle - \vert a\rangle)/\sqrt{2}$	$n - 1$
$1 \leqslant k \leqslant n-1$	$a = q(0)$	$e^{i\omega_k}$	$\vert\alpha_+^{\vec{k}}\rangle$	1
	$a = q(1)$	$e^{-i\omega_k}$	$\vert\alpha_+^{\vec{k}}\rangle^*$	1
	$\{a \mid k_a = 1\} \setminus \{q(1)\}$	1	$[\vert q(1)\rangle - \vert a\rangle]/\sqrt{2}$	$k - 1$
	$\{a \mid k_a = 0\} \setminus \{q(0)\}$	-1	$[\vert q(0)\rangle - \vert a\rangle]/\sqrt{2}$	$n - k - 1$
$k = n$	$a = 1$	-1	$\sum_{b=1}^{n} \vert b\rangle/\sqrt{n}$	1
	$a \in [2, n]$	1	$(\vert 1\rangle - \vert a\rangle)/\sqrt{2}$	$n - 1$

Using

$$\langle d_c | \alpha_+^{\vec{k}} \rangle = \langle d_c | \alpha_-^{\vec{k}} \rangle = \frac{1}{\sqrt{2}} \qquad (68)$$

for $0 < k < n$, we obtain $A = B = D = 0$,

$$C = -\frac{4n}{N^2} \sum_{\substack{\vec{k}=1 \\ \vec{k}\cdot\vec{v}_0 \text{ odd}}}^{N-1} \frac{1}{k}, \qquad (69)$$

$$E = \frac{n^2}{2N^2} \sum_{\substack{\vec{k},\vec{k}'=1 \\ (\vec{k}\oplus\vec{k}')\cdot\vec{v}_0 \text{ odd}}}^{N-1} \frac{1}{kk'}. \qquad (70)$$

The asymptotic behavior of variables C and E is obtained in Appendix B, and is given by

$$C = -\frac{4}{N} + O\left(\frac{1}{n}\right), \qquad (71)$$

$$E = 1 + O\left(\frac{1}{n}\right), \qquad (72)$$

for any location of the second marked vertex $\vec{v}_0$.

Using Eq. (18) in the asymptotic limit, we obtain

$$\lambda = \frac{2}{\sqrt{N}}. \qquad (73)$$

Note that $\lambda \ll \omega_k$ when $N \gg 1$ because $\omega_k = O(1/\sqrt{n})$. Using Eqs. (15), (16), and (27), in the asymptotic limit we obtain

$$\Lambda_{m_0 m_0}^{\lambda} = \Lambda_{m_1 m_1}^{\lambda} = -\Lambda_{m_0 m_1}^{\lambda} = -\frac{1}{\sqrt{N}} + O\left(\frac{1}{n}\right). \qquad (74)$$

Using Eq. (28), we obtain

$$\Lambda'_{m_0 m_0} = \Lambda'_{m_1 m_1} = \Lambda'_{m_0 m_1} + 1 = 2 + O\left(\frac{1}{n}\right). \qquad (75)$$

Replacing those results in Eq. (26), we obtain $\alpha = \sqrt{N/8}$ and $\langle 0' | \lambda \rangle = \langle 1' | \lambda \rangle = 1/\sqrt{8}$ asymptotically. Using Eq. (6), we obtain

$$\langle \psi(0) | \lambda \rangle = \frac{i}{\sqrt{2}} + O\left(\frac{1}{n}\right). \qquad (76)$$

The equation above can be used to show that the terms ϵ_m in Eq. (4) tend to zero when $N \to \infty$. Using Eqs. (10) and (11), we have asymptotically

$$p(t) = \frac{1}{2} \sin^2\left(\frac{2t}{\sqrt{N}}\right). \qquad (77)$$

Taking the running time as the optimal t, that is,

$$t_{\text{opt}} = \frac{\pi\sqrt{N}}{4}, \qquad (78)$$

we obtain asymptotically

$$p_{\text{succ}} = \frac{1}{2}. \qquad (79)$$

V. NUMERICAL METHODS

In this section we show how to enhance numerical methods that estimate the time complexity of quantum-walk-based search algorithms with multiple marked vertices using a representative example. We consider a Grover walk on hypercubes, as described in Sec. IV, but now we take an arbitrary number of marked vertices, and run Python simulations on an ordinary laptop. By using Eq. (9), we determine λ numerically for an increasing number of marked vertices. Figure 1 shows $t_{\text{opt}} = \pi/(2\lambda)$ as a function of n for $|M|$ equal to 3, 9, and
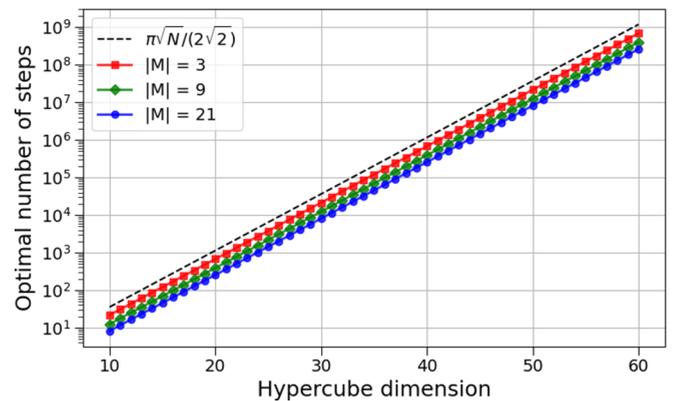
FIG. 1. Plot of the optimal number of steps t_{opt} as a function of the hypercube dimension n for an increasing number of marked vertices $|M|$, whose locations are chosen at random.
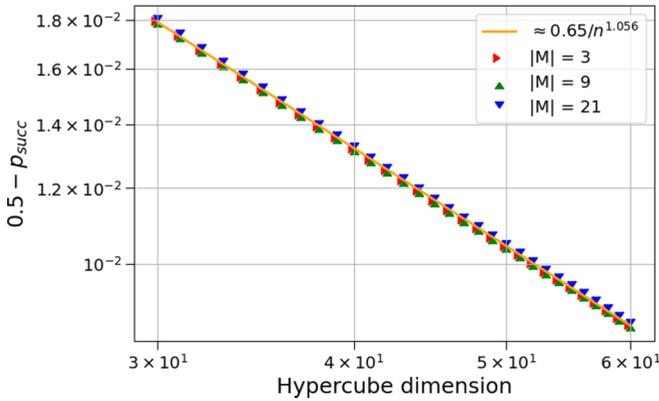
FIG. 2. Plot of $(0.5 - p_{\text{succ}})$ as a function of the hypercube dimension n for the same number of marked vertices of Fig. 1, where p_{succ} is the success probability. The axes are in log-log scale and the points are the tips of the triangles.

21. The plot shows that t_{opt} scales as $\sqrt{N}$ and, by rescaling N into $N/|M|$, all lines merge into the dashed line $\pi\sqrt{N}/(2\sqrt{2})$. This shows that the optimal number of steps right before measurement is the one given by Eq. (80) below. These results do not depend on the locations of the marked vertices.

The next step is the analysis of the success probability p_{succ} given by Eq. (11). Figure 2 shows $(0.5 - p_{\text{succ}})$ as a function of n for the same values of $|M|$ of Fig. 1, and $0.65/n^{1.056}$ as a function of n, which is a straight line in log-log scale obtained by curve fitting. We have eliminated values corresponding to $n < 30$ because the high order terms of the asymptotic expansion of p_{succ} play a relevant role for those values and cannot be fitted into a straight line. These numerical results show that the asymptotic success probability is $1/2$. These results do not depend on the locations of the marked vertices.

Without using Eq. (9), the only way to simulate the time evolution of the quantum walk requires the implementation of the coin and shift operators, which uses exponential resources as a function of n, and cannot be performed in the range of Fig. 1 even in the largest supercomputers. Our numerical results support the following conjecture.

Conjecture. The asymptotic optimal running time for the Grover-walk-based search algorithm on the hypercube with $|M|$ marked vertices is

$$t_{\text{opt}} = \frac{\pi}{2\sqrt{2}}\sqrt{\frac{N}{|M|}} \qquad (80)$$

and the asymptotic success probability is $1/2$.

VI. CONCLUSIONS

We have developed an analytical method to calculate the time complexity of quantum-walk-based search algorithms with multiple marked vertices. The method relies on two eigenvectors of the evolution operator associated with the eigenvalues that are closest to 1 but different from 1. The number of steps is given by $\pi/(2\lambda)$ in the simplest case when the evolution operator has real entries, where $\pm\lambda$ are the phases of the eigenvalues $\exp(\pm i\lambda)$. Usually, the success probability decreases as a function of the number of vertices, with the

hypercube and the complete graph being notable exceptions. This method employs some hypotheses in order to proceed with the analytical calculations that must be checked on specific applications, and if the hypotheses are not confirmed, it means that the method cannot be used. In fact, our method may fail if the time complexity depends on all eigenvectors of the evolution operator. The method can also be used to speed up numerical analysis of search algorithms when the graph structure is too complex for an analytical approach.

We have applied our method to the Grover walk on the two-dimensional lattice and hypercube both with N vertices, and we have shown that the optimal running time is $O(\sqrt{N \ln N})$ with success probability $O(1/\ln N)$ for the two-dimensional lattice and $O(\sqrt{N})$ with success probability $O(1)$ for the hypercube when they have two marked vertices. Since we have obtained the exact asymptotic limits for the running time and the success probability, we can draw further conclusions. When we compare our results with the corresponding ones for the two-dimensional lattice with one marked vertex, the behavior of the algorithm depends on the distance of the marked vertices. If we take them as far apart as possible, the success probability for the two-marked case is the same as the one-marked case but the running time is shorter by a factor of $\sqrt{2}$. If we take the marked vertices as close as possible, the running time for the two-marked case is the same as the one-marked case but the success probability is smaller by a factor of 2. Note that the presence of an extra marked vertex in the first case makes the searching easier and in the latter case makes it worse, different from what is usually expected.

When we compare our results with the corresponding ones for the hypercube with one marked vertex, the success probability is the same but the optimal number of steps for the two-marked case is smaller by a factor of $\sqrt{2}$ for arbitrary locations of the marked vertices. Our numerical calculations show that those results can be extended for more marked vertices regardless of their locations.

It would be interesting to apply our method to other quantum-walk models, such as the staggered model [27], and to analyze whether it can help to find exceptional configurations.

APPENDIX A: ASYMPTOTIC EXPRESSIONS FOR THE TWO-DIMENSIONAL LATTICE

In this Appendix we obtain simpler expressions for C and E, and asymptotic expressions for λ. Let us define the sums S_1 and S_2 as

$$S_1 = \sum_{\substack{k,\ell=0 \\ (k,\ell)\neq(0,0)}}^{\sqrt{N}-1} \frac{1}{1-\cos\theta_{k\ell}}, \qquad (A1)$$

$$S_2 = \sum_{\substack{k,\ell=0 \\ (k,\ell)\neq(0,0)}}^{\sqrt{N}-1} \frac{\sin^2\frac{\pi(kx_0+\ell y_0)}{\sqrt{N}}}{\sin^2\frac{\pi k}{\sqrt{N}}+\sin^2\frac{\pi\ell}{\sqrt{N}}}. \qquad (A2)$$

S_1 has the following bounds [2,8]:

$$\frac{2}{\pi^2} N \ln N \leqslant S_1 \leqslant N \ln N \qquad (A3)$$

up to $O(N)$ terms.

Using Eq. (38) and the identity $\cos 2a = 1 - 2 \sin^2 a$, we obtain

$$C = -\frac{8S_2}{N^2}. \qquad (A4)$$

Using Eq. (39) and the antisymmetry of the sine function, the expression for E can be simplified to

$$E = \frac{4S_2}{N^2}(S_1 - S_2). \qquad (A5)$$

Using Eq. (40), we obtain

$$\lambda = \frac{\sqrt{2}}{\sqrt{S_1 - S2}}. \qquad (A6)$$

Let us proceed with two kinds of pairs of marked points.

Case $(x_0, y_0) = (1, 0)$

If $(x_0, y_0) = (1, 0)$, then

$$S_2 = \sum_{\substack{k, \ell = 0 \\ (k, \ell) \neq (0, 0)}}^{\sqrt{N}-1} \frac{\sin^2 \frac{\pi k}{\sqrt{N}}}{\sin^2 \frac{\pi k}{\sqrt{N}} + \sin^2 \frac{\pi \ell}{\sqrt{N}}}. \qquad (A7)$$

Note that if we interchange k and ℓ inside the sum, we obtain the same result, which can be used to obtain

$$S_2 = \frac{N - 1}{2}. \qquad (A8)$$

Using Eq. (41), we obtain

$$\lambda = \frac{\sqrt{2}}{\sqrt{cN \ln N + O(N)}}. \qquad (A9)$$

Case $(x_0, y_0) = (\sqrt{N}/2, \sqrt{N}/2)$

In this case, we assume that $\sqrt{N}$ is even. Replacing $(x_0, y_0) = (\sqrt{N}/2, \sqrt{N}/2)$ into $(S_1 - S_2)$, we obtain

$$S_1 - S_2 = \frac{1}{2} \sum_{\substack{k, \ell = 0 \\ (k, \ell) \neq (0, 0)}}^{\sqrt{N}-1} \frac{1 + (-1)^{k+l}}{\sin^2 \frac{\pi k}{\sqrt{N}} + \sin^2 \frac{\pi \ell}{\sqrt{N}}}. \qquad (A10)$$

Using $4k^2/\pi^2 \leqslant \sin^2 k \leqslant k^2$, for $-\pi/2 \leqslant k \leqslant \pi/2$ and $(k + \ell)^2/2 \leqslant k^2 + \ell^2 \leqslant (k + \ell)^2$, we obtain $S_1 - S_2 = cNS_3 + O(N)$, where [28]

$$S_3 = \sum_{\substack{k, \ell = 0 \\ (k, \ell) \neq (0, 0)}}^{\sqrt{N}/2} \frac{1 + (-1)^{k+l}}{(k + \ell)^2} = \frac{\ln N}{2} + O(1). \qquad (A11)$$

Using Eq. (41), we obtain

$$\lambda = \frac{2}{\sqrt{cN \ln N + O(N)}}. \qquad (A12)$$

APPENDIX B: ASYMPTOTIC EXPRESSIONS FOR THE HYPERCUBE

In this Appendix we obtain asymptotic expressions for C and E given by Eqs. (69) and (70), respectively. Let us define S_{odd} and S_{even} as

$$S_{\text{odd}} = \frac{n}{N} \sum_{\substack{\vec{k} = 1 \\ \vec{k} \cdot \vec{v}_0 \text{ odd}}}^{N-1} \frac{1}{|\vec{k}|},$$

$$S_{\text{even}} = \frac{n}{N} \sum_{\substack{\vec{k} = 1 \\ \vec{k} \cdot \vec{v}_0 \text{ even}}}^{N-1} \frac{1}{|\vec{k}|},$$

where $|\vec{k}|$ is the Hamming weight of $\vec{k}$ and $N = 2^n$. Using Eq. (69), we have that

$$C = -\frac{4 S_{\text{odd}}}{N}.$$

Using Eq. (70) and that $(\vec{k} \oplus \vec{k}') \cdot \vec{v}_0$ is odd in two cases—(1) $\vec{k} \cdot \vec{v}_0$ is even and $\vec{k}' \cdot \vec{v}_0$ is odd and (2) $\vec{k} \cdot \vec{v}_0$ is odd and $\vec{k}' \cdot \vec{v}_0$ is even—we obtain

$$E = S_{\text{odd}} S_{\text{even}}.$$

Now we use the following lemmas to show that asymptotically $S_{\text{odd}} = S_{\text{even}} = 1$ for any $\vec{v}_0 \neq (0, \ldots, 0)$.

Lemma 1. Let n be a positive integer. Then

$$\frac{n}{2^n} \sum_{k=1}^{n} \frac{1}{k} \binom{n}{k} = 2 \sum_{k=0}^{\infty} \frac{a_k}{n^k} - \frac{n}{2^n}\left(H_n + \frac{2}{n}\right),$$

where a_k is the kth ordered Bell number (Fubini numbers [29], where $a_0 = a_1 = 1$, $a_2 = 3, \ldots$), and H_n is the nth harmonic number.

Lemma 2. Let n be a positive integer, $\vec{v}$ be an n-bit vector, and v be the Hamming weight of $\vec{v}$. Then

$$\sum_{\vec{k}=1}^{2^n-1} \frac{(-1)^{\vec{k} \cdot \vec{v}}}{|\vec{k}|} = \binom{n}{v}^{-1} \sum_{k=1}^{n-v} \frac{1}{k} \binom{n}{v+k} - [\psi(v + 1) + \gamma],$$

where $|\vec{k}|$ is the Hamming weight of $\vec{k}$, ψ is the digamma function, γ is Euler's constant, and the left-hand sum runs over all n-bit vectors different from the null vector.

Using

$$\sum_{\vec{k}=1}^{N-1} \frac{1}{|\vec{k}|} = \sum_{k=1}^{n} \frac{1}{k} \binom{n}{k},$$

we obtain

$$S_{\text{even}} + S_{\text{odd}} = \frac{n}{N} \sum_{k=1}^{n} \frac{1}{k} \binom{n}{k},$$

$$S_{\text{even}} - S_{\text{odd}} = \frac{n}{N} \sum_{\vec{k}=1}^{N-1} \frac{(-1)^{\vec{k} \cdot \vec{v}_0}}{|\vec{k}|}.$$

Using Lemma 1, we conclude that

$$S_{\text{even}} + S_{\text{odd}} = 2 + O\left(\frac{1}{n}\right).$$

To find the asymptotic expression of $S_{\text{even}} - S_{\text{odd}}$, we split the analysis in the following cases.

Case 1: v fixed

If the Hamming weight v of $\vec{v}_0$ is fixed, that is, v does not depend on n, we use the inequality

$$\sum_{k=1}^{n-v} \frac{1}{k}\binom{n}{v+k} \leqslant \sum_{k=1}^{n-v}\binom{n}{v+k} = 2^n - \sum_{k=1}^{v}\binom{n}{k} - 1$$

and Lemma 2 to show that

$$S_{\text{even}} - S_{\text{odd}} = O\!\left(\frac{1}{n}\right),$$

if $v > 1$. We obtain the same result for the case $v = 1$ by using the inequality

$$\sum_{k=1}^{n-1} \frac{1}{k}\binom{n}{k+1} \leqslant \frac{3}{n+1}\sum_{k=1}^{n-1}\binom{n+1}{k+2}$$
$$= \frac{3}{n+1}\left(2\,2^n - \frac{n(n+1)}{2} - 2\right).$$

Case 2: v depends on n

Here we assume that v depends on n and $\lim_{n\to\infty} v = \infty$, where $v = |\vec{v}_0|$. Using the inequality

$$\binom{n}{v+k} \leqslant \binom{n}{v}\binom{n-v}{k},$$

we have

$$\binom{n}{v}^{-1}\sum_{k=1}^{n-v}\frac{1}{k}\binom{n}{v+k} \leqslant \sum_{k=1}^{n-v}\frac{1}{k}\binom{n-v}{k}.$$

Using the inequality above, Lemma 1 (substituting $n - v$ for n), and Lemma 2, we obtain

$$S_{\text{even}} - S_{\text{odd}} = O\!\left(\frac{1}{n}\right),$$

when $1 \leqslant v < n$. We obtain the same result for the case $v = n$ by using Lemma 2 alone.

---

[1] R. Motwani and P. Raghavan, Randomized algorithms, ACM Comput. Surv. **28**, 33 (1996).

[2] R. Portugal, *Quantum Walks and Search Algorithms* (Springer, Cham, 2018).

[3] N. Konno, R. Portugal, I. Sato, and E. Segawa, Partition-based discrete-time quantum walks, Quantum Inf. Process. **17**, 100 (2018).

[4] W. Zhou, Review on quantum walk algorithm, J. Phys.: Conf. Ser. **1748**, 032022 (2021).

[5] M. Mohseni, P. Rebentrost, S. Lloyd, and A. Aspuru-Guzik, Environment-assisted quantum walks in photosynthetic energy transfer, J. Chem. Phys. **129**, 174106 (2008).

[6] S. E. Venegas-Andraca, Quantum walks: A comprehensive review, Quantum Inf. Process. **11**, 1015 (2012).

[7] P. Benioff, *Space Searches with a Quantum Robot*, AMS Contemporary Math Series Vol. 305 (AMS, Providence, RI, 2002).

[8] A. Ambainis, J. Kempe, and A. Rivosh, Coins make quantum walks faster, in *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms SODA* (SIAM, Philadelphia, PA, 2005), pp. 1099–1108.

[9] A. Tulsi, Faster quantum-walk algorithm for the two-dimensional spatial search, Phys. Rev. A **78**, 012310 (2008).

[10] A. Tulsi, General framework for quantum search algorithms, Phys. Rev. A **86**, 042331 (2012).

[11] A. Ambainis, Quantum walk algorithm for element distinctness, SIAM J. Comput. **37**, 210 (2007).

[12] M. Szegedy, Quantum speed-up of Markov chain based algorithms, in *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science, FOCS '04* (IEEE, Washington, DC, 2004), pp. 32–41.

[13] H. Krovi, F. Magniez, M. Ozols, and J. Roland, Quantum walks can find a marked element on any graph, Algorithmica **74**, 851 (2016).

[14] A. Ambainis, A. Gilyén, S. Jeffery, and M. Kokainis, Quadratic speedup for finding marked vertices by quantum walks, in *Proceedings of the Annual ACM Symposium on Theory of Computing* (ACM, New York, 2020), pp. 412–424.

[15] D. Aharonov, A. Ambainis, J. Kempe, and U. Vazirani, Quantum walks on graphs, in *Proceedings of the 33rd STOC* (ACM, New York, 2001), pp. 50–59.

[16] R. Portugal, Establishing the equivalence between Szegedy's and coined quantum walks using the staggered model, Quantum Inf. Process. **15**, 1387 (2016).

[17] F. Zhang, Y.-C. Chen, and Z. Chen, Clique-inserted-graphs and spectral dynamics of clique-inserting, J. Math. Anal. Appl. **349**, 211 (2009).

[18] T. G. Wong and R. A. M. Santos, Exceptional quantum walk search on the cycle, Quantum Inf. Process. **16**, 154 (2017).

[19] J. Abhijith and A. Patel, Spatial search using flip-flop quantum walk, Quantum Inf. Comput. **18**, 1295 (2018).

[20] M. Li and Y. Shang, Generalized exceptional quantum walk search, New J. Phys. **22**, 123030 (2020).

[21] A. Glos, N. Nahimovs, K. Balakirev, and K. Khadiev, Upperbounds on the probability of finding marked connected components using quantum walks, Quantum Inf. Process. **20**, 6 (2021).

[22] N. Shenvi, J. Kempe, and K. B. Whaley, A quantum random walk search algorithm, Phys. Rev. A **67**, 052307 (2003).

[23] L. K. Grover, Quantum Mechanics Helps in Searching for a Needle in a Haystack, Phys. Rev. Lett. **79**, 325 (1997).

[24] G. Brassard, P. Høyer, M. Mosca, and A. Tapp, Quantum amplitude amplification and estimation, in *Quantum Computation and Quantum Information Science*, AMS Contemporary Mathematics Series Vol. 305 (AMS, Providence, RI, 2002), pp. 53–74.

[25] F. L. Marquezino, R. Portugal, and G. Abal, Mixing times in quantum walks on two-dimensional grids, Phys. Rev. A **82**, 042341 (2010).

[26] C. Moore and A. Russell, Quantum walks on the hypercube, in *Proceedings of the 6th International Workshop on Randomization and Approximation Techniques RANDOM* (Springer, Berlin, 2002), pp. 164–178.

[27] R. Portugal, R. A. M. Santos, T. D. Fernandes, and D. N. Gonçalves, The staggered quantum walk model, Quantum Inf. Process. **15**, 85 (2016).

[28] http://www-elsa.physik.uni-bonn.de/~dieckman/InfProd/InfProd.html.

[29] https://oeis.org/A000670.