Continuous-time quantum search on balanced trees

Pascal Philipp* and Luís Tarrataca
National Laboratory for Scientific Computing, Petrópolis, CEP 25651-075, Rio de Janeiro, Brazil

Stefan Boettcher
Department of Physics, Emory University, Atlanta, Georgia 30322, USA
(Received 3 January 2016; published 4 March 2016)

We examine the effect of network heterogeneity on the performance of quantum search algorithms. To this end, we study quantum search on a tree for the oracle Hamiltonian formulation employed by continuous-time quantum walks. We use analytical and numerical arguments to show that the exponent of the asymptotic running time $\sim N^{\beta}$ changes uniformly from $\beta = 0.5$ to $\beta = 1$ as the searched-for site is moved from the root of the tree towards the leaves. These results imply that the time complexity of the quantum search algorithm on a balanced tree is closely correlated with certain path-based centrality measures of the searched-for site.

DOI: 10.1103/PhysRevA.93.032305

I. INTRODUCTION

In this paper we study continuous-time quantum search on balanced binary trees, on which all leaves have the same distance from the root and where no branches are missing. Our goal is to determine (i) how network heterogeneity influences the performance of the algorithm and (ii) whether there is speedup over the $O(N)$ time complexity of classical approaches, where N is the total number of sites. Suppose a quantum walker undertakes a blind search on such a tree structure that provides no global information and where edges leading to descendent sites cannot be distinguished from edges leading to parent sites. The walker is only given an oracle Hamiltonian that allows one to check whether the searched-for site (which we will also call the marked site) has been reached. Then, starting from a uniform initial state, how long does it take to find that site?

Grover's algorithm [1] provides a way to perform a discrete-time quantum search in an unstructured space with $O(\sqrt{N})$ oracle queries, which is optimal [2]. A continuous-time version with the same running time was presented in Ref. [3]. Reference [4] describes a discrete-time algorithm capable of searching a d-dimensional periodic lattice in $O(\sqrt{N})$ time for $d \geqslant 3$ and $O(\sqrt{N}\mathrm{poly}(\log N))$ for $d = 2$. In the continuous-time setting, the problem on the lattice is analyzed in Ref. [5]. The authors show that (i) we have quadratic speedup $O(\sqrt{N})$ for $d > 4$, (ii) $O(\sqrt{N}\mathrm{poly}(\log N))$ time is required for $d = 4$, and (iii) there is no significant speedup in lower dimensions. Recently, dimensionality reduction methods using symmetries were formalized [6] and a variety of new structures has been studied [6,7]. In Ref. [7] quadratic speedup was obtained only after modifying the weights of certain edges of a simplex of complete graphs.

The references mentioned so far examine homogeneous structures in which all sites are equivalent. The behavior changes significantly if one looks at graphs in which there are qualitatively different sites. The tree under consideration belongs to this category; for example, the degrees of leaf sites differ from those in the interior. Quantum search on structures that are less symmetric and more heterogeneous was explored in Refs. [8–10] and [11] in the discrete-time and continuous-time setting, respectively. References [8,10,12] investigate the correlation between the efficiency for the search of a certain site and its centrality or connectivity. More recently, quantum walks on Erdős-Rényi random graphs [13,14] and on scale-free graphs and hierarchical structures [14] have been studied.

It is thus natural to ask how location affects time complexity and if any variation in algorithmic behavior can be tied to site-specific properties, e.g., to its degree or centrality. Such questions regarding location and site-specific properties are particularly pertinent for quantum walks, as these do not converge in the sense of classical diffusion. Instead, they require a more finely tuned prescription on exactly when to measure the state of the system. In this paper we will see how these matters influence the time complexity for quantum search on a balanced tree.

Whether or not there is speedup on balanced trees is of interest, because there are conflicting intuitive arguments: Quantum walks tend to be more effective on high-dimensional structures and on structures that have a multitude of paths connecting any given pair of sites. Regarding trees, with exactly one path between any two sites, this suggests poor performance. On the other hand, it seems possible that a quantum algorithm can take advantage of the very small diameter of the tree. There are other properties of trees (such as the exponential spread of volume, the poor transport properties on trees [6], and the good transport properties across glued trees [15]) that may influence our expectation. What efficiency does the combination of all these factors lead to?

The main result of this work is that the time complexity of the quantum search algorithm on a balanced binary tree depends on the location of the searched-for site. The root can be found in $\Theta(\sqrt{N})$ time, while for finding a leaf there is no speedup and $O(N)$ time is needed. In between these two cases, the exponent of the time complexity $\sim N^{\beta}$ changes linearly from $\beta = 0.5$ to $\beta = 1$. In order to arrive at this conclusion, we reduce calculations on the balanced tree to a quasi-one-dimensional problem. We then solve analytically the case when the marked site is the root of the tree and the other cases we

*Corresponding author: pasc.philipp@gmail.com

treat numerically for systems large enough (up to size $N \approx 2^{64}$) to allow for the identification of the scaling exponent of the running time.

The above results describe the performance of a quantum algorithm that works with minimal knowledge of the structure of the graph and that searches in an unsystematic way starting from a uniform distribution. If more information is available (for example, the location of the root and whether an outgoing edge leads to a parent site or to a child) and if different operators can be used alternatingly, then the search can be accelerated. Using an effective technique [16] to determine whether a subgraph contains the marked site, this was recently done in Ref. [17] in the discrete-time setting, achieving a $O(\sqrt{N}\text{poly}(\log N))$ running time that covers all cases for the position of the searched-for site.

The paper is divided into two parts and structured as follows. In the first part, in Secs. II–VIII, we focus on the case when the marked site is the root in order to be able to carry out a symbolic analysis. The second part, the generic case of a marked site placed anywhere in the tree, consists of numerical investigations. We first introduce the setting for our quantum search problem on the tree (Sec. II). Next we discuss a technique for reducing the size of a system (Sec. III), which is then applied to the tree (Sec. IV). Working with the reduced system, we proceed to find the Laplace transform of the state at the searched-for site exactly (Sec. V). We then approximate this Laplace transform with expressions that have simple inverse transforms (Secs. VI and VII). These steps yield an explicit formula for the asymptotic running time in the root case. Following the result for the quantum algorithm, we briefly compare to a classical random walk (Sec. VIII). Afterward we start to consider the general situation by extending the reduction method (Sec. IX). We then continue with numerical experiments (Secs. X and XI), taking advantage of the fact that the small size of the reduced system allows for simulation of very large systems. As a last topic, we compare the time complexities we have found to different centrality measures (Sec. XII). Finally, we summarize our conclusions (Sec. XIII).

II. SETTING FOR THE QUANTUM SEARCH ALGORITHM

Consider a balanced binary tree of depth d, as is pictured in Fig. 1. The total number of sites is $N = 2^d - 1$. Let D be the degree matrix and A the adjacency matrix and define the graph Laplacian $L = D - A$. For example, for a tree with $d = 3$ levels we have

$$L_3 = \begin{bmatrix} 2 & -1 & -1 & & & & \\ -1 & 3 & & -1 & -1 & & \\ -1 & & 3 & & & -1 & -1 \\ & -1 & & 1 & & & \\ & -1 & & & 1 & & \\ & & -1 & & & 1 & \\ & & -1 & & & & 1 \end{bmatrix}.$$

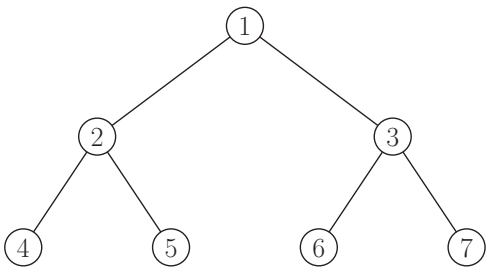

FIG. 1. Balanced binary tree of depth $d = 3$.

In the first part of this paper (up to Sec. VIII), we restrict our attention to the case where the marked site $|w\rangle$, the site that is being sought, is the root: $|w\rangle = |1\rangle$. We study the quantum search algorithm given by the Hamiltonian

$$H = \gamma L - |w\rangle\langle w|, \tag{1}$$

which was proposed in Ref. [5]. The initial state is the uniform distribution

$$|\psi(0)\rangle = |s\rangle = \frac{1}{\sqrt{N}} \sum_{k=1}^{N} |k\rangle.$$

Depending on the graph under consideration and on the location of the searched-for site, there may exist values of the search parameter γ for which evolution with respect to H is very effective in shifting statistical weight towards $|w\rangle$; for analyzing the algorithm it is crucial to find these optimal values γ_*. For example, Ref. [5] computes γ_* for search with (1) on a periodic lattice and shows $O(\sqrt{N})$ time complexity in sufficiently large spatial dimensions. For γ away from γ_*, however, the quantum algorithm fails to provide speedup over the $O(N)$ running time of classical search. Note that γ_* might not be constant as the system size increases.

III. REDUCTION IN GENERAL

We first review a technique for reducing the size of the system. A similar approach was recently presented in Ref. [6]. In the following, state vectors and operators in the reduced space will always be denoted by an overline.

Given H and an initial state $\psi(0)$, the solution to an evolution problem is

$$\psi(t) = f(t,H)\psi(0),$$

where, for instance, $f(t,z) = e^{-itz}$ for evolution according to the Schrödinger equation. Now suppose we have a linear reduction method V that transforms the system of size N to a system of size n. The initial state in the reduced system is $\overline{\psi}(0) = V\psi(0)$. For the evolution

$$\overline{\psi}(t) = f(t,\overline{H})\overline{\psi}(0)$$

in the reduced system to reflect the dynamics of the original system, it is necessary that

$$\overline{\psi}(t) = V\psi(t).$$

If f is analytic, we are led to the condition

$$\overline{H}^k V\psi(0) = VH^k\psi(0) \quad \text{for } k \in \mathbb{N}$$

and hence to

$$VHu = \overline{H}Vu \quad \text{for } u \in U = \text{span}_{k\geqslant 0}\{H^k\psi(0)\}. \tag{2}$$

Let V^+ be the pseudoinverse of the $n \times N$ matrix V. For example, if V transforms a graph of size 3 to a graph of size 2 by simply adding up the values of two of the sites, then V and V^+ are

$$V = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}, \quad V^+ = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{bmatrix}.$$

The matrix V has full rank n and therefore we have $VV^+ = I_n$. Now suppose that the set on which V^+V acts as the identity matrix coincides with the subspace U in (2). Then we have

$$\overline{H}Vu = VHu = VHV^+Vu$$

and we obtain the operator

$$\overline{H} = VHV^+ \tag{3}$$

on the reduced space. The operator $\overline{H}$ reproduces evolution starting from the initial state $\psi(0)$ without any loss of information [provided the two conditions above are satisfied; for the spectra of the two Hamiltonians we have $\sigma(\overline{H}) \subseteq \sigma(H)$ and the eigenvalues of H that are not in $\sigma(\overline{H})$ play no part in the dynamics since their eigenvectors do not overlap with $\psi(0)$].

IV. REDUCTION FOR THE TREE

We will reduce the tree by combining all sites with the same distance to $|w\rangle = |1\rangle$. Hence the size of the reduced system is $n = d$ and we will use matrices of the form

$$V_3 = \begin{bmatrix} 1 & & & & & & \\ & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & & & & \\ & & & \frac{1}{\sqrt{4}} & \frac{1}{\sqrt{4}} & \frac{1}{\sqrt{4}} & \frac{1}{\sqrt{4}} \end{bmatrix}$$

(for $d = 4$ append one row and eight columns and set the eight entries at the bottom right equal to $1/\sqrt{8}$, etc.). For this reduction method we have $V^+ = V^\top$.

Define the states

$$|u_j\rangle = \frac{1}{\sqrt{2^{j-1}}} \sum_{k=2^{j-1}}^{2^j-1} |k\rangle$$

and note that

$$\mathrm{span}_{1 \leqslant j \leqslant n}\{|u_j\rangle\} = \mathrm{span}_{k \geqslant 0}\{H^k|\psi(0)\rangle\} = U.$$

Moreover, one can check that $V^+V|u\rangle = |u\rangle$ for $|u\rangle \in U$. Therefore, by the theory from the previous section, the reduction V is suitable for the quantum search problem under consideration.

Letting the basis of the reduced space be

$$\overline{|j\rangle} = V|u_j\rangle,$$

we obtain Hamiltonians of the form

$$\overline{H}_3 = \gamma \begin{bmatrix} 2 - \frac{1}{\gamma} & -\sqrt{2} & 0 \\ -\sqrt{2} & 3 & -\sqrt{2} \\ 0 & -\sqrt{2} & 1 \end{bmatrix}$$

[the matrix is tridiagonal for all n, the diagonal entries are $\gamma(2-\frac{1}{\gamma},3,3,\ldots,3,1)$, and all off-diagonal entries are $-\sqrt{2}\gamma$].

The marked state of the reduced system is $\overline{|w\rangle} = \overline{|1\rangle}$ and the initial state is

$$\overline{|\psi(0)\rangle} = \overline{|s\rangle} = V|s\rangle = \frac{1}{\sqrt{N}} \sum_{j=1}^{n} \sqrt{2^{j-1}}\,\overline{|j\rangle}.$$

V. LAPLACE TRANSFORM OF THE STATE AT THE MARKED SITE

We are interested in the amplitude $\langle\psi|w\rangle = \langle\psi|1\rangle = \psi_1$ of the wave vector at the searched-for site. As a first step, we now compute its Laplace transform exactly. Note that $\overline{\psi}_1 = \psi_1$.

Taking the Laplace transform of the evolution equation $i\partial_t\overline{\psi}(t) = \overline{H}\,\overline{\psi}(t)$ gives

$$i\alpha s\,\widetilde{\overline{\psi}}(s) - i\alpha\overline{\psi}(0) = \alpha\overline{H}\,\widetilde{\overline{\psi}}(s),$$

where $\alpha = \gamma^{-1}$. Writing out that system of equations, multiplying the kth equation by x^{k-1}, and adding up all of them, we find

$$G(s;x) = \left\{\left[1 - \frac{1+\alpha}{\sqrt{2}}x\right]\widetilde{\overline{\psi}}_1 + [x^{n+1} - \sqrt{2}x^n]\widetilde{\overline{\psi}}_n + \frac{i\alpha}{\sqrt{2N}}\sum_{k=1}^{n}\sqrt{2^{k-1}}x^k\right\} \Big/ \left(x^2 - \frac{3 - i\alpha s}{\sqrt{2}}x + 1\right) \tag{4}$$

for $G(s;x) = \sum_{k=1}^{n}\widetilde{\overline{\psi}}_k(s)x^{k-1}$. If we denote the zeros of the denominator in (4) by x_0 and x_1, we have $x_0x_1 = 1$ and we let x_0 be the zero that lies in the unit disk. Next we divide (4) by x and integrate with respect to x over the unit circle. This leads to

$$\left[x_1 - \frac{1+\alpha}{\sqrt{2}}\right]\widetilde{\overline{\psi}}_1 + \left[x_0^{n-1}(x_0 - \sqrt{2})\right]\widetilde{\overline{\psi}}_n + \frac{i\alpha}{\sqrt{2N}}\sum_{k=1}^{n}(\sqrt{2}x_0)^{k-1} = 0. \tag{5}$$

We derive a second equation for $\widetilde{\overline{\psi}}_1$ and $\widetilde{\overline{\psi}}_n$ by multiplying (4) by x^{-n} and again integrating over the unit circle:

$$\left[x_0^{n-1}\left(x_0 - \frac{1+\alpha}{\sqrt{2}}\right)\right]\widetilde{\overline{\psi}}_1 + [x_1 - \sqrt{2}]\widetilde{\overline{\psi}}_n + \frac{i\alpha}{\sqrt{2N}}\sum_{k=1}^{n}\sqrt{2^{k-1}}x_0^{n-k} = 0. \tag{6}$$

Combining (5) and (6), we obtain an explicit formula for the Laplace transform of ψ_1:

$$\widetilde{\psi}_1 = \frac{i}{\sqrt{N}}\frac{x_1^n - x_0^n}{x_1^{n-1}[(is+1)x_1 - \sqrt{2}] - x_0^{n-1}[(is+1)x_0 - \sqrt{2}]}. \tag{7}$$

VI. APPROXIMATION FOR SMALL VALUES OF THE SEARCH PARAMETER

We will now find an approximation of ψ_1 in the range $\gamma \in [0, 1-\varepsilon]$ and we will see that the algorithm fails for those values of the search parameter. Hence the optimal value γ_* must be 1 or larger. That is noteworthy, since in most known

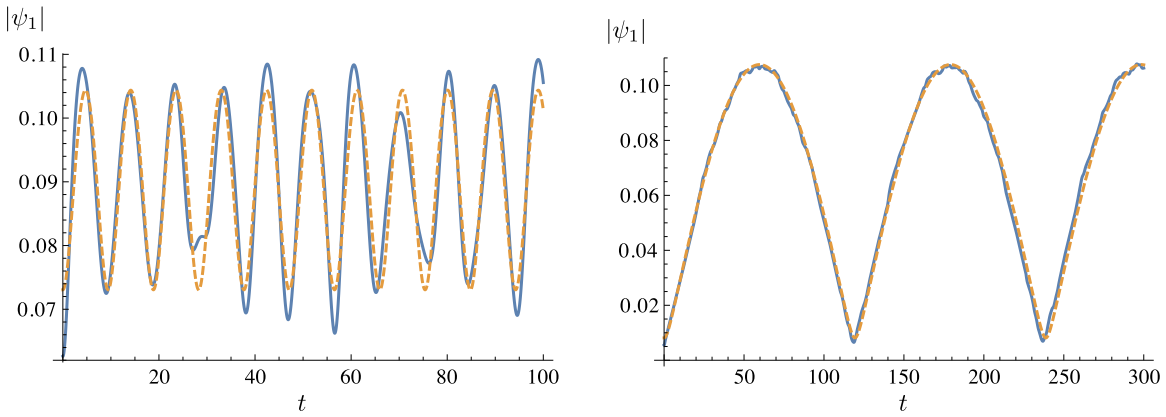

FIG. 2. Absolute value of ψ_1 (blue solid line) and of the approximation (9) (orange dashed line) as a function of time for $\gamma = 0.2$ and $n = 8$ (left) and for $\gamma = 0.9$ and $n = 15$ (right).

examples γ_* tends to small positive constants or to zero as $N \to \infty$; the general trend is $\gamma_* \sim \frac{1}{\delta}$, where δ is the degree of the searched-for site [5–7,11–13,18].

Recall the definition of x_0 and x_1 after (4). Since $|x_0| < 1$, we have $x_0^n \to 0$ and (7) yields

$$\widetilde{\psi}_1 \approx \frac{i}{\sqrt{N}} \frac{x_1}{(is+1)x_1 - \sqrt{2}} \tag{8}$$

for large n. This expression has poles $s = 0$ and $s = i\frac{\alpha-1}{\alpha+1}$ and computing the corresponding residues gives

$$\psi_1 \approx \frac{1}{\sqrt{N}}\left[\frac{1}{1-\alpha}1 + \frac{\alpha^2+2\alpha-1}{\alpha^2-1}\exp\left(i\frac{\alpha-1}{\alpha+1}t\right)\right], \tag{9}$$

where $\alpha = \gamma^{-1}$.

Numerical experiments show that (9) is adequate for $\gamma \in [0, 1-\varepsilon]$ (cf. Fig. 2). Note that (9) gives the exact solution for $\gamma = 0$. However, the low success probabilities in Fig. 2 (as well as the fact that the frequency does not explicitly depend on n) suggest that the oscillations in this range of γ are trivial and not useful for finding the marked site.

The time complexity of a quantum search algorithm is quantified by means of the expression

$$\frac{t_0}{p_0} = \frac{t_0}{|\langle w|e^{-it_0H}|s\rangle|^2}, \tag{10}$$

the time of the measurement t_0 divided by the success probability $p_0 = p(t_0)$. The division by p_0 takes into account that the search might have to be repeated if $p_0 < 1$. In cases where the success probability is not bounded from below by a constant as $N \to \infty$, the search has to be repeated increasingly often (we will see such behavior in Sec. XI; cf. the $d = 4$ case in Ref. [5] for another example). When (9) is valid, we can compute p_0 and with it the time complexity directly by evaluating $|\psi_1|$ at its first maximum $t_0 = \frac{\alpha+1}{\alpha-1}\pi$:

$$\frac{t_0}{p_0} \approx \pi\frac{(\alpha+1)^3(\alpha-1)}{\alpha^2(\alpha+3)^2}N = O(N).$$

VII. SEARCH WITH THE CORRECT PARAMETER

Let us now set $\gamma = 1$. In this case the asymptotic approximation (8) has a double pole at $s = 0$ and we need to find the sequence of poles that gives rise to it. To this end, we first write (7) as

$$\widetilde{\psi}_1 = \frac{f(s)}{g(s)},$$

where the prefactor $iN^{-1/2}$ is contained in the numerator $f(s)$. The Laplace transform $\widetilde{\psi}_1$ has a number of poles, but the method from the previous section suggests that the ones close to $s = 0$ are the relevant ones and we use a second-order approximation of $g(s)$ to find them. This gives, in the limit $n \to \infty$,

$$p_\pm \approx \frac{\pm i}{\sqrt{2^{n+1}}} \tag{11}$$

and we obtain

$$\psi_1 \approx r_+ e^{p_+ t} + r_- e^{p_- t}, \tag{12}$$

where the coefficients $r_\pm$ are the evaluations of the residue function $\frac{f(s)}{g'(s)}$ at $p_\pm$. We further simplify by evaluating $\frac{i}{\widetilde{g}(s)}$ [we have $f(0) \approx i$], where $\widetilde{g}$ is the first-order approximation of g' at 0, instead:

$$r_\pm \approx \pm\frac{1}{2\sqrt{2}}.$$

These computations lead to

$$\psi_1 \approx \frac{1}{2\sqrt{2}}[e^{p_+ t} - e^{p_- t}] = \frac{i}{\sqrt{2}}\sin\left(\frac{t}{\sqrt{2^{n+1}}}\right). \tag{13}$$

Figure 3 shows that the approximations are accurate for large system sizes. The maximum amplitude at $|w\rangle$ is $2^{-1/2}$ and the wavelength of the oscillation is $\sqrt{2^{n+3}}\,\pi$. We can now determine the time complexity for the search algorithm; it is

$$\frac{t_0}{p_0} \approx \pi\sqrt{2^{n+1}} \approx \sqrt{2}\pi\sqrt{N} = \Theta(\sqrt{N}). \tag{14}$$

As concluding remarks for this section, we first remind the reader that the asymptotic formula (14) is not general; it covers only the case when the marked site is the root of the tree. Second, we point out that the quantities $\pm 2^{-(n+1)/2}$ in (11) are, in the limit $n \to \infty$, the two smallest eigenvalues of $\overline{H}$; this observation relates our calculations to the more immediate method of predicting measurement times via energy gaps

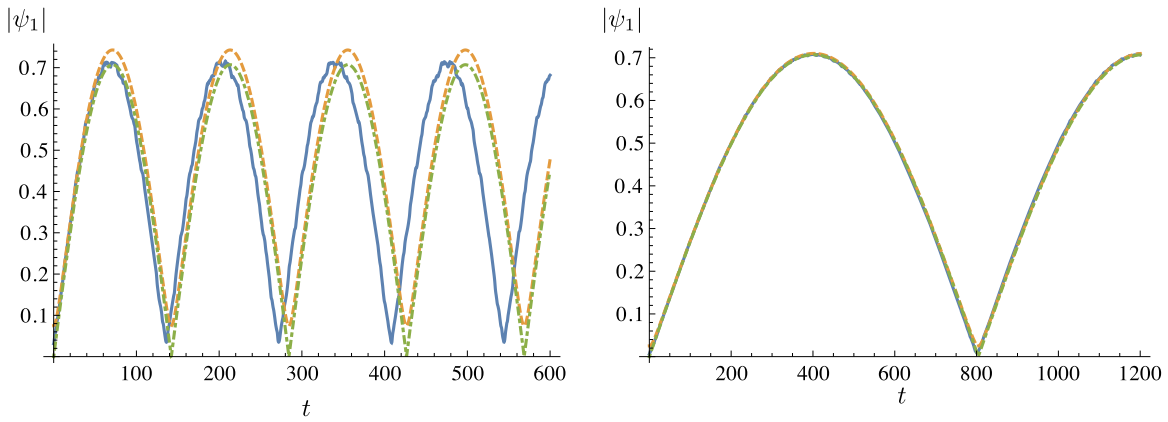

FIG. 3. Absolute value of ψ_1 (blue solid line) and of the approximations (12) (orange dashed line) and (13) (green dot-dashed line) as a function of time for $n = 10$ (left) and $n = 15$ (right), with $\gamma = 1$.

(cf., for example, Ref. [5]; note that the second smallest eigenvalue of H is not always the same as the second smallest eigenvalue of $\overline{H}$). See Fig. 5 for a comparison of (14) to other approaches for obtaining estimates of the time complexity.

VIII. SEARCH WITH A CLASSICAL RANDOM WALK

We briefly compare to the performance of a classical random walker. Suppose the walker is randomly placed in the tree: How long does it take to locate the root (i.e., the one site that has degree 2)?

Let t_k be the average time a random walker that starts from a site on level k needs to find the root. Then we have

$$\begin{aligned} t_1 &= 0, \\ t_k &= \tfrac{1}{3}t_{k-1} + \tfrac{2}{3}t_{k+1} + 1, \\ t_n &= t_{n-1} + 1. \end{aligned} \tag{15}$$

Define the weighted times $\widetilde{t}_k$ and the average search time T:

$$\widetilde{t}_k = \frac{2^{k-1}}{N} t_k, \quad T = \sum_{k=1}^{n} \widetilde{t}_k.$$

Then (15) leads to

$$N \begin{bmatrix} 1\times 2^{-1} & -\frac{2}{3}\times 2^{-2} & & & \\ -\frac{1}{3}\times 2^{-1} & 1\times 2^{-2} & -\frac{2}{3}\times 2^{-3} & & \\ & -\frac{1}{3}\times 2^{-2} & 1\times 2^{-3} & & \\ & & & \ddots & \\ & & & 1\times 2^{-(n-2)} & -\frac{2}{3}\times 2^{-(n-1)} \\ & & & -1\times 2^{-(n-2)} & 1\times 2^{-(n-1)} \end{bmatrix} \begin{bmatrix} \widetilde{t}_2 \\ \widetilde{t}_3 \\ \widetilde{t}_4 \\ \vdots \\ \widetilde{t}_{n-1} \\ \widetilde{t}_n \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \\ 1 \end{bmatrix}.$$

We next renormalize the diagonal by multiplying all equations by appropriate powers of 2 and then we add all of them. This gives

$$\frac{N-1}{N} = \left[1 - \tfrac{1}{3} - \tfrac{2}{3}\right]T + \tfrac{1}{3}\widetilde{t}_2 - \tfrac{4}{3}\widetilde{t}_{n-1} + \tfrac{2}{3}\widetilde{t}_n$$

and consequently

$$t_2 = N - 2.$$

Hence, since the walker needs linear time even when starting from a neighbor of the searched-for site, we have $\Omega(N)$ (i.e., at least $\sim N$) average time complexity for this classical approach and we see that the quantum algorithm provides a significant speedup. As a side note we point out that in Ref. [19] it was shown that the average hitting times for random walks on arbitrary trees are always integers.

IX. REDUCTION IN THE GENERAL CASE

We now generalize our reduction method and allow $|w\rangle$ to be anywhere in the tree. We denote the level on which the marked site is by l (and n is the total number of levels, as before). We will see below that we can reduce the full system to one of size less than $n^2 \approx (\log_2 N)^2$ in a way similar to what has been done in Sec. IV. Equation (3) for the Hamiltonian in the reduced system $\overline{H} = VHV^+$ is still valid and useful, but the necessary conditions are not constructive; they do not show how to find the matrix V. We use the following intuitive argument: We can group together sites that are indistinguishable in the sense that (a) their positions in the graph are qualitatively the same and (b) their positions relative to $|w\rangle$ are the same. For example, for grouping sites together it is certainly necessary that they (a) have the same degree and (b) have the same distance from $|w\rangle$.

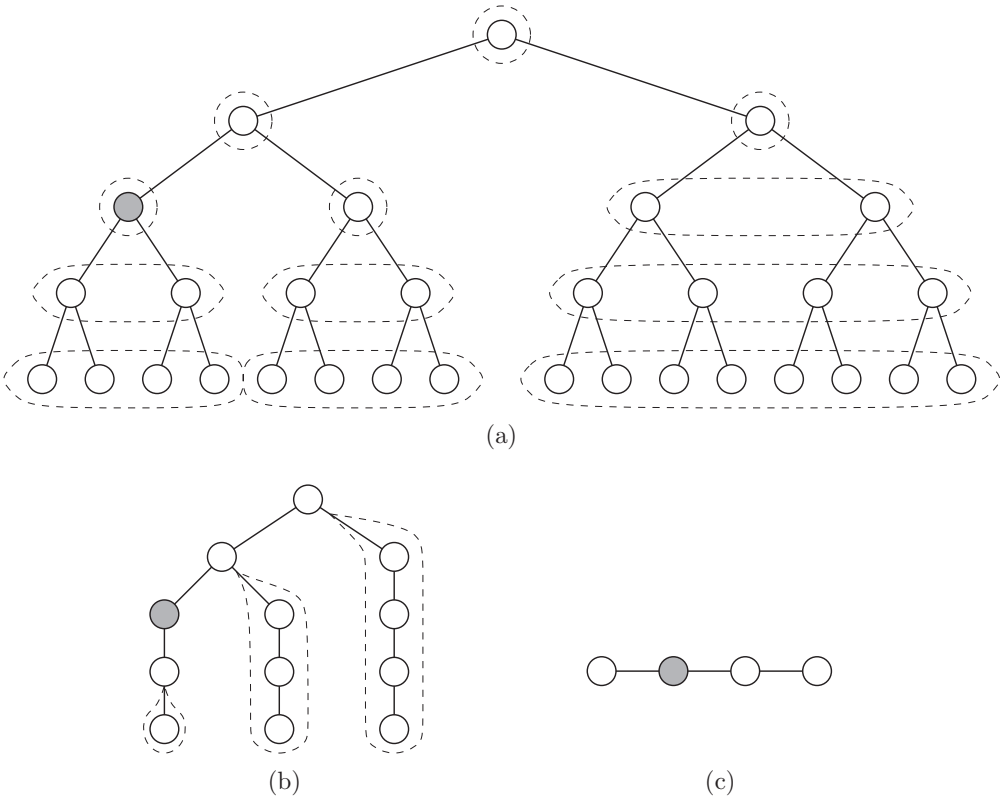

FIG. 4. Reduction and absorption for a tree of depth $n = 5$ with $|w\rangle$ on level $l = 3$: The (a) original tree is transformed first into (b) the comb structure and then into (c) the line graph.

An illustration of this reduction is given by Fig. 4. We can see how the full tree is transformed into a comb structure. We consider $|w\rangle$, all its ancestors, and the combination of its two children the backbone of that comb. Then we can transform the system to a line graph by absorbing the side chains into the backbone. However, this leads to an inhomogeneous Hamiltonian with nonconstant coefficients. The formula for absorption into ancestors of the searched-for site is

$$\begin{aligned}
&\left[1 + x_0^m(x_0 - \sqrt{2})\frac{x_0^m - x_1^m}{x_0 - x_1}\right]\widetilde{\phi}_1 \\
&= \frac{1}{\sqrt{2}}\left[x_0 + x_0^m(x_0 - \sqrt{2})\frac{x_0^{m-1} - x_1^{m-1}}{x_0 - x_1}\right]\widetilde{\phi}_0 \\
&\quad - \frac{1}{\sqrt{N}}\left[\frac{x_0^m(x_0 - \sqrt{2})}{s}\left(\sqrt{2}^{m-1} - \frac{x_0^m - x_1^m}{x_0 - x_1}\right.\right. \\
&\quad \left.\left. + \frac{1}{\sqrt{2}}\frac{x_0^{m-1} - x_1^{m-1}}{x_0 - x_1}\right) + \frac{i\alpha x_0}{\sqrt{2}}\frac{1 - (\sqrt{2}x_0)^m}{1 - \sqrt{2}x_0}\right]
\end{aligned}$$

in Laplace space, where ϕ_0 is the backbone site into which we want to absorb, ϕ_1 is the first element of the side chain we want to eliminate, and m is its size. For the absorption below $|w\rangle$, we have the same expression with the $\widetilde{\phi}_0$ term and the constant term changing by factors of $\sqrt{2}$ and 2, respectively.

The line graph in Fig. 4(c) might be useful for analytical considerations, but we will not work with it in the remainder of our analysis. Instead we focus on numerical simulations of the comb structure in Fig. 4(b), which is of size at most $\approx \frac{n^2}{2}$. For $\overline{H}$ we could generate the $N \times N$ matrix H and the reduction matrix V and then apply (3). After that we can use the much smaller matrix $\overline{H}$ for operations such as finding eigenvalues and for repeated evaluations of the propagator. This approach is systematic, but ineffective in terms of both memory and computing time. It is better to generate $\overline{H}$ directly, which is not difficult since the form of $\overline{L}$ is rather simple: The self-terms are always the degree of the respective site (or rather of one of its representatives) in the original tree and the entries coming from the adjacency matrix of the comb are $-\sqrt{2}$ if the corresponding edge actually is a combination of edges and -1 otherwise. (Recall that we renormalize when grouping sites of the tree into sites of the comb.)

It seems likely that the general case $l \neq 1$ can be solved analytically as well, either via the line graph in Fig. 4(c) or using a recursive method, but since our numerical method is very effective, doing that is not necessary for the purpose of this paper.

X. NUMERICAL EXPERIMENTS IN THE ROOT CASE

For numerical experiments we first revisit the case when the searched-for site is the root. Then we know from Sec. VII that $\gamma_* \to 1$. Figure 5 shows the time complexity in that situation for system sizes up to $n = 15$. It can be found in three ways: (a) via the asymptotics (14), (b) by using the gap between the two smallest eigenvalues of $\overline{H}$ to predict the measurement time and then evaluating the propagator once to determine the corresponding success probability, and (c) by extracting the

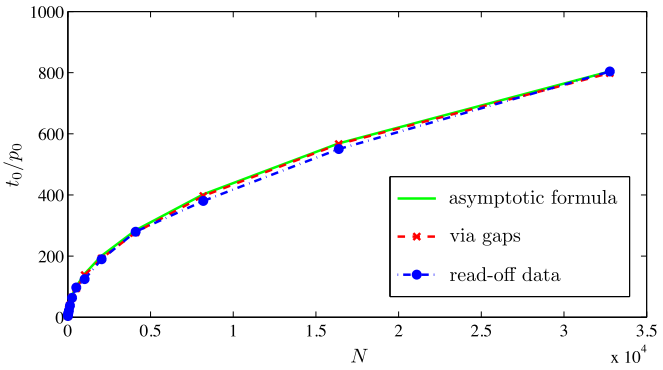

FIG. 5. Time complexity when $|w\rangle = |1\rangle$, with $\gamma = 1$.

necessary information from simulations of the evolution at the marked site.

In the next section we will study situations other than $l = 1$. Then we do not have a formula like (a) to compare to. Method (b) will not be available either, since the most significant oscillation at $|w\rangle$ does not always come from the two smallest energy levels. Therefore, we will be working with (c) only. An additional difficulty is that we will need different values for γ. If we can establish a formula $\gamma_* = \gamma_*(n,l)$, then repeating the search algorithm accordingly would increase the running time by at most a factor of $\log_2 N$.

XI. NUMERICAL EXPERIMENTS IN THE GENERAL CASE

First we need to find the optimal values of the search parameter. Hence we start by generating plots that show the maximum achievable concentration of probability at the searched-for site as a function of γ. For instance, for $l = 1$ we would see sharp peaks with their maxima quickly converging to $(\gamma_*, p_{\max}) = (1.0, 0.5)$ as $n \to \infty$. Let us denote the γ coordinates of those maxima by γ'_*. Figure 6 allows us to read off γ'_* for various configurations (n,l). We see that, depending on the level l of $|w\rangle$, different values are needed to maximize the success probability. If $|w\rangle$ is located far down in the tree, the best concentrations at it are obtained for large γ.

The optimal value γ_* of the search parameter is the value that minimizes $\frac{t_0}{p(t_0)}$, where t_0 is the measurement time. Figure 7 shows that the two notions of optimality agree away from $\frac{l}{n} = 1$. As $|w\rangle$ is placed farther down towards the bottom of the tree, both γ_* and t_0 are becoming less well defined. For example, for $l = n$ the choice of γ hardly matters and we always have the same qualitative behavior with constant measurement times. This explains the discrepancy close to $\frac{l}{n} = 1$ in Fig. 7. The same plot suggests $\gamma_* \approx \frac{2}{\delta}$, where δ is the degree of the searched-for site and which applies in most cases [we have, however, $\gamma_*(l = 2) = 0.75$].

More problematic than the fact that γ_* varies is the decay of success probabilities. While they are constant for fixed l,

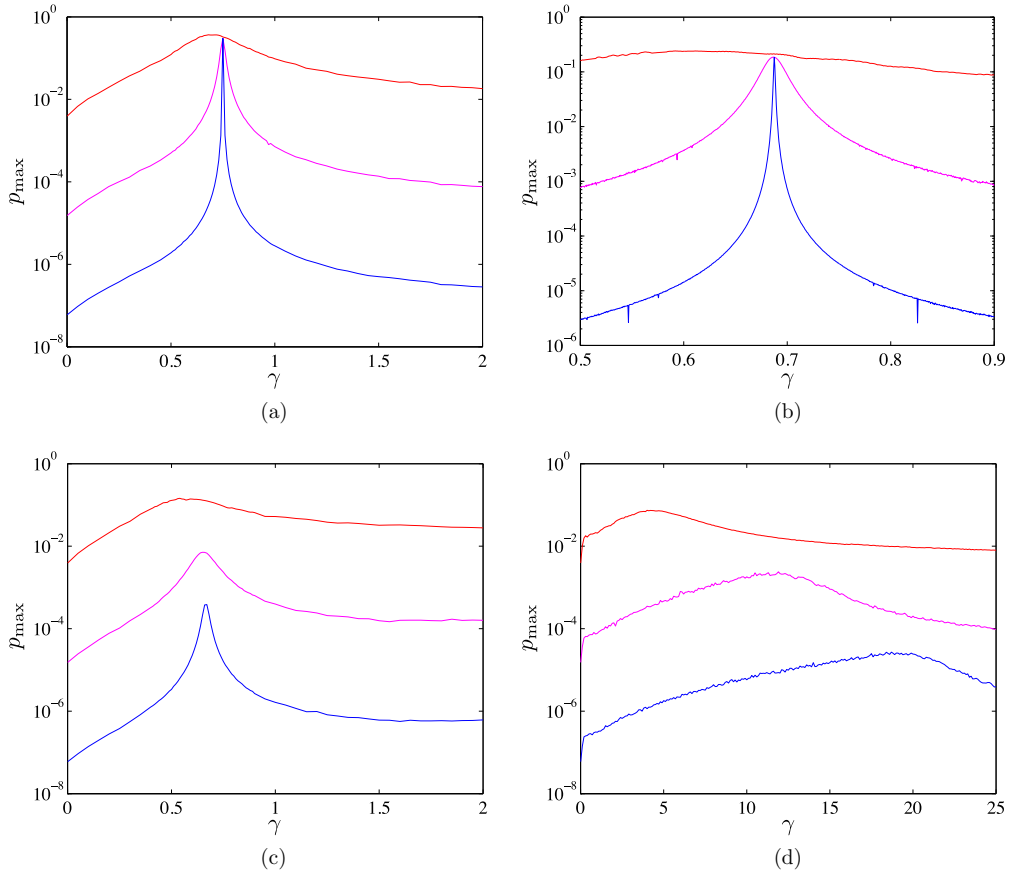

FIG. 6. Maximum success probability as a function of γ for $n = 8, 16, 24$ (red, magenta, blue, respectively, from top to bottom) for different positions l of the marked site: (a) $l = 2$, (b) $l = 3$, (c) $l = n/2$, and (d) $l = n - 1$.

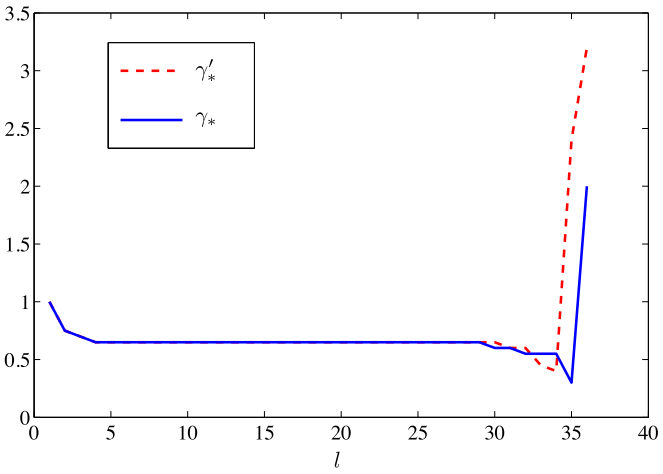

FIG. 7. Comparison of γ'_* and γ_* for $n = 36$.

we have a substantial decrease in the more significant case $l \sim n$. This behavior has been pointed out in Refs. [8–11] for different examples of nonhomogeneous graphs. For analyzing the time complexity, it remains to include the measurement times in our considerations.

For $l = \frac{n}{2}$, γ_* converges to $\frac{2}{3}$ [cf. Figs. 6(c) and 7]. Therefore, we simulate the evolution of the system for increasing sizes with $\gamma = \frac{2}{3}$. Collecting the data $\frac{t_0}{p(t_0)}$ for $n = 8,12,16,\ldots,64$ yields Fig. 8, which shows that the search is better than linear in time. By means of an extrapolation plot, we can determine the exponent of the asymptotic behavior $\sim N^\beta$ with high accuracy; we have $\beta = 0.7500 = \frac{3}{4}$.

Simulations of the time evolution at $|w\rangle$ for $l = n$ show the kind of trivial oscillations that we have seen in Sec. VI. These oscillations do not facilitate a search; they have constant wavelength and maxima (which correspond to success

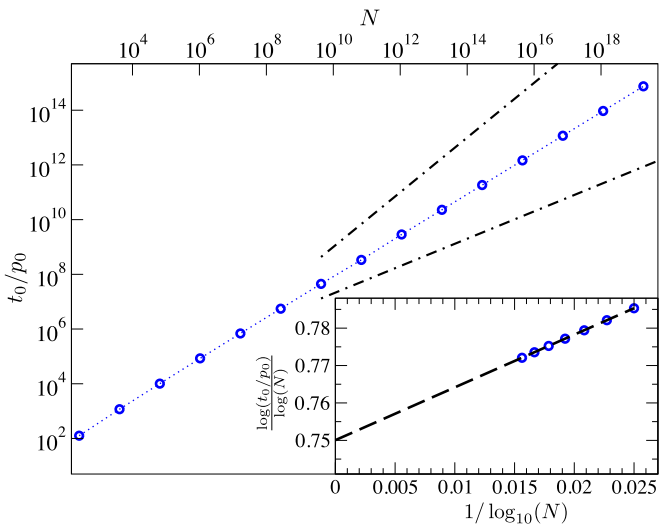

FIG. 8. Running time when $l = \frac{n}{2}$ for systems up to size $N = 2^{64} - 1 \approx 10^{19}$ ($\gamma = \frac{2}{3}$). The extrapolation plot in the bottom right corner shows that the exponent of the time complexity $\sim N^\beta$ is $\beta = \frac{3}{4}$. The dash-dotted lines represent scaling of order N (top) and $\sqrt{N}$ (bottom), for comparison.

TABLE I. Time complexities $\sim N^\beta$ for sites on various levels $l \sim n$ (obtained numerically by simulating systems of sizes $n = 4,8,\ldots,52$).

l	γ_*	β
1	1	0.500
$\frac{n}{4}$	$\frac{2}{3}$	0.625
$\frac{n}{2}$	$\frac{2}{3}$	0.750
$\frac{3n}{4}$	$\frac{2}{3}$	0.878
n	2	1.000

probabilities) decreasing of order N^{-1} in magnitude. Hence the quantum algorithm runs in $O(N)$ time and fails to provide speedup when the marked site is a leaf. We credit this to the less-than-good transport properties of trees. Note that more than half of the sites are leaves.

Table I provides an overview over the time complexities in several situations $l \sim n$; it shows that the exponent of the running time $\sim N^\beta$ changes uniformly from $\beta = 0.5$ to $\beta = 1$ as $|w\rangle$ is moved down the tree from the root to one of the leaves. To be more precise, we are led to the formula

$$\beta(l) = \frac{1}{2} + \frac{l}{2n}. \tag{16}$$

For l between $\frac{n}{4}$ and $\frac{3n}{4}$, one can confirm (16) with the same numerical experiments. Close to $\frac{l}{n} = 0$ and $\frac{l}{n} = 1$, simulations of even larger systems would be needed. In these cases, the data point towards the predictions given by (16), but they are not quite good enough to allow for clean extrapolations (for small $\frac{l}{n}$, the γ_* converge only slowly to $\frac{2}{3}$). However, since the trend is correct and since (16) also holds true at the boundary of $\frac{l}{n} \in [0,1]$, there is no reason to doubt that they do apply in the full range. When $l = \frac{3n}{4}$, there already is very little change in measurement times and the average complexity $\sim N^{0.88}$ comes mostly from the decrease in success probabilities. For $l = \text{const}$ the algorithm runs in optimal time $\Theta(\sqrt{N})$.

XII. COMPARISON TO CENTRALITY

Now that we have a very clear notion of how the time complexity for quantum search on a balanced tree varies depending on the location of the searched-for site, we would like to correlate our findings to some centrality measure. A multitude of different measures has been suggested (see Ref. [20] for an overview) and defining new ones that best answer to different purposes on different structures is an active field of study. In this section we will examine several common centrality measures, our aim being to find one that can predict how searchable a given site is; we would like to identify a centrality measure C that relates either to the time complexity $\sim N^\beta$ or to the scaling exponent β itself.

The most basic centrality measure is degree centrality $C_D(v) = \mathcal{D}(v)$, where $\mathcal{D}$ denotes degree. We see immediately that it does not serve our purpose, since the root is not ranked the highest. The same is true for the following walk-based

TABLE II. Exponent of the running time β and asymptotics of the closeness centrality C_C and the betweenness centrality C_B for sites on various levels $l \sim n$.

l	β	C_C	C_B
1	0.500	$(1.00n)^{-1}$	$\frac{1}{2}$
$\frac{n}{4}$	0.625	$(1.25n)^{-1}$	$4N^{-1/4}$
$\frac{n}{2}$	0.750	$(1.50n)^{-1}$	$4N^{-1/2}$
$\frac{3n}{4}$	0.878	$(1.75n)^{-1}$	$4N^{-3/4}$
n	1.000	$(2.00n)^{-1}$	0

measures, which all give preference to sites in the midrange or upper midrange of the tree: communicability [21], eigenvector centrality, subgraph centrality [22], and Katz centrality [23] (for connections between the latter three and degree centrality, see Ref. [24]).

We next test two path-based measures of centrality, which take into account only the shortest paths between sites rather than all possible walks. Betweenness centrality was introduced in Ref. [25] and is defined as follows:

$$\widetilde{C}_B(v) = \sum_{i\neq v}\sum_{j\neq v}\delta_{ij}(v),$$

where $\delta_{ij}(v)$ is the number of shortest paths from i to j that pass through v divided by the total number of shortest paths from i to j. The formula for closeness centrality [26] is

$$\widetilde{C}_C(v) = \left(\sum_j d(j,v)\right)^{-1},$$

where $d(j,v)$ is the distance between j and v. We produce normalized centralities C_B and C_C by dividing by the largest values of $\widetilde{C}_B$ and $\widetilde{C}_C$ a site in a graph of size N can possibly have. These factors come, in both cases, from the center site of the star graph.

Table II shows that both C_B and C_C rank sites on different levels of the tree in the correct order. Betweenness centrality can be linked to the time complexity $\sim N^\beta$, but the identity $C_B \sim \frac{N}{(N^\beta)^2}$ does not hold in the leaf case. For closeness centrality we have a perfect correlation between the scaling exponent β and the constant κ in the asymptotic closeness $(\kappa n)^{-1}$. Note that κn is, in the limit $n \to \infty$, the distance from the marked site to the leaves on the opposite side of the tree.

The strong correlation between the scaling exponent and closeness centrality raises the question of whether we have similar trends for other nonhomogeneous graphs or even for complex networks. As far as the authors are aware, there are only two studies of the relation between quantum searchability and centrality: In Ref. [8], it is also a closeness-type centrality measure that is being used. In Ref. [10], the success probability that can be obtained at the searched-for site is compared to eccentricity, which is defined as the maximum distance from the marked site to other sites of the graph and which for the tree under consideration is asymptotically equal to C_C^{-1} (cf. remark at the end of the previous paragraph).

XIII. CONCLUSION

In this paper we presented an analysis of a continuous-time quantum search algorithm on balanced binary trees. We saw that the running time depends on the location of the marked site. If it is a leaf that is being sought, there is no improvement of the linear-in-size running time of classical algorithms. However, the root can be found with quadratic speedup, in $\Theta(\sqrt{N})$ time. In between these two cases, the exponent of the time complexity $\sim N^\beta$ changes linearly from $\beta = 0.5$ to $\beta = 1$. Our work relied heavily on a dimensionality reduction method, which, besides allowing us to perform numerical experiments with very large systems, was also crucial for symbolic computations in a special case.

We would like to point out that our results do not imply that there is no effective continuous-time quantum algorithm for search on a balanced tree; with (1) we have only studied the most commonly used Hamiltonian for that purpose. For example, by changing the weights of some of the edges of a certain graph, Ref. [7] improved the running time from $\Theta(N^{3/4})$ to nearly $\Theta(N^{1/2})$.

We have also examined the relation of how effectively a site can be searched with its centrality. For the graphs under consideration, we have found a strong correlation between the scaling exponent β of the time complexity and closeness centrality. However, since balanced binary trees are highly nongeneric structures, more evidence is needed before one can formulate a general hypothesis. Given the apparent limited amount of studies on this topic, i.e., Refs. [8,10], it would be interesting to see more examples of quantum walks on nonhomogeneous graphs and to compare the differences in running times to closeness or other centrality measures.

As a concluding remark we would like to compare our work to Ref. [15]. In that paper the authors study a pair of balanced trees that are glued together along the leaves and they show propagation from one root to the other in $O(\log(N))$ time. In our framework, this corresponds to extending the reduced matrix in Sec. IV by an identical flipped copy, so that the quantum walk on a large structure is instead performed on a line of length $\sim\log(N)$. Reference [15] does not contradict the poor efficiency we have found in the case when the searched-for site is a leaf; instead it shows that there is a significant difference between transporting to one particular leaf and transporting to the collection of all leaves.

ACKNOWLEDGMENTS

P.P. and L.T. were supported by CNPq CSF and BJT Grants No. 400216/2014-0 and No. 301181/2014-4. S.B. acknowledges financial support from the U.S. National Science Foundation through Grant No. DMR-1207431 and from CNPq through the "Ciência sem Fronteiras" program and thanks LNCC for its hospitality. P.P. and L.T. would also like to thank Renato Portugal and Isabel Chen-Philipp for useful discussions.

[1] L. K. Grover, in *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing* (ACM, New York, 1996), pp. 212–219.
[2] C. H. Bennett, E. Bernstein, G. Brassard, and U. Vazirani, Strengths and weaknesses of quantum computing, SIAM J. Comput. **26**, 1510 (1997).
[3] E. Farhi and S. Gutmann, Analog analog of a digital quantum computation, Phys. Rev. A **57**, 2403 (1998).
[4] S. Aaronson and A. Ambainis, in *Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science* (IEEE, Piscataway, 2003), pp. 200–209.
[5] A. M. Childs and J. Goldstone, Spatial search by quantum walk, Phys. Rev. A **70**, 022314 (2004).
[6] L. Novo, S. Chakraborty, M. Mohseni, H. Neven, and Y. Omar, Systematic dimensionality reduction for quantum walks: Optimal spatial search and transport on non-regular graphs, Sci. Rep. **5**, 13304 (2015).
[7] T. G. Wong, Faster quantum walk search on a weighted graph, Phys. Rev. A **92**, 032320 (2015).
[8] S. Berry and J. Wang, Quantum-walk-based search and centrality, Phys. Rev. A **82**, 042333 (2010).
[9] N. Lovett, M. Everitt, M. Trevers, D. Mosby, D. Stockton, and V. Kendon, Spatial search using the discrete time quantum walk, Nat. Comput. **11**, 23 (2012).
[10] A. Mahasinghe, J. B. Wang, and J. K. Wijerathna, Quantum walk-based search and symmetries in graphs, J. Phys. A: Math. Theor. **47**, 505301 (2014).
[11] E. Agliari, A. Blumen, and O. Mülken, Quantum-walk approach to searching on fractal structures, Phys. Rev. A **82**, 012305 (2010).
[12] D. A. Meyer and T. G. Wong, Connectivity is a Poor Indicator of Fast Quantum Search, Phys. Rev. Lett. **114**, 110503 (2015).
[13] S. Chakraborty, L. Novo, A. Ambainis, and Y. Omar, Phys. Rev. Lett. (2016) (to be published).
[14] G. D. Paparo, M. Müller, F. Comellas, and M. A. Martin-Delgado, Quantum google in a complex network, Sci. Rep. **3**, 2773 (2013).
[15] A. M. Childs, E. Farhi, and S. Gutmann, An example of the difference between quantum and classical random walks, Quantum Inf. Process. **1**, 35 (2002).
[16] A. Belovs (unpublished).
[17] A. Montanaro (unpublished).
[18] T. G. Wong, Spatial search by continuous-time quantum walk with multiple marked vertices, Quantum Inf. Process. 1 (216), doi:10.1007/s11128-015-1239-y.
[19] C. Haiyan and Z. Fuji, The expected hitting times for graphs with cutpoints, Stat. Probab. Lett. **66**, 9 (2004).
[20] M. Newman, *Networks: An Introduction* (Oxford University Press, New York, 2010).
[21] E. Estrada and N. Hatano, Communicability in complex networks, Phys. Rev. E **77**, 036111 (2008).
[22] E. Estrada and J. A. Rodríguez-Velázquez, Subgraph centrality in complex networks, Phys. Rev. E **71**, 056103 (2005).
[23] L. Katz, A new status index derived from sociometric analysis, Psychometrika **18**, 39 (1953).
[24] M. Benzi and C. Klymko, On the limiting behavior of parameter-dependent network centrality measures, SIAM J. Matrix Anal. Appl. **36**, 686 (2015).
[25] L. C. Freeman, A set of measures of centrality based on betweenness, Sociometry **40**, 35 (1977).
[26] L. C. Freeman, Centrality in social networks conceptual clarification, Soc. Networks **1**, 215 (1979).