EFFICIENT QUANTUM ALGORITHMS FOR ANALYZING LARGE SPARSE ELECTRICAL NETWORKS

GUOMING WANG

Joint Center for Quantum Information and Computer Science,
University of Maryland, College Park, MD 20742, USA

Analyzing large sparse electrical networks is a fundamental task in physics, electrical engineering and computer science. We propose two classes of quantum algorithms for this task. The first class is based on solving linear systems, and the second class is based on using quantum walks. These algorithms compute various electrical quantities, including voltages, currents, dissipated powers and effective resistances, in time $\mathrm{poly}(d,c,\log(N),1/\lambda,1/\epsilon)$, where N is the number of vertices in the network, d is the maximum unweighted degree of the vertices, c is the ratio of largest to smallest edge resistance, λ is the spectral gap of the normalized Laplacian of the network, and ϵ is the accuracy. Furthermore, we show that the polynomial dependence on $1/\lambda$ is necessary. This implies that our algorithms are optimal up to polynomial factors and cannot be significantly improved.

1 Introduction

Quantum computers are believed to be more powerful than classical computers, in the sense that quantum algorithms can solve some computational problems exponentially faster than their classical counterparts. So far, this kind of speedup has been mainly demonstrated for two types of problems: simulation of quantum systems (e.g. [1, 2, 3, 4, 5]), and algebraic or number theoretic problems (e.g. [6, 7, 8, 9, 10]). While the first category is quantum in nature, the second category has some group structure so that quantum Fourier transform can be applied to find periodicity. In contrast, many computational problems in natural science and engineering do not possess this kind of structure, and it remains an important task to understand how efficiently quantum algorithms can solve them.

In this paper, we investigate the power and limitation of quantum algorithms for analyzing (resistive) electrical networks. The problems we consider are as follows. Suppose a connected undirected graph is given such that each edge has associated with it a real positive resistance, and an electric current is injected at some vertices and extracted at some other vertices. The goal is to determine the induced voltage (i.e. potential difference) between two given vertices, or the induced current on a given edge, or the total power dissipated by this graph. We are also interested in computing the *effective resistance* between two given vertices, which is defined as the induced voltage between these vertices when a unit current is injected at one of them and extracted at the other. These problems are fundamental in physics and electrical engineering, and have numerous applications. Remarkably, they play an important role in computer science as well. The idea of viewing a graph as an electrical network turns out to be very fruitful in the design of fast classical algorithms (e.g. [11, 12, 13, 14, 15, 16, 17, 18]) and analysis of random walks (e.g. [19, 20]). In

recent years, electrical network theory has also begun to be used in the design of efficient quantum algorithms (e.g. [21, 22, 23, 24, 25, 26, 27, 28, 29]) and analysis of quantum walks (e.g. [30, 26, 31, 32]). The ability to quickly compute the above electrical quantities is a requisite for these ideas to work.

Classically, one calculates the electric potentials and currents in an electrical network as follows.[a] Kirchoff's current law stipulates that the sum of the currents entering a vertex equals the sum of the currents leaving it. Ohm's law states that the voltage across a resistor equals the product of the resistance and the current through it. Combining these two facts, one obtain a system of linear equations. Currently, the best way to solve this linear system is by using Spielman and Teng's algorithm [33, 34], which takes nearly linear time in the number of edges. Note that this is nearly optimal for this approach, because simply writing down the vector of potentials (or currents) requires linear time in the number of vertices (or edges), which can be time-consuming for large graphs. Given this limitation, we naturally ask whether quantum algorithms can perform better on this task.

We answer this question affirmatively by giving a series of efficient quantum algorithms for analyzing large sparse electrical networks. These networks might contain (exponentially) many vertices, but each vertex has only a small number of neighbors (which can be efficiently found). Such networks frequently arise in both physics and computer science contexts. Our algorithms compute various electrical quantities, including voltages, currents, dissipated powers and effective resistances, in time $\text{poly}(d, c, \log(N), 1/\lambda, 1/\epsilon)$, where N is the number of vertices in the network, d is the maximum unweighted degree of the vertices, c is the ratio of largest to smallest edge resistance, λ is the spectral gap of the normalized Laplacian of the network, and ϵ is the accuracy. In particular, their dependence on N is exponentially better than that of known classical algorithms.

Our algorithms can be divided into two classes depending on the main techniques used. The first class of algorithms build certain linear systems and extract useful information from their solutions. These systems include the Laplacian system whose solution encodes the electric potentials, and another linear system whose solution roughly encodes the electric currents. To solve these systems most efficiently, we develop variants of a recent quantum linear system algorithm (QLSA) proposed by Childs, Kothari and Somma [35] (which improves the previous algorithms of Harrow, Hassidim and Lloyd [36] and Ambainis [37]). Previous QLSAs yield a quantum state proportional to the solution of a given linear system, and this has caused some controversy over the years. Our variants output a number, and hence do not have this issue. This number can be the norm of the solution, or the norm of one entry of the solution, or the norm of the difference of two entries of the solution, all of which have a natural physical meaning in the context of our linear systems.

Our second class of algorithms take advantage of the graph structure of the problems under consideration, and beat the first class in computing dissipated powers and effective resistances. They are based on a modern use of quantum walks [38, 39], which are a powerful tool for designing fast quantum algorithms. In the early years, quantum walks were mainly used for amplitude amplification in search problems (e.g. [38, 40, 39, 41]). But during recent years, their spectral properties became more and more useful for tackling decision problems (e.g. [42, 43, 25, 30]). Here we follow the second approach. Specifically, we first establish a re-

---

[a] Once these quantities are known, one can infer the dissipated powers and effective resistances from them.

lationship between the kernel of the signed weighted incidence matrix of a network and the electrical flow in this network. Then we show that a state encoding this flow can be obtained by performing a boosted version of phase estimation on the quantum walk corresponding to this matrix. We also give efficient implementations of this quantum walk. Finally, we demonstrate how to extract useful information from this state by performing appropriate operations on it.

As mentioned before, all of our algorithms have polynomial dependence on the parameter $1/\lambda$, where λ is the spectral gap of the normalized Laplacian of the network (see Section 2.2 for the precise definition). One may wonder whether this dependence is necessary. We show that this is indeed the case. Specifically, we prove that in order to estimate any of the above electrical quantities within a reasonable accuracy, one has to make $\Omega\left(1/\lambda^k\right)$ queries to the network, for some constant $k > 0$. This lower bound implies that our algorithms are optimal up to polynomial factors and cannot be dramatically improved.

The remainder of this paper is organized as follows. In Section 2, we provide some requisite background information, and formally state the problems studied in this work. In Section 3, we describe a class of quantum algorithms for analyzing electrical networks based on solving linear systems. In Section 4, we present another class of quantum algorithms for the same problems based on using quantum walks. In Section 5, we prove lower bounds on the quantum query complexity of electrical network analysis. Finally, we conclude in Section 6 with some comments and future research directions.

2 Preliminaries

In this section, we provide the necessary background information to understand this paper. In Section 2.1, we introduce the notation used in this paper. In Section 2.2 and Section 2.3, we give some basic results in spectral graph theory and electrical network theory, respectively. In Section 2.4, we formally state the problems studied in this work.

2.1 Notation

Given a set U, we use $\mathbb{R}^U$ to denote the set of all functions from U to $\mathbb{R}$. If U is finite, we also treat any $f \in \mathbb{R}^U$ as a $|U|$-dimensional vector in the natural way.

Given a real number z, we define $\mathrm{sgn}(z) = 1$ if $z \geq 0$, and -1 otherwise. Given two real numbers a, b and a real number $\delta > 0$, we say that a is a *δ-additive* approximation of b if $|a - b| \leq \delta$, and say that a is a *δ-multiplicative* approximation of b if $|a - b| \leq \delta|b|$. We say that an algorithm estimates a quantity x up to additive error δ if it outputs a δ-additive approximation of x, and say that an algorithm estimates a quantity x up to multiplicative error δ if it outputs a δ-multiplicative approximation of x.

We will use the Dirac notation to describe both quantum states and abstract vectors. Namely, depending on the context, $|\varphi\rangle$ can be a (possibly unnormalized) state or a vector in a Hilbert space, and $\langle\varphi|$ is its conjugate transpose. Moreover, if we write $|\psi\rangle \perp |\varphi\rangle$, we mean that $\langle\psi|\varphi\rangle = 0$.

Given a vector x, we use $\|x\|$ to denote the l^2 norm of x. Given a matrix A, we use $\|A\|$ to denote the spectral norm of A.

Given a matrix A, we say that A is d-sparse if each row and column of A contains at most d nonzero entries. Moreover, we use $\mathrm{Range}(A)$ to denote the range (i.e. column space) of

A, and use $\mathrm{Ker}(A)$ to denote the kernel (i.e. null space) of A. We also use $\Pi(A)$ to denote the projection onto $\mathrm{Range}(A)$, and use $\mathrm{Ref}(A)$ to denote the reflection about $\mathrm{Range}(A)$, i.e. $\mathrm{Ref}(A) := 2\Pi(A) - I$. We also use $s_j(A)$ to denote j-th smallest singular value of A (counted with multiplicity), and use $\lambda_j(A)$ to denote the j-th smallest eigenvalue of A (counted with multiplicity), starting with $j = 1$. The *condition number* of A, denoted by $\kappa(A)$, is defined as the ratio of largest to smallest singular value of A, and the *finite condition number* of A, denoted by $\kappa_f(A)$, is defined as the ratio of largest to smallest nonzero singular value of A. Futhermore, we use A^+ to denote the *Moore-Penrose pseudoinverse* of A. That is, if A has the singular value decomposition $A = \sum_j s_j |u_j\rangle\langle v_j|$ (where $s_j > 0$), then $A^+ := \sum_j s_j^{-1} |v_j\rangle\langle u_j|$.

Given two Hermitian matrices A and B, if we write $A \succcurlyeq B$ (or $A \preccurlyeq B$), we mean that $A - B$ (or $B - A$) is positive semidefinite.

Given a unitary operation U and a real number $\epsilon > 0$, we say that a circuit (or procedure) implements U with precision ϵ if this circuit (or procedure) implements a unitary operation V satisfying $\|U - V\| \le \epsilon$.

2.2   *Graph theory definitions*

All the graphs considered in this paper will be *connected*, *weighted* and *undirected*, unless otherwise stated. If any unweighted graph is mentioned, we also treat it as a weighted graph with unit edge weights.

Let $G = (V, E, w)$ be a graph with edge weights $w_e > 0$. For any vertex v, let $E(v)$ be the set of edges incident to v. The *unweighted degree* of v is defined as $\deg(v) := |E(v)|$, and the *weighted degree* of v is defined as $\widetilde{\deg}(v) := \sum_{e \in E(v)} w_e$. The *maximum unweighted degree* of G is defined as $\deg(G) := \max_{v \in V} \deg(v)$, and the *maximum weighted degree* of G is defined as $\widetilde{\deg}(G) := \max_{v \in V} \widetilde{\deg}(v)$. For any $S \subseteq V$, the *volume* of S is defined as $\mathrm{vol}(S) := \sum_{v \in S} \widetilde{\deg}(v)$.

Now we arbitrarily orient the edges in E. For each edge e, let e^+ denote its head, and let e^- denote its tail. For each vertex v, let $E^+(v) := \{e \in E(v): e^+ = v\}$, and let $E^-(v) := \{e \in E(v): e^- = v\}$. These orientations are merely for notational convenience, and they are used to interpret the meaning of a positive flow on an edge. That is, if the flow runs from the tail to the head of the edge, then it is positive; otherwise, it is negative. One should keep in mind that the graph G is still undirected, and the flow on an edge can go in either direction, regardless of this edge's orientation.

Now we define several matrices associated with the graph G. The *weighted degree matrix* of G is defined as

$$D_G := \sum_{v \in V} \widetilde{\deg}(v) |v\rangle\langle v|. \tag{1}$$

The *weighted adjacency matrix* of G is defined as

$$A_G := \sum_{e \in E} w_e \left(|e^-\rangle\langle e^+| + |e^+\rangle\langle e^-| \right). \tag{2}$$

The *signed (vertex-edge) incidence matrix* of G is defined as

$$B_G := \sum_{e \in E} \left(|e^-\rangle - |e^+\rangle \right) \langle e|. \tag{3}$$

The *edge weight matrix* of G is defined as

$$W_G := \sum_{e \in E} w_e |e\rangle\langle e|. \tag{4}$$

The *signed weighted (vertex-edge) incidence matrix* of G is defined as

$$C_G := B_G W_G^{1/2} = \sum_{e \in E} \sqrt{w_e}(|e^-\rangle - |e^+\rangle)\langle e|. \tag{5}$$

The *Laplacian* of G is defined as

$$L_G := C_G C_G^T = B_G W_G B_G^T = D_G - A_G. \tag{6}$$

The *normalized Laplacian* of G is defined as

$$\overline{L}_G := D_G^{-1/2} L_G D_G^{-1/2} = I - D_G^{-1/2} A_G D_G^{-1/2}. \tag{7}$$

Both L_G and $\overline{L}_G$ are real symmetric matrices, and they satisfy

$$\mathrm{Ker}(L_G) = \mathrm{span}\{|\mathbf{1}\rangle\}, \tag{8}$$

$$\mathrm{Ker}(\overline{L}_G) = \mathrm{span}\left\{D_G^{1/2}|\mathbf{1}\rangle\right\}, \tag{9}$$

and

$$\mathrm{Range}(L_G) = \{|\psi\rangle : |\psi\rangle \perp |\mathbf{1}\rangle\}, \tag{10}$$

$$\mathrm{Range}(\overline{L}_G) = \left\{D_G^{-1/2}|\psi\rangle : |\psi\rangle \perp |\mathbf{1}\rangle\right\}, \tag{11}$$

where $|\mathbf{1}\rangle := \sum_{v \in V} |v\rangle$. Furthermore, it can be shown that

$$0 = \lambda_1(L_G) < \lambda_2(L_G) \leq \ldots \lambda_N(L_G) \leq 2\widetilde{\deg}(G) \tag{12}$$

and

$$0 = \lambda_1(\overline{L}_G) < \lambda_2(\overline{L}_G) \leq \ldots \lambda_N(\overline{L}_G) \leq 2, \tag{13}$$

where $N := |V|$. In particular, $\lambda_2(L_G)$ is called the *spectral gap* of L_G, and $\lambda_2(\overline{L}_G)$ is called the *spectral gap* of $\overline{L}_G$.

The following lemma establishes a relationship between $\lambda_2(L_G)$ and $\lambda_2(\overline{L}_G)$:

Lemma 1 *If $\widetilde{\deg}(v) \geq 1$ for all $v \in V$, then $\lambda_2(L_G) \geq \lambda_2(\overline{L}_G)$.*

Proof. Suppose $\lambda_2(\overline{L}_G) = \lambda$. We need to show that for any $|\psi\rangle \perp |\mathbf{1}\rangle$, $\langle\psi|\psi\rangle \neq 0$,

$$\langle\psi|L_G|\psi\rangle \geq \lambda\langle\psi|\psi\rangle. \tag{14}$$

Since $|\varphi\rangle := D_G^{-1/2}|\psi\rangle \in \mathrm{Range}(\overline{L}_G)$ and $\lambda_2(\overline{L}_G) = \lambda$, we have

$$\overline{L}_G \succcurlyeq \lambda \frac{|\varphi\rangle\langle\varphi|}{\langle\varphi|\varphi\rangle} = \lambda \frac{D_G^{-1/2}|\psi\rangle\langle\psi|D_G^{-1/2}}{\langle\psi|D_G^{-1}|\psi\rangle}. \tag{15}$$

Consequently, we get

$$\langle\psi|L_G|\psi\rangle = \langle\psi|D_G^{1/2}\overline{L}_G D_G^{1/2}|\psi\rangle \geq \lambda \frac{\langle\psi|\psi\rangle\langle\psi|\psi\rangle}{\langle\psi|D_G^{-1}|\psi\rangle} \geq \lambda\langle\psi|\psi\rangle, \tag{16}$$

where the last step follows from the fact that $D_G = \sum_v \widetilde{\deg}(v)|v\rangle\langle v| \succcurlyeq I$ and hence $\langle\psi|D_G^{-1}|\psi\rangle \leq \langle\psi|\psi\rangle$. $\square$

Now we define a combinatorial quantity associated with the graph G. For any $S, T \subset V$, $S \cap T = \emptyset$, let $E(S,T)$ be the set of edges with one endpoint in S and another endpoint in T, and let $w(S,T) := \sum_{e \in E(S,T)} w_e$. Then for any $S \subset V$, $S \neq \emptyset$, the *conductance* of S is defined as

$$\phi_S := \frac{w(S,\bar{S})}{\min\{\mathrm{vol}(S), \mathrm{vol}(\bar{S})\}}, \tag{17}$$

where $\bar{S} := V \setminus S$. Then the *conductance* of G is defined as

$$\phi_G := \min_{S \subset V,\ S \neq \emptyset} \phi_S. \tag{18}$$

Remarkably, Cheeger's inequality [44, 45] establishes a polynomial relationship between the algebraic quantity $\lambda_2(\overline{L}_G)$ and the combinatorial quantity ϕ_G:

$$\frac{\phi_G^2}{2} \leq \lambda_2(\overline{L}_G) \leq 2\phi_G. \tag{19}$$

2.3   Electrical flows

In this paper, we treat a graph $G = (V, E, w)$ with edge weights $w_e > 0$ as an electrical network with the same topology and edge resistances $r_e := 1/w_e$ (or equivalently, edge conductances w_e), and vice versa. So from now on we will interchange the terms "graph" and "electrical network", as they refer to the same thing.

Let $\mathbf{i}_{ext} \in \mathbb{R}^V$ satisfy $\mathbf{i}_{ext} \perp \mathbf{1} := (1, 1, \ldots, 1)^T$. Suppose we inject an electric current of value $\mathbf{i}_{ext}(v)$ at vertex v, for each $v \in V$ (if $\mathbf{i}_{ext} < 0$, then we extract an electric current of value $-\mathbf{i}_{ext}(v)$ at vertex v). The condition $\mathbf{i}_{ext} \perp \mathbf{1}$ ensures that the total amount of injected currents equals the total amount of extracted currents, which is physically reasonable. Let $\mathbf{v} \in \mathbb{R}^V$ be the induced potentials at the vertices, and let $\mathbf{i} \in \mathbb{R}^E$ be the induced currents on the edges. By Ohm's law, the current on an edge is equal to the voltage (i.e. potential difference) between its endpoints times its conductance:

$$\mathbf{i} = W_G B_G^T \mathbf{v}. \tag{20}$$

Meanwhile, by Kirchoff's current law, the sum of the currents leaving a vertex is equal to the amount injected at the vertex:

$$B_G \mathbf{i} = \mathbf{i}_{ext}. \tag{21}$$

Combining these two facts, we get

$$\mathbf{i}_{ext} = B_G W_G B_G^T \mathbf{v} = L_G \mathbf{v}. \tag{22}$$

Since $\mathbf{i}_{ext} \in \mathrm{Range}(L_G)$, we have

$$\mathbf{v} = L_G^+ \mathbf{i}_{ext}. \tag{23}$$

Furthermore, by Joule's first law, the power dissipated by an edge is equal to the square of the current on it times its resistance (or equivalently, the square of the voltage across it times its conductance). So the total power dissipated by the graph G is

$$\mathcal{E}(\mathbf{i}) := \mathbf{i}^T W_G^{-1} \mathbf{i} = \mathbf{v}^T L_G \mathbf{v} = \mathbf{i}_{ext}^T L_G^+ \mathbf{i}_{ext}. \tag{24}$$

Often we are interested in the special case where a unit current is injected at a vertex s and extracted at another vertex t. Namely, $\mathbf{i}_{ext} = \chi_{s,t} := |s\rangle - |t\rangle$. The *effective resistance* between s and t, denoted by $\mathrm{R}_{\mathrm{eff}}(s,t)$, is defined as the induced voltage between s and t in this case. By Joule's first law, $\mathrm{R}_{\mathrm{eff}}(s,t)$ is also equal to the power dissipated by the graph G in this case. So we have

$$\mathrm{R}_{\mathrm{eff}}(s,t) = \mathbf{v}(s) - \mathbf{v}(t) = \mathcal{E}(\mathbf{i}) = \chi_{s,t}^T L_G^+ \chi_{s,t}, \tag{25}$$

where $\mathbf{v} = L_G^+ \chi_{s,t}$ and $\mathbf{i} = W_G B_G^T \mathbf{v} = W_G B_G^T L_G^+ \chi_{s,t}$.

There is an alternative definition for electrical flow which turns out to be very useful. Let $\mathbf{i}_{ext} \in \mathbb{R}^V$ satisfy $\mathbf{i}_{ext} \perp \mathbf{1} = (1,1,\dots,1)^T$. We say that $\mathbf{f} \in \mathbb{R}^E$ is a flow *consistent* with $\mathbf{i}_{ext}$ if it obeys the flow-conservation constraints:

$$\sum_{e \in E^-(v)} \mathbf{f}(e) - \sum_{e \in E^+(v)} \mathbf{f}(e) = \mathbf{i}_{ext}(v), \quad \forall\, v \in V. \tag{26}$$

The *power* of the flow $\mathbf{f}$ (with respect to the edge resistances $r_e = 1/w_e$) is defined as

$$\mathcal{E}(\mathbf{f}) := \sum_{e \in E} r_e \mathbf{f}^2(e) = \sum_{e \in E} \frac{\mathbf{f}^2(e)}{w_e}. \tag{27}$$

Then, among all the flows consistent with $\mathbf{i}_{ext}$, the electrical flow $\mathbf{i}$ induced by $\mathbf{i}_{ext}$ is the unique flow that minimizes this power function:

Lemma 2 *Let $\mathbf{i} = W_G B_G^T L_G^+ \mathbf{i}_{ext}$ be the electrical flow induced by $\mathbf{i}_{ext}$. Then any flow $\mathbf{f} \in \mathbb{R}^E$ consistent with $\mathbf{i}_{ext}$ satisfies $\mathcal{E}(\mathbf{f}) \geq \mathcal{E}(\mathbf{i})$.*

Proof. Suppose C_G has the singular value decomposition $C_G = \sum_j s_j |u_j\rangle\langle v_j|$, where $s_j > 0$, $|u_j\rangle$ and $|v_j\rangle$ are real vectors, for all j. Then $L_G = C_G C_G^T$ has the spectral decomposition $L_G = \sum_j s_j^2 |u_j\rangle\langle u_j|$. In addition, since $\mathbf{i}_{ext} \in \mathrm{Range}(L_G)$, we have $\mathbf{i}_{ext} = \sum_j \alpha_j |u_j\rangle$ for some numbers α_j's. Consequently, we get

$$W_G^{-1/2} \mathbf{i} = W_G^{1/2} B_G^T L_G^+ \mathbf{i}_{ext} = C_G^T L_G^+ \mathbf{i}_{ext} = \sum_j s_j^{-1} \alpha_j |v_j\rangle. \tag{28}$$

So the power of the electrical flow $\mathbf{i}$ is

$$\mathcal{E}(\mathbf{i}) = \left\| W_G^{-1/2} \mathbf{i} \right\|^2 = \sum_j \left| s_j^{-1} \alpha_j \right|^2. \tag{29}$$

Meanwhile, for any flow $\mathbf{f}$ consistent with $\mathbf{i}_{ext}$, we have $B_G \mathbf{f} = \mathbf{i}_{ext}$ and hence $C_G(W_G^{-1/2}\mathbf{f}) = \mathbf{i}_{ext}$. This implies that

$$W_G^{-1/2} \mathbf{f} = \sum_j s_j^{-1} \alpha_j |v_j\rangle + |\Phi^\perp\rangle, \tag{30}$$

where $|\Phi^\perp\rangle$ is an unnormalized vector satisfying $|\Phi^\perp\rangle \perp |v_j\rangle$ for all j. As a result, we obtain

$$\mathcal{E}(\mathbf{f}) = \left\| W^{-1/2}\mathbf{f} \right\|^2 = \sum_j \left| s_j^{-1} \alpha_j \right|^2 + \left\| |\Phi^\perp\rangle \right\|^2 \geq \mathcal{E}(\mathbf{i}). \tag{31}$$

$\square$.

Lemma 2 implies that the effective resistance $\mathrm{R}_{\mathrm{eff}}(s,t)$ between s and t is equal to the minimum power of a flow consistent with $\mathbf{i}_{ext} = \chi_{s,t}$. This is an alternative definition of effective resistance.

2.4   Problem statement

Given an electrical network $G = (V, E, w)$ driven by an external current $\mathbf{i}_{ext}$, we are interested in the quantum complexity of the following problems:

- Compute the voltage between two vertices s and t.

- Compute the current on an edge e.

- Compute the power dissipated by the graph G.

- Compute the effective resistance between two vertices s and t.

We will mainly focus on large sparse graphs. Namely, G might contain (exponentially) many vertices, but each vertex has only a small number of neighbors (which can be efficiently found). Our model is as follows. Suppose $V = \{v_1, v_2, \ldots, v_N\}$, $E = \{e_1, e_2, \ldots, e_M\}$ and $\deg(G) = d$. Then we assume there exists a procedure $\mathcal{P}_v$ that, on input $(i, k) \in \{1, 2, \ldots, N\} \times \{1, 2, \ldots, d\}$, outputs (the index of) the k-th edge incident to v_i. We also assume there exists a procedure $\mathcal{P}_e$ that, on input $j \in \{1, 2, \ldots, M\}$, outputs (the index of) the two endpoints of e_j as well as the weight of e_j. Furthermore, except for computing effective resistances, we assume there exists a produre $\mathcal{P}_i$ that prepares the state $\frac{|\mathbf{i}_{ext}\rangle}{\||\mathbf{i}_{ext}\rangle\|}$, where $|\mathbf{i}_{ext}\rangle := \sum_{v \in V} \mathbf{i}_{ext}(v)|v\rangle$. We assume that $\mathcal{P}_v$, $\mathcal{P}_e$ and $\mathcal{P}_i$ are all efficient, in the sense that they can be implemented in time $\mathrm{poly}(\log(N))$.

Formally, we define our Electrical Network Analysis (ENA) problems as follows:

Problem 1 (ENA-V) *Let $G = (V, E, w)$ be an electrical network such that $|V| = N$, $\deg(G) \leq d$, $1 \leq w_e \leq c$, for all $e \in E$, and $\lambda_2(\overline{L}_G) \geq \lambda > 0$. Suppose G is driven by an external current $\mathbf{i}_{ext} \in \mathbb{R}^V$ satisfying $\mathbf{i}_{ext} \perp \mathbf{1}$ and $\|\mathbf{i}_{ext}\| = 1$ [b] Let $\mathbf{v} = L_G^+ \mathbf{i}_{ext}$ be the induced potentials at the vertices, and let $\mathbf{i} = W_G B_G^T \mathbf{v}$ be the induced currents on the edges. Let $\epsilon \in (0, 1)$. Given $s, t \in V$ and access to the procedures $\mathcal{P}_v$, $\mathcal{P}_e$ and $\mathcal{P}_i$, the goal is to estimate $|\mathbf{v}(s) - \mathbf{v}(t)|$ up to additive error ϵ, succeeding with probability at least $2/3$.*

Problem 2 (ENA-C) *The assumption is the same as in ENA-V. Given $e \in E$ and access to the procedures $\mathcal{P}_v$, $\mathcal{P}_e$ and $\mathcal{P}_i$, the goal is to estimate $|\mathbf{i}(e)|$ up to additive error ϵ, succeeding with probability at least $2/3$.*

Problem 3 (ENA-P) *The assumption is the same as in ENA-V. Given access to the procedures $\mathcal{P}_v$, $\mathcal{P}_e$ and $\mathcal{P}_i$, the goal is to estimate $\mathcal{E}(\mathbf{i})$ up to multiplicative error ϵ, succeeding with probability at least $2/3$.*

Problem 4 (ENA-ER) *The assumption is almost the same as in ENA-V, except that we do not need $\mathbf{i}_{ext}$ or $\mathcal{P}_i$. Given $s, t \in V$ and access to the procedures $\mathcal{P}_v$ and $\mathcal{P}_e$, the goal is to estimate $R_{eff}(s, t)$ up to multiplicative error ϵ, succeeding with probability at least $2/3$.*

These problems are not completely independent of each other. For example, when s and t are adjacent vertices, we can infer the voltage between s and t from the current on (s, t) by using Ohm's law, and vice versa. So ENA-V and ENA-C are equivalent (up to a query of the weight of (s, t)) in this case. Moreover, when $\mathbf{i}_{ext} = \chi_{s,t}$, the power of the flow $\mathbf{i}$ is equal to the effective resistance $R_{eff}(s, t)$ between s and t. So ENA-ER can be viewed as a special case

---

[b] For the readers who have skipped Section 2.1, $\|\mathbf{i}_{ext}\|$ means the l^2 norm of $\mathbf{i}_{ext}$.

of ENA-P. Despite such connections among these problems, one can see that no two of them are completely equivalent.

Although in the above problems we assume that the edge conductances are in the range $[1, c]$ and the external current $\mathbf{i}_{ext}$ has unit l^2 norm, this is without loss of generality. Suppose instead that the edge conductances are in the range $[a, ac]$, and $\|\mathbf{i}_{ext}\| = b$, for some constants $a, b > 0$. Namely, the edge conductances are rescaled by a factor of a, and the external current is rescaled by a factor of b. By Eqs. (20), (23), (24) and (25), this would rescale the voltages, currents, disspated powers and effective resistances in the electrical network by a factor of b/a, b, b^2/a and $1/a$, respectively. So we only need to solve the problems ENA-V, ENA-C, ENA-P and ENA-ER described above, and then multiply their solutions by these factors, resepectively.

We will develop quantum algorithms for solving the above ENA problems. We quantify the resource requirements of these algorithms using two measures. The *query complexity* is the number of uses of the procedures $\mathcal{P}_v$, $\mathcal{P}_e$ and $\mathcal{P}_i$ in the algorithm. The *gate complexity* is the number of 2-qubit gates used in the algorithm. An algorithm is *gate-efficient* if its gate complexity is larger than its query complexity only by a logarithmic factor. Formally, an algorithm with query complexity Q is gate-efficient if its gate complexity is $O(Q \cdot \mathrm{poly}(\log(QN)))$, where $N = |V|$ is the number of vertices in G. All the algorithms presented in this paper will be gate-efficient.

3 Analyzing Electrical Networks by Solving Linear Systems

In this section, we describe a class of quantum algorithms for analyzing electrical networks based on solving certain linear systems. These systems include the Laplacian system whose solution encodes the electric potentials, and another linear system whose solution roughly encodes the electric currents. To solve these systems most efficiently, we first develop several variants of a recent quantum linear system algorithm in Section 3.1. Then we show how to use them to solve the ENA problems in Section 3.2.

3.1 *Quantum linear system algorithms*

Recently, Childs, Kothari and Somma (CKS) [35] proposed a quantum linear system algorithm (QLSA) which improves the previous algorithms of Harrow, Hassidim and Lloyd (HHL) [36] and Ambainis [37]. Their main result can be summarized as follows:

Theorem 1 ([35]) *Let A be a d-sparse $N \times N$ Hermitian matrix such that all the eigenvalues of A are in the range $D_\kappa := [-1, -1/\kappa] \cup [1/\kappa, 1]$. Assume there exists a procedure $\mathcal{P}_A$ that runs in time $\mathrm{poly}(\log(N))$ and on input $(i, j) \in \{1, 2, \ldots, N\} \times \{1, 2, \ldots, d\}$, outputs the location and value of the j-th nonzero entry in the i-th row of A. Let $\vec{b} = (b_1, b_2, \ldots, b_N)^T$ be an N-dimensional vector. Assume there exists a procedure $\mathcal{P}_b$ that runs in time $\mathrm{poly}(\log(N))$ and produces the state $\left| \bar{b} \right\rangle := \frac{\sum_j b_j |j\rangle}{\left\| \sum_j b_j |j\rangle \right\|}$. Let $\vec{x} = (x_1, x_2, \ldots, x_N)^T := A^{-1}\vec{b}$, and let $|\bar{x}\rangle := \frac{\sum_j x_j |j\rangle}{\left\| \sum_j x_j |j\rangle \right\|}$. Let $\epsilon \in (0, 1)$. Then there exists a gate-efficient ^{c}quantum algorithm that makes*

$$O\left(d\kappa \cdot \mathrm{poly}\left(\log\left(\frac{d\kappa}{\epsilon} \right) \right) \right)$$

---

^{c} In this subsection, we define *gate-efficient* algorithms as follows: An algorithm with query complexity Q is gate-efficient if its gate complexity is $O(Q \cdot \mathrm{poly}(\log(QN)))$, where N is the dimension of matrix A.

uses of $\mathcal{P}_A$ and $\mathcal{P}_b$, and produces a state ϵ-close to $|\bar{x}\rangle$ in l^2 norm, succeeding with $\Omega(1)$ probability, with a flag indicating success.

CKS mainly focused on how to prepare a state proportional to the solution of a given linear system. But for electrical network analysis, the following problems are actually more relevant: (1) Compute the norm of this solution; (2) Compute the norm of an entry of this solution; (3) Compute the norm of the difference of two entries of this solution. So we develop variants of their algorithm for solving these problems [d]

Lemma 3 *Under the same assumption as in Theorem 1, supposing $||\vec{b}|| = q$ is known, there exists a gate-efficient quantum algorithm that makes*

$$O\left(\frac{d\kappa^2}{\epsilon} \cdot \mathrm{poly}\left(\log\left(\frac{d\kappa}{\epsilon}\right)\right)\right)$$

uses of $\mathcal{P}_A$ and $\mathcal{P}_b$, and outputs an ϵ-multiplicative approximation of $\|\vec{x}\|$ with probability at least $2/3$.

Lemma 4 *Under the same assumption as in Theorem 1, supposing $||\vec{b}|| = q$ is known, there exists a gate-efficient quantum algorithm that makes*

$$O\left(\frac{dq^2\kappa^3}{\epsilon^2} \cdot \mathrm{poly}\left(\log\left(\frac{dq\kappa}{\epsilon}\right)\right)\right)$$

uses of $\mathcal{P}_A$ and $\mathcal{P}_b$, and outputs an ϵ-additive approximation of $|x_i|$, for any given $i \in \{1, 2, \ldots, N\}$, with probability at least $2/3$.

Lemma 5 *Under the same assumption as in Theorem 1, supposing $||\vec{b}|| = q$ is known, there exists a gate-efficient quantum algorithm that makes*

$$O\left(\frac{dq^2\kappa^3}{\epsilon^2} \cdot \mathrm{poly}\left(\log\left(\frac{dq\kappa}{\epsilon}\right)\right)\right)$$

uses of $\mathcal{P}_A$ and $\mathcal{P}_b$, and outputs an ϵ-additive approximation of $|x_i - x_j|$, for any given $i, j \in \{1, 2, \ldots, N\}$, with probability at least $2/3$.

Before proving these lemmas, let us briefly review the algorithm in Theorem 1. Then we show how to modify this algorithm to solve the problems in Lemmas 3, 4 and 5.

This algorithm uses the following technique to implement a linear combination of unitary operations. Let $M = \sum_j \alpha_j U_j$ be a linear combination of unitary operators U_j with $\alpha_j > 0$ for all j. Let V be any unitary operator that satisfies $V|0^m\rangle = \frac{1}{\sqrt{\alpha}} \sum_j \sqrt{\alpha_j}|j\rangle$, where m is a positive integer, $\alpha := \|\vec{\alpha}\|_1 = \sum_j \alpha_j$. Let $U := \sum_j |j\rangle\langle j| \otimes U_j$. Then $W := V^\dagger U V$ satisfies

$$W|0^m\rangle|\varphi\rangle = \frac{1}{\alpha}|0^m\rangle M|\varphi\rangle + \left|\Phi^\perp\right\rangle \tag{32}$$

$$= \left(\frac{\||M|\varphi\rangle\|}{\alpha}\right)|0^m\rangle \frac{M|\varphi\rangle}{\||M|\varphi\rangle\|} + \left|\Phi^\perp\right\rangle, \tag{33}$$

---

[d]CKS actually gave two quantum algorithms that meet the constraints of Theorem 1, one based on the Fourier approach, and another based on the Chebyshev approach. Our variants are based on the former one. Moreover, after completing this work, we realize that Ref. [46] gave an alternative proof of Lemma 3 based on a modification of HHL's algorithm.

where $|\Phi^\perp\rangle$ is an unnormalized state (depending on $|\varphi\rangle$) satisfying $(|0^m\rangle\langle 0^m| \otimes I)|\Phi^\perp\rangle = 0$, for all state $|\varphi\rangle$. Then, if we measure the first m qubits of this state in the standard basis, then with probability $\frac{\||M|\varphi\rangle\|^2}{\alpha^2}$, the outcome is 0^m and we obtain the state $\frac{M|\varphi\rangle}{\||M|\varphi\rangle\|}$.

To apply this technique to implement the operator A^{-1}, Ref. [35] finds certain α_j's and U_j's such that $A^{-1} \approx \sum_j \alpha_j U_j$ and each U_j is of the form e^{-iAt_j} for some $t_j \in \mathbb{R}$. Specifically, let $\gamma > 0$ be arbitrary, and let the function $h(x)$ be defined as

$$h(x) := \sum_{j=0}^{J-1} \sum_{k=-K}^{K} \alpha(j,k) e^{-ix\beta(j,k)}, \tag{34}$$

where

$$\alpha(j,k) := \frac{i}{\sqrt{2\pi}} k \delta_y \delta_z^2 e^{-k^2 \delta_z^2/2}, \tag{35}$$

$$\beta(j,k) := jk\delta_y\delta_z, \tag{36}$$

for some $J = \Theta((\kappa/\gamma) \cdot \log(\kappa/\gamma))$, $K = \Theta(\kappa \cdot \log(\kappa/\gamma))$, $\delta_y = \Theta\left(\gamma/\sqrt{\log(\kappa/\gamma)}\right)$ and $\delta_z = \Theta\left(1/(\kappa\sqrt{\log(\kappa/\gamma)})\right)$. Then $h(x)$ is γ-close to $1/x$ on the domain D_κ, i.e. $\left|h(x) - x^{-1}\right| \leq \gamma$ for all $x \in D_\kappa$. Then since A is a Hermitian matrix with eigenvalues in the range D_κ,

$$\left\|h(A) - A^{-1}\right\| = \left\|\sum_{j=0}^{J-1} \sum_{k=-K}^{K} \alpha(j,k) e^{-iA\beta(j,k)} - A^{-1}\right\| \leq \gamma. \tag{37}$$

It follows that

$$\left\|h(A)|\bar{b}\rangle - A^{-1}|\bar{b}\rangle\right\| = O(\gamma). \tag{38}$$

Then, since $\left\|A^{-1}|\bar{b}\rangle\right\| \geq 1$, by Lemma A.1 in Appendix A, we get

$$\left\|\frac{h(A)|\bar{b}\rangle}{\|h(A)|\bar{b}\rangle\|} - \frac{A^{-1}|\bar{b}\rangle}{\|A^{-1}|\bar{b}\rangle\|}\right\| = O(\gamma). \tag{39}$$

Furthermore, we have

$$\alpha := \sum_{j=0}^{J-1} \sum_{k=-K}^{K} |\alpha(j,k)| = \Theta\left(\kappa\sqrt{\log(\kappa/\gamma)}\right), \tag{40}$$

and

$$|\beta(j,k)| \leq JK\delta_y\delta_z = \Theta(\kappa \cdot \log(\kappa/\gamma)) \tag{41}$$

for all j, k.

Now we pick $\gamma = \Theta(\epsilon)$, and define the operators V and U corresponding to this Fourier approximation of x^{-1}. Let V be a unitary operator such that

$$V|0^m\rangle = \frac{1}{\sqrt{\alpha}} \sum_{j=0}^{J-1} \sum_{k=-K}^{K} \sqrt{|\alpha(j,k)|}|j,k\rangle, \tag{42}$$

where $m = O(\log(JK)) = O(\log(\kappa/\epsilon))$. Let U be defined as

$$U := i \sum_{j=0}^{J-1} \sum_{k=-K}^{K} |j,k\rangle \langle j,k| \otimes \mathrm{sgn}(k) e^{-iA\beta(j,k)}. \tag{43}$$

Ref. [35] shows that V can be implemented with $O(\kappa \cdot \mathrm{poly}(\log(\kappa/\epsilon)))$ 2-qubit gates, and U can be implemented with precision $\epsilon'(\leq \epsilon)$ by a gate-efficient procedure that makes $O(d\kappa \cdot \mathrm{poly}(\log(d\kappa/\epsilon')))$ uses of $\mathcal{P}_A$.

Now we define $W = V^\dagger U V$, and get

$$W|0^m\rangle|\bar{b}\rangle = \left(\frac{\||h(A)|\bar{b}\rangle\|}{\alpha} \right) |0^m\rangle \frac{h(A)|\bar{b}\rangle}{\||h(A)|\bar{b}\rangle\|} + |\Phi^\perp\rangle, \tag{44}$$

where $|\Phi^\perp\rangle$ is an unnormalized state satisfying $(|0^m\rangle\langle 0^m| \otimes I)|\Phi^\perp\rangle = 0$. Then, if we measure the first m qubits of this state in the standard basis, then with probability

$$p := \frac{\||h(A)|\bar{b}\rangle\|^2}{\alpha^2} = \Omega\left(\frac{1}{\alpha^2} \right), \tag{45}$$

(note that $\||h(A)|\bar{b}\rangle\| \geq \|A^{-1}|\bar{b}\rangle\| - O(\epsilon) \geq 1 - O(\epsilon)$ by Eq. (38) and $\gamma = \Theta(\epsilon)$), the outcome is 0^m and we obtain the state $\frac{h(A)|\bar{b}\rangle}{\||h(A)|\bar{b}\rangle\|}$, which is ϵ-close to $|\bar{x}\rangle$ in l^2 norm by Eq. (39). To raise the success probability to $\Omega(1)$, we use the standard ampltitude amplification [47], which requires

$$O\left(\frac{1}{\sqrt{p}} \right) = O(\alpha) = O\left(\kappa\sqrt{\log(\kappa/\epsilon)} \right) \tag{46}$$

repetitions of the above procedure. This means that we need to implement each U with precision

$$\epsilon' = O\left(\frac{\epsilon}{\alpha} \right) = O\left(\frac{\epsilon}{\kappa\sqrt{\log(\kappa/\epsilon)}} \right). \tag{47}$$

The resulting algorithm, denoted by $\mathcal{A}$, makes

$$O\left(d\kappa^2 \cdot \mathrm{poly}\left(\log\left(\frac{d\kappa}{\epsilon} \right) \right) \right)$$

uses of $\mathcal{P}_A$ and $\mathcal{P}_b$, and is gate-efficient. The κ-dependence can be decreased from quadratic to nearly linear by using Ambainis' variable-time amplitude amplification [37]. However, we do not need this technique to prove Lemma 3, 4 or 5.

Proof. [Proof of Lemma 3] Suppose $\||\vec{b}\|| = q$ is known. Then $\|\vec{x}\| = q\|A^{-1}|\bar{b}\rangle\|$. So in order to estimate $\|\vec{x}\|$ up to multiplicative error $O(\epsilon)$, we only need to get an $O(\epsilon)$-multiplicative approximation of $\|A^{-1}|\bar{b}\rangle\|$. To achieve this, we modify $\mathcal{A}$ by replacing amplitude amplification with amplitude estimation [47]. Specifically, we still choose $\gamma = \Theta(\epsilon)$ and define the corresponding operators V, U and W as before. Then, if we measure the first m qubits of $W|0^m\rangle|b\rangle$ in the standard basis, the probability of getting outcome 0^m is

$$p = \frac{\||h(A)|\bar{b}\rangle\|^2}{\alpha^2} = \Omega\left(\frac{1}{\alpha^2} \right), \tag{48}$$

where $\alpha = \Theta\left(\kappa\sqrt{\log(\kappa/\epsilon)}\right)$. We use amplitude estimation to obtain an $O(\epsilon)$-multiplicative approximation $\hat{p}$ of p (succeeding with probability at least 3/4). Then $\sqrt{\hat{p}}$ is an $O(\epsilon)$-multiplicative approximation of $\sqrt{p}$ (note that $1 - \delta \leq \sqrt{1-\delta} \leq \sqrt{1+\delta} \leq 1 + \delta$ for all $\delta \in [0,1]$), and hence $\alpha\sqrt{\hat{p}}$ is an $O(\epsilon)$-multiplicative approximation of $\|h(A)|\bar{b}\rangle\|$. Meanwhile, by Eq. (38), $\gamma = \Theta(\epsilon)$ and $\|A^{-1}|\bar{b}\rangle\| \geq 1$, we know that $\|h(A)|\bar{b}\rangle\|$ is an $O(\epsilon)$-multiplicative approximation of $\|A^{-1}|\bar{b}\rangle\|$. Combining these two facts, we get that $\alpha\sqrt{\hat{p}}$ is an $O(\epsilon)$-multiplicative approximation of $\|A^{-1}|\bar{b}\rangle\|$. Therefore, $q\alpha\sqrt{\hat{p}}$ is an $O(\epsilon)$-multiplicative approximation of $\|\vec{x}\|$, as desired.

Let us analyze the complexity of this algorithm. Since we want to estimate $p = \Omega(1/\alpha^2)$ up to multiplicative error $O(\epsilon)$, amplitude estimation requires

$$O\left(\frac{1}{\epsilon\sqrt{p}}\right) = O\left(\frac{\alpha}{\epsilon}\right) = O\left(\frac{\kappa\sqrt{\log(\kappa/\epsilon)}}{\epsilon}\right) \tag{49}$$

repetitions of W and $\mathcal{P}_b$. This means that we need to implement each U with precision

$$O\left(\frac{\epsilon}{\alpha}\right) = O\left(\frac{\epsilon}{\kappa\sqrt{\log(\kappa/\epsilon)}}\right). \tag{50}$$

This can be achieved by a gate-efficient procedure that makes $O(d\kappa \cdot \mathrm{poly}(\log(d\kappa/\epsilon)))$ uses of $\mathcal{P}_A$. So the resulting algorithm makes

$$O\left(\frac{d\kappa^2}{\epsilon} \cdot \mathrm{poly}\left(\log\left(\frac{d\kappa}{\epsilon}\right)\right)\right)$$

uses of $\mathcal{P}_A$ and $\mathcal{P}_b$, and is gate-efficient. $\square$

Proof. [Proof of Lemma 4] Let $|y\rangle = A^{-1}|\bar{b}\rangle$. Then $|x_i| = q|\langle i|y\rangle|$. So in order to estimate $|x_i|$ up to additive error $O(\epsilon)$, we only need to get an $O(\epsilon/q)$-additive approximation of $|\langle i|y\rangle|$. To achieve this, we choose $\gamma = \Theta(\epsilon')$ where $\epsilon' := \epsilon/q$, so that

$$\left\|h(A)|\bar{b}\rangle - A^{-1}|\bar{b}\rangle\right\| = O(\epsilon'). \tag{51}$$

Let $|y'\rangle = h(A)|\bar{b}\rangle$. Then $\||y'\rangle - |y\rangle\| = O(\epsilon')$. Next, we define the operators V, U and W corresponding to this γ. We also define a unitary operator R such that $R|0\rangle|i\rangle = |1\rangle|i\rangle$ and $R|0\rangle|i'\rangle = |0\rangle|i'\rangle$ for all $i' \neq i$. Then we have

$$RW|0^m\rangle|0\rangle|\bar{b}\rangle = \frac{\langle i|y'\rangle}{\alpha}|0^m\rangle|1\rangle|i\rangle + \sum_{i'\neq i}\frac{\langle i'|y'\rangle}{\alpha}|0^m\rangle|0\rangle|i'\rangle + |\Xi^\perp\rangle, \tag{52}$$

where $\alpha = \Theta\left(\kappa\sqrt{\log(\kappa/\epsilon')}\right)$, and $|\Xi^\perp\rangle$ is an unnormalized state satisfying $(|0^m\rangle\langle 0^m| \otimes I)|\Xi^\perp\rangle = 0$. Then, if we measure the first $m + 1$ qubits of this state in the standard basis, the probability of getting outcome $0^m 1$ is

$$p' := \frac{|\langle i|y'\rangle|^2}{\alpha^2}. \tag{53}$$

We use amplitude estimation to obtain an $O(\epsilon'')$-additive approximation $\hat{p}'$ of p' (succeeding with probability at least 3/4), where $\epsilon'' := (\epsilon')^2/\alpha^2$. Then $\sqrt{\hat{p}'}$ is an $O\left(\sqrt{\epsilon''}\right)$-additive

approximation of $\sqrt{p'}$ (note that $\sqrt{z} - \sqrt{\delta} \leq \sqrt{z-\delta} \leq \sqrt{z+\delta} \leq \sqrt{z} + \sqrt{\delta}$ for all $z \geq \delta \geq 0$), and hence

$$\big|\,|\langle i|y'\rangle| - \alpha\sqrt{\hat{p}'}\,\big| = \big|\alpha\sqrt{p'} - \alpha\sqrt{\hat{p}'}\big| = O\big(\alpha\sqrt{\epsilon''}\big) = O(\epsilon'). \tag{54}$$

Meanwhile, note that

$$\big|\langle i|y'\rangle - \langle i|y\rangle\big| \leq \big\|\,|y'\rangle - |y\rangle\big\| = O(\epsilon'). \tag{55}$$

Combining Eqs. (54) and (55), we get

$$\big|\alpha\sqrt{\hat{p}'} - |\langle i|y\rangle|\big| = O(\epsilon'). \tag{56}$$

Then, since $\epsilon = q\epsilon'$, we know that $q\alpha\sqrt{\hat{p}'}$ is an $O(\epsilon)$-additive approximation of $|x_i| = q|\langle i|y\rangle|$, as desired.

Let us analyze the complexity of this algorithm. Since we want to estimate p' up to additive error $O(\epsilon'')$ where $\epsilon'' = \Theta(\epsilon^2/(q^2\kappa^2\log(q\kappa/\epsilon)))$, amplitude estimation requires

$$O\left(\frac{1}{\epsilon''}\right) = O\left(\frac{q^2\kappa^2\log(q\kappa/\epsilon)}{\epsilon^2}\right) \tag{57}$$

repetitions of R, W and $\mathcal{P}_b$. This means that we need to implement each U with precision

$$O(\epsilon'') = O\left(\frac{\epsilon^2}{q^2\kappa^2\log(q\kappa/\epsilon)}\right). \tag{58}$$

This can be achieved by a gate-efficient procedure that makes $O(d\kappa \cdot \mathrm{poly}(\log(d\kappa q/\epsilon)))$ uses of $\mathcal{P}_A$. So the resulting algorithm makes

$$O\left(\frac{dq^2\kappa^3}{\epsilon^2} \cdot \mathrm{poly}\left(\log\left(\frac{dq\kappa}{\epsilon}\right)\right)\right)$$

uses of $\mathcal{P}_A$ and $\mathcal{P}_b$, and is gate-efficient. $\square$

Proof. [Proof of Lemma 5] The proof of this lemma is similar to that of Lemma 4. The main difference is that here we replace R with a unitary operation Q which satisfies $Q|0\rangle|-_{i,j}\rangle = |1\rangle|-_{i,j}\rangle$, $Q|0\rangle|+_{i,j}\rangle = |0\rangle|+_{i,j}\rangle$, and $Q|0\rangle|l\rangle = |0\rangle|l\rangle$ for all $l \neq i,j$, where $|\pm_{i,j}\rangle :=$ $(|i\rangle \pm |j\rangle)/\sqrt{2}$. Then we have

$$QW|0^m\rangle|0\rangle|\bar{b}\rangle \;=\; \frac{\langle -_{i,j}|y'\rangle}{\alpha}|0^m\rangle|1\rangle|-_{i,j}\rangle + \frac{\langle +_{i,j}|y'\rangle}{\alpha}|0^m\rangle|0\rangle|+_{i,j}\rangle \tag{59}$$

$$+ \sum_{l \neq i,j} \frac{\langle l|y'\rangle}{\alpha}|0^m\rangle|0\rangle|l\rangle + \big|\Xi^\perp\big\rangle, \tag{60}$$

where W is defined as in the proof of Lemma 4, $|y'\rangle = h(A)|\bar{b}\rangle$, and $|\Xi^\perp\rangle$ is an unnormalized state satisfying $(|0^m\rangle\langle 0^m| \otimes I)|\Xi^\perp\rangle = 0$. We still pick $\gamma = \Theta(\epsilon/q)$ such that $\big\|\,|y\rangle - |y'\rangle\big\| = O(\epsilon/q)$, where $|y\rangle = A^{-1}|\bar{b}\rangle$. If we measure the first $m+1$ qubits of $QW|0^m\rangle|0\rangle|\bar{b}\rangle$, then the probability of getting outcome 0^m1 is

$$p' := \frac{|\langle -_{i,j}|y'\rangle|^2}{\alpha^2}. \tag{61}$$

We use amplitude estimation to obtain an $O(\epsilon^2/(q^2\alpha^2))$-additive approximation $\hat{p}'$ of p' (succeeding with probability at least $3/4$). Then $\sqrt{\hat{p}'}$ is an $O(\epsilon/(q\alpha))$-additive approximation of $\sqrt{p'}$, and hence $\alpha\sqrt{\hat{p}'}$ is an $O(\epsilon/q)$-additive approximation of $\alpha\sqrt{p'} = |\langle -_{i,j}|y'\rangle|$. Meanwhile,

$$|\langle -_{i,j}|y'\rangle - \langle -_{i,j}|y\rangle| \leq \||y'\rangle - |y\rangle\| = O\left(\frac{\epsilon}{q}\right), \tag{62}$$

Combining these two facts, we know that $q\alpha\sqrt{\hat{p}'}$ is an $O(\epsilon)$-additive approximation of

$$q|\langle -_{i,j}|y\rangle| = \frac{|x_i - x_j|}{\sqrt{2}}. \tag{63}$$

All the parameters are on the same order as in the proof of Lemma 4. So this algorithm also makes

$$O\left(\frac{dq^2\kappa^3}{\epsilon^2} \cdot \mathrm{poly}\left(\log\left(\frac{dq\kappa}{\epsilon}\right)\right)\right)$$

uses of $\mathcal{P}_A$ and $\mathcal{P}_b$, and is gate-efficient. □.

It is worth noting that the algorithms in Theorem 1 and Lemmas 3, 4, 5 still work when A is not invertible but $\vec{b} \in \mathrm{Range}(A)$. In this case, we only need to replace A^{-1} with A^+, and replace the condition number κ of A with the finite condition number κ_f of A. This property will be useful in the next subsection.

3.2 Using QLSAs to analyze electrical networks

Now we show how to use the QLSAs in Lemmas 3, 4 and 5 to analyze electrical networks. Let $G = (V, E, w)$ be an electrical network driven by an external current $\mathbf{i}_{ext}$, where $|V| = N$, $\deg(G) = d$, $\lambda_2(\overline{L}_G) \geq \lambda > 0$, $1 \leq w_e \leq c$ for all $e \in E$, $\mathbf{i}_{ext} \perp \mathbf{1}$ and $\|\mathbf{i}_{ext}\| = 1$. Let $\mathbf{v} \in \mathbb{R}^V$ be the induced potentials at the vertices, and let $\mathbf{i} \in \mathbb{R}^E$ be the induced currents on the edges. To solve the ENA-V problem, we consider the Laplacian system

$$L_G\mathbf{v} = \mathbf{i}_{ext}. \tag{64}$$

Theorem 2 *The ENA-V problem can be solved by a gate-efficient quantum algorithm that makes*

$$O\left(\frac{cd^3}{\lambda^3\epsilon^2} \cdot \mathrm{poly}\left(\log\left(\frac{cd}{\lambda\epsilon}\right)\right)\right)$$

uses of $\mathcal{P}_v$, $\mathcal{P}_e$ and $\mathcal{P}_i$.

Proof. For any $v \in V$, we have $1 \leq \widetilde{\deg}(v) = \sum_{e\in E(v)} w_e \leq cd$. So by Eq. (12) and Lemma 1, we have

$$0 = \lambda_1(L_G) < \lambda \leq \lambda_2(L_G) \leq \cdots \leq \lambda_N(L_G) \leq 2cd. \tag{65}$$

Let $A := \frac{1}{2cd}L_G$ and $\vec{b} := \frac{1}{2cd}\mathbf{i}_{ext}$. Then we get

$$\mathbf{v} = L_G^+\mathbf{i}_{ext} = A^+\vec{b}. \tag{66}$$

Note that all the nonzero eigenvalues of A are in the range $[\lambda/(2cd), 1]$, which means that A has finite condition number $\kappa_f \leq 2cd/\lambda$. In addition, A is $(d+1)$-sparse, and given any $(i,j) \in \{1,2,\ldots,N\} \times \{1,2,\ldots,d+1\}$, we can find the location and value of the j-th nonzero

entry in the i-th row of A by making $O(d)$ uses of $\mathcal{P}_v$ and $\mathcal{P}_e$. Meanwhile, we have $\vec{b} \in$ Range(A), $g := ||\vec{b}|| = 1/(2cd)$, and we can prepare a state proportional to $\vec{b}$ by calling $\mathcal{P}_i$ once. Using these facts and Lemma 5, we know that there exists a gate-efficient quantum algorithm that estimates $|\mathbf{v}(s) - \mathbf{v}(t)|$ up to additive error ϵ, for any given $s, t \in V$, by making

$$O\left(d \cdot \frac{dg^2 \kappa_f^3}{\epsilon^2} \cdot \text{poly}\left(\log\left(\frac{dg\kappa_f}{\epsilon} \right) \right) \right) = O\left(\frac{cd^3}{\lambda^3 \epsilon^2} \cdot \text{poly}\left(\log\left(\frac{cd}{\lambda\epsilon} \right) \right) \right) \tag{67}$$

uses of $\mathcal{P}_v$, $\mathcal{P}_e$ and $\mathcal{P}_i$, as claimed. $\square$.

To solve the ENA-C, ENA-P and ENA-ER problems, we consider another linear system. This system has the advantage that the finite condition number of its coefficient matrix is the square root of that of Laplacian system. As a result, it can be solved more efficiently. Recall that $B_G \mathbf{i} = \mathbf{i}_{ext}$ and $C_G = B_G W_G^{1/2}$. So we have

$$\begin{pmatrix} 0 & C_G \\ C_G^T & 0 \end{pmatrix} \begin{pmatrix} 0 \\ W_G^{-1/2}\mathbf{i} \end{pmatrix} = \begin{pmatrix} \mathbf{i}_{ext} \\ 0 \end{pmatrix} \tag{68}$$

Furthermore, we claim:

Lemma 6

$$\begin{pmatrix} 0 & C_G \\ C_G^T & 0 \end{pmatrix}^{+} \begin{pmatrix} \mathbf{i}_{ext} \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ W_G^{-1/2}\mathbf{i} \end{pmatrix}. \tag{69}$$

Proof.   Suppose C_G has the singular value decomposition $C_G = \sum_j s_j |u_j\rangle \langle v_j|$, where $s_j > 0$, $|u_j\rangle$ and $|v_j\rangle$ are real vectors, for all j. Then $L_G = C_G C_G^T$ has the spectral decomposition $L_G = \sum_j s_j^2 |u_j\rangle \langle u_j|$. Moreover, since $\mathbf{i}_{ext} \in$ Range(L_G), we have $\mathbf{i}_{ext} = \sum_j \alpha_j |u_j\rangle$ for some numbers α_j's. It follows that

$$W_G^{-1/2}|\mathbf{i}\rangle = W_G^{1/2}B_G^T L_G^{+}|\mathbf{i}_{ext}\rangle = C_G^T L_G^{+}|\mathbf{i}_{ext}\rangle = \sum_j s_j^{-1} \alpha_j |v_j\rangle. \tag{70}$$

Meanwhile, we have

$$\begin{pmatrix} 0 & C_G \\ C_G^T & 0 \end{pmatrix} \quad = \quad |1\rangle\langle 0| \otimes C_G^T + |0\rangle\langle 1| \otimes C_G \tag{71}$$

$$= \quad \sum_j s_j (|1\rangle\langle 0| \otimes |v_j\rangle\langle u_j| + |0\rangle\langle 1| \otimes |u_j\rangle\langle v_j|) \tag{72}$$

$$= \quad \sum_j s_j |+_j\rangle\langle +_j| - \sum_j s_j |-_j\rangle\langle -_j|, \tag{73}$$

where

$$|\pm_j\rangle := \frac{|1\rangle|v_j\rangle \pm |0\rangle|u_j\rangle}{\sqrt{2}}. \tag{74}$$

We also have

$$\begin{pmatrix} \mathbf{i}_{ext} \\ 0 \end{pmatrix} = |0\rangle|\mathbf{i}_{ext}\rangle = \sum_j \alpha_j |0\rangle|u_j\rangle = \sum_j \frac{\alpha_j}{\sqrt{2}} (|+_j\rangle - |-_j\rangle). \tag{75}$$

These facts imply that

$$\begin{pmatrix} 0 & C_G \\ C_G^T & 0 \end{pmatrix}^+ \begin{pmatrix} \mathbf{i}_{ext} \\ 0 \end{pmatrix} = \sum_j \frac{\alpha_j s_j^{-1}}{\sqrt{2}} (|+_j\rangle + |-_j\rangle) \tag{76}$$

$$= \sum_j \alpha_j s_j^{-1} |1\rangle |v_j\rangle \tag{77}$$

$$= \begin{pmatrix} 0 \\ W_G^{-1/2} \mathbf{i} \end{pmatrix}. \tag{78}$$

□.

Theorem 3 *The ENA-C problem can be solved by a gate-efficient quantum algorithm that makes*

$$O\left(\frac{c^{1.5} d^{1.5}}{\lambda^{1.5} \epsilon^2} \cdot \mathrm{poly}\left(\log\left(\frac{cd}{\lambda\epsilon} \right) \right) \right)$$

uses of $\mathcal{P}_v$, $\mathcal{P}_e$ and $\mathcal{P}_i$.

Proof. Let $A := \frac{1}{\sqrt{2cd}} (|1\rangle\langle 0| \otimes C_G^T + |0\rangle\langle 1| \otimes C_G)$ [e] and $|b\rangle := \frac{1}{\sqrt{2cd}} |0\rangle |\mathbf{i}_{ext}\rangle$. Then Lemma 6 implies

$$|x\rangle := |1\rangle (W_G^{-1/2} |\mathbf{i}\rangle) = A^+ |b\rangle. \tag{79}$$

Thus, for any given $e \in E$, we have

$$|\mathbf{i}(e)| = \sqrt{w_e} |\langle 1, e | x \rangle|, \tag{80}$$

where $|1, e\rangle := |1\rangle |e\rangle$. So in order to estimate $|\mathbf{i}(e)|$ up to additive error ϵ, we only need to get an ϵ'-additive approximation of $|\langle 1, e | x \rangle|$, where $\epsilon' := \epsilon/\sqrt{c}$, since $1 \le w_e \le c$.

By the proof of Theorem 2, we know that all the nonzero eigenvalues of L_G are in the range $[\lambda, 2cd]$. Then by the proof of Lemma 6, we know that all the nonzero singular values of C_G are in the range $[\sqrt{\lambda}, \sqrt{2cd}]$, and all the nonzero eigenvalues of A are in the range $[-1, -\sqrt{\lambda/(2cd)}] \cup [\sqrt{\lambda/(2cd)}, 1]$. This implies that A has finite condition number $\kappa_f \le \sqrt{2cd/\lambda}$. Moreover, A is d-sparse, and for any given $(i,j) \in \{1, 2, \ldots, 2|E|\} \times \{1, 2, \ldots, d\}$, we can find the location and value of the j-th nonzero entry in the i-th row of A by making $O(1)$ uses of $\mathcal{P}_v$ and $\mathcal{P}_e$. Furthermore, by the proof of Lemma 6, we know that $|b\rangle \in \mathrm{Range}(A)$. We also have $g := \||b\rangle\| = 1/\sqrt{2cd}$, and we can prepare a state proportional to $|b\rangle$ by calling $\mathcal{P}_i$ once. Using these facts and Lemma 4, we know that there exists a gate-efficient quantum algorithm that estimates $|\langle 1, e | x \rangle|$ up to additive error ϵ' by making

$$O\left(\frac{dg^2 \kappa_f^3}{(\epsilon')^2} \cdot \mathrm{poly}\left(\log\left(\frac{dg\kappa_f}{\epsilon'} \right) \right) \right) = O\left(\frac{c^{1.5} d^{1.5}}{\lambda^{1.5} \epsilon^2} \cdot \mathrm{poly}\left(\log\left(\frac{cd}{\lambda\epsilon} \right) \right) \right) \tag{81}$$

uses of $\mathcal{P}_v$, $\mathcal{P}_e$ and $\mathcal{P}_i$. This concludes the proof. □.

Theorem 3 also provides a method for estimating the voltage between two *adjacent* vertices s and t:

---

[e] Here we consider A as a $2|E| \times 2|E|$ matrix with $|E| + |V|$ nonzero rows and $|E| + |V|$ nonzero columns.

Corollary 1 *Under the promise that s and t are adjacent vertices, the ENA-V problem can be solved by a gate-efficient algorithm that makes*

$$O\left(\frac{c^{1.5}d^{1.5}}{\lambda^{1.5}\epsilon^2} \cdot \text{poly}\left(\log\left(\frac{cd}{\lambda\epsilon}\right)\right)\right)$$

uses of $\mathcal{P}_v$, $\mathcal{P}_e$ and $\mathcal{P}_i$.

Proof. Let $e = (s,t) \in E$. By Ohm's law, we have $|\mathbf{v}(s) - \mathbf{v}(t)| = |\mathbf{i}(e)|/w_e$, where $1 \le w_e \le c$. So in order to estimate $|\mathbf{v}(s) - \mathbf{v}(t)|$ up to additive error ϵ, we only need to get an ϵ-additive approximation of $|\mathbf{i}(e)|$. By Theorem 3, this can be achieved by a gate-efficient quantum algorithm that makes $O((c^{1.5}d^{1.5}/(\lambda^{1.5}\epsilon^2)) \cdot \text{poly}(\log(cd/(\lambda\epsilon))))$ uses of $\mathcal{P}_v$, $\mathcal{P}_e$ and $\mathcal{P}_i$. $\square$

One can compare the algorithm in Corollary 1 with the one in Theorem 2 for solving ENA-V in the general case. The former has better dependence on d and $1/\lambda$, but slightly worse dependence on c, than the latter. So they are incomparable.

Finally, we solve the ENA-P and ENA-ER problems by utilizing the algorithm in Lemma 3:

Theorem 4 *The ENA-P problem can be solved by a gate-efficient quantum algorithm that makes*

$$O\left(\frac{cd^2}{\lambda\epsilon} \cdot \text{poly}\left(\log\left(\frac{cd}{\lambda\epsilon}\right)\right)\right)$$

uses of $\mathcal{P}_v$, $\mathcal{P}_e$ and $\mathcal{P}_i$.

Proof. Let us use the same notation as in the proof of Theorem 3. Then we have

$$\||x\rangle\|^2 = \left\|W_G^{-1/2}|\mathbf{i}\rangle\right\| = \langle\mathbf{i}|W_G^{-1}|\mathbf{i}\rangle = \mathcal{E}(\mathbf{i}). \tag{82}$$

Thus, in order to estimate $\mathcal{E}(\mathbf{i})$ up to multiplicative error $O(\epsilon)$, we only need to get an $O(\epsilon)$-multiplicative approximation of $\||x\rangle\|$ (note that $1 - 2\delta \le (1-\delta)^2 \le (1+\delta)^2 \le 1 + 3\delta$ for all $\delta \in [0,1]$). By Lemma 3, this can be accomplished by a gate-efficient quantum algorithm that makes

$$O\left(\frac{d\kappa_f^2}{\epsilon} \cdot \text{poly}\left(\log\left(\frac{d\kappa_f}{\epsilon}\right)\right)\right) = O\left(\frac{cd^2}{\lambda\epsilon} \cdot \text{poly}\left(\log\left(\frac{cd}{\lambda\epsilon}\right)\right)\right) \tag{83}$$

uses of $\mathcal{P}_v$, $\mathcal{P}_e$ and $\mathcal{P}_i$. This concludes the proof. $\square$

Corollary 2 *The ENA-ER problem can be solved by a gate-efficient quantum algorithm that makes*

$$O\left(\frac{cd^2}{\lambda\epsilon} \cdot \text{poly}\left(\log\left(\frac{cd}{\lambda\epsilon}\right)\right)\right)$$

uses of $\mathcal{P}_v$ and $\mathcal{P}_e$.

Proof. Recall that $R_{\text{eff}}(s,t) = \mathcal{E}(\mathbf{i})$, where $\mathbf{i} = W_G B_G^T L_G^+ \chi_{s,t}$ is the electrical flow induced by the external current $\chi_{s,t} = |s\rangle - |t\rangle$. Clearly, we can prepare the state $(|s\rangle - |t\rangle)/\sqrt{2}$ in time $\text{poly}(\log(N))$. Then, by Theorem 4, there is a gate-efficient algorithm that makes $O((cd^2/(\lambda\epsilon)) \cdot \text{poly}(\log(cd/(\lambda\epsilon))))$ uses of $\mathcal{P}_v$ and $\mathcal{P}_e$, and outputs an $O(\epsilon)$-multiplicative approximation of $\mathcal{E}(\tilde{\mathbf{i}}) = \mathcal{E}(\mathbf{i})/2$, where $\tilde{\mathbf{i}} = \mathbf{i}/\sqrt{2}$ is the electrical flow induced by the external current $\chi_{s,t}/\sqrt{2}$. Then we multiply this result by a factor of 2, and obtain an $O(\epsilon)$-multiplicative approximation of $\mathcal{E}(\mathbf{i}) = R_{\text{eff}}(s,t)$. $\square$

4 Analyzing Electrical Networks by Using Quantum Walks

In this section, we present a set of quantum algorithms for analyzing electrical networks based on using quantum walks. In Section 4.1, we define several graph-related matrices, and show that they have some nice properties. In Section 4.2, we quantize one of the matrices to obtain a quantum walk. Then we analyze the spectral properties of this quantum walk, and give its efficient implementations. In Section 4.3, we describe how to utilize this quantum walk to solve the ENA-P and ENA-ER problems.

4.1 Modifying graphs and defining matrices

Suppose $G = (V, E, w)$ is an electrical network driven by an external current $\mathbf{i}_{ext}$, where $|V| = N$, $\deg(G) = d$, $\lambda_2(\overline{L}_G) \geq \lambda > 0$, $1 \leq w_e \leq c$ for all $e \in E$, $\mathbf{i}_{ext} \perp \mathbf{1}$ and $\|\mathbf{i}_{ext}\| = 1$. For any $e \in E$, let

$$|\varphi_e\rangle := \sqrt{w_e}(|e^-\rangle - |e^+\rangle). \tag{84}$$

Then we have

$$C_G = B_G W_G^{1/2} = \sum_{e \in E} |\varphi_e\rangle\langle e|. \tag{85}$$

Now we modify the graph G as follows. We add a special hyperedge e_0 among all the vertices in V, and set its weight to be λ. Let $G' = (V, E', w)$ be the modified hypergraph, where $E' := E \cup \{e_0\}$, w_e is the same as before for all $e \in E$, and $w_{e_0} = \lambda$. Let

$$|\varphi_{e_0}\rangle := -\sqrt{2\lambda}|\mathbf{i}_{ext}\rangle = -\sum_{v \in V} \sqrt{2\lambda}\mathbf{i}_{ext}(v)|v\rangle. \tag{86}$$

Then we define

$$C_{G'} := \sum_{e \in E'} |\varphi_e\rangle\langle e| = C_G + |\varphi_{e_0}\rangle\langle e_0|. \tag{87}$$

Moreover, for any $v \in V$, let $E'(v) := E(v) \cup \{e_0\}$, and let $\widetilde{\deg}'(v) := \widetilde{\deg}(v) + \lambda$. Then we define

$$D_{G'} := \sum_{v \in V} \widetilde{\deg}'(v)|v\rangle\langle v|. \tag{88}$$

We also define

$$L_{G'} := C_{G'}C_{G'}^T, \tag{89}$$

and

$$\overline{L}_{G'} := D_{G'}^{-1/2}L_{G'}D_{G'}^{-1/2}. \tag{90}$$

One can easily see that

$$\text{Ker}(L_{G'}) \;\;=\;\; \text{span}\{|\mathbf{1}\rangle\}, \tag{91}$$

$$\text{Ker}(\overline{L}_{G'}) \;\;=\;\; \text{span}\left\{D_{G'}^{1/2}|\mathbf{1}\rangle\right\}, \tag{92}$$

and

$$\text{Range}(L_{G'}) \;\;=\;\; \{|\psi\rangle : |\psi\rangle \perp |\mathbf{1}\rangle\}, \tag{93}$$

$$\text{Range}(\overline{L}_{G'}) \;\;=\;\; \left\{D_{G'}^{-1/2}|\psi\rangle : |\psi\rangle \perp |\mathbf{1}\rangle\right\}, \tag{94}$$

where $|\mathbf{1}\rangle = \sum_v |v\rangle$.

The following lemma establishes a relationship between $\lambda_2(\overline{L}_G)$ and $\lambda_2(\overline{L}_{G'})$:

Lemma 7 *If $\lambda_2(\overline{L}_G) \geq \lambda > 0$, then $\lambda_2(\overline{L}_{G'}) \geq \lambda/3 > 0$.*

Proof. It is sufficient to show that, for any $|\varphi\rangle \in \mathrm{Range}(\overline{L}_{G'})$, $\langle\varphi|\varphi\rangle \neq 0$, we have

$$\langle\varphi|\overline{L}_{G'}|\varphi\rangle \geq \frac{\lambda}{3} \cdot \langle\varphi|\varphi\rangle. \tag{95}$$

Suppose $|\varphi\rangle = D_{G'}^{-1/2}|\psi\rangle$ for some $|\psi\rangle \perp |\mathbf{1}\rangle$, $\langle\psi|\psi\rangle \neq 0$. Then, since $D_G^{-1/2}|\psi\rangle \in \mathrm{Range}(\overline{L}_G)$ and $\lambda_2(\overline{L}_G) \geq \lambda$, we get

$$\overline{L}_G \succcurlyeq \lambda \frac{D_G^{-1/2}|\psi\rangle\langle\psi|D_G^{-1/2}}{\langle\psi|D_G^{-1}|\psi\rangle}. \tag{96}$$

This implies that

$$D_G^{1/2}\overline{L}_G D_G^{1/2} \succcurlyeq \lambda \frac{|\psi\rangle\langle\psi|}{\langle\psi|D_G^{-1}|\psi\rangle}. \tag{97}$$

Meanwhile, note that

$$L_{G'} = C_{G'}C_{G'}^T = \sum_{e \in E'}|\varphi_e\rangle\langle\varphi_e| \succcurlyeq \sum_{e \in E}|\varphi_e\rangle\langle\varphi_e| = C_G C_G^T = L_G. \tag{98}$$

This implies that

$$\overline{L}_{G'} = D_{G'}^{-1/2}L_{G'}D_{G'}^{-1/2} \succcurlyeq D_{G'}^{-1/2}L_G D_{G'}^{-1/2} = D_{G'}^{-1/2}D_G^{1/2}\overline{L}_G D_G^{1/2}D_{G'}^{-1/2}. \tag{99}$$

Combining Eqs. (97) and (99), we get

$$\begin{aligned}
\langle\varphi|\overline{L}_{G'}|\varphi\rangle &\geq \langle\varphi|D_{G'}^{-1/2}D_G^{1/2}\overline{L}_G D_G^{1/2}D_{G'}^{-1/2}|\varphi\rangle & (100) \\
&= \langle\psi|D_{G'}^{-1}D_G^{1/2}\overline{L}_G D_G^{1/2}D_{G'}^{-1}|\psi\rangle & (101) \\
&\geq \lambda \frac{\langle\psi|D_{G'}^{-1}|\psi\rangle\langle\psi|D_{G'}^{-1}|\psi\rangle}{\langle\psi|D_G^{-1}|\psi\rangle} & (102) \\
&\geq \lambda\langle\varphi|\varphi\rangle\frac{\langle\psi|D_{G'}^{-1}|\psi\rangle}{\langle\psi|D_G^{-1}|\psi\rangle}. & (103)
\end{aligned}$$

Suppose $|\psi\rangle = \sum_{v \in V}\alpha_v|v\rangle$ for some numbers α_v's. Then

$$\frac{\langle\psi|D_{G'}^{-1}|\psi\rangle}{\langle\psi|D_G^{-1}|\psi\rangle} = \frac{\sum\limits_{v \in V}\widetilde{\deg}'(v)^{-1}|\alpha_v|^2}{\sum\limits_{v \in V}\widetilde{\deg}(v)^{-1}|\alpha_v|^2}. \tag{104}$$

For any $v \in V$, we have $1 \leq \widetilde{\deg}(v) \leq cd$. Then, since $\lambda \leq \lambda_2(\overline{L}_G) \leq 2$, we have

$$\frac{\widetilde{\deg}(v)}{\widetilde{\deg}'(v)} = \frac{\widetilde{\deg}(v)}{\widetilde{\deg}(v) + \lambda} \geq \frac{1}{1 + \lambda} \geq \frac{1}{3}. \tag{105}$$

It follows that

$$\frac{\langle\psi|D_{G'}^{-1}|\psi\rangle}{\langle\psi|D_G^{-1}|\psi\rangle} \geq \min_{v \in V}\frac{\widetilde{\deg}'(v)^{-1}}{\widetilde{\deg}(v)^{-1}} \geq \frac{1}{3}. \tag{106}$$

Plugging this into Eq. (103) yields

$$\langle \varphi | \overline{L}_{G'} | \varphi \rangle \geq \frac{\lambda}{3} \cdot \langle \varphi | \varphi \rangle, \tag{107}$$

as desired. □.

Now let us consider $\text{Ker}(C_{G'})$. We claim:

Lemma 8

$$\text{Ker}(C_{G'}) = \{ g(\mathbf{f}) : \mathbf{f} \text{ is a flow consistent with } \alpha \cdot \mathbf{i}_{ext} \text{ for some number } \alpha \}, \tag{108}$$

where

$$g(\mathbf{f}) := \frac{1}{\sqrt{2\lambda}} h(\mathbf{f}) | e_0 \rangle + \sum_{e \in E} \frac{\mathbf{f}(e)}{\sqrt{w_e}} | e \rangle, \tag{109}$$

in which $h(\mathbf{f}) = \alpha$ if $\mathbf{f}$ is consistent with $\alpha \cdot \mathbf{i}_{ext}$.

Proof. Let $| \psi \rangle = \beta | e_0 \rangle + \sum_{e \in E} \beta_e | e \rangle$ be arbitrary. Then

$$C_{G'} | \psi \rangle = \beta | \varphi_{e_0} \rangle + \sum_{e \in E} \beta_e | \varphi_e \rangle = 0 \tag{110}$$

if and only if

$$\sum_{e \in E^-(v)} \sqrt{w_e} \beta_e - \sum_{e \in E^+(v)} \sqrt{w_e} \beta_e = \sqrt{2\lambda} \beta \cdot \mathbf{i}_{ext}(v), \quad \forall v \in V. \tag{111}$$

This is equivalent to the condition that the flow $\mathbf{f}$ defined as $\mathbf{f}(e) = \sqrt{w_e} \beta_e$ for all $e \in E$ is consistent with $\sqrt{2\lambda} \beta \cdot \mathbf{i}_{ext}$. □.

Now let Π be the projection onto $\text{Ker}(C_{G'})$. We claim:

Lemma 9

$$\Pi | e_0 \rangle \propto | \Phi \rangle := g(\mathbf{i}) = \frac{1}{\sqrt{2\lambda}} | e_0 \rangle + \sum_{e \in E} \frac{\mathbf{i}(e)}{\sqrt{w_e}} | e \rangle, \tag{112}$$

where $\mathbf{i} = W_G B_G^T L_G^+ \mathbf{i}_{ext}$ is the electrical flow induced by $\mathbf{i}_{ext}$.

We will give two proofs of Lemma 9. The first one is algebraic and more rigorous, and the second one is geometric and more intuitive.

Proof. [Proof 1 of Lemma 9] It is sufficient to show that for any $| \Psi \rangle \in \text{Ker}(C_{G'})$, if $| \Psi \rangle \perp | \Phi \rangle$, then $| \Psi \rangle \perp | e_0 \rangle$. We prove its contrapositive by contradiction.

Suppose $| \Psi \rangle \in \text{Ker}(C_{G'})$ satisfies $| \Psi \rangle \not\perp | e_0 \rangle$. By Lemma 8, after appropriate rescaling of $| \Psi \rangle$, we can write it as

$$| \Psi \rangle = g(\mathbf{f}) = \frac{1}{\sqrt{2\lambda}} | e_0 \rangle + \sum_{e \in E} \frac{\mathbf{f}(e)}{\sqrt{w_e}} | e \rangle \tag{113}$$

for some flow $\mathbf{f}$ consistent with $\mathbf{i}_{ext}$. We claim that, if

$$| \Psi \rangle \perp | \Phi \rangle = g(\mathbf{i}) = \frac{1}{\sqrt{2\lambda}} | e_0 \rangle + \sum_{e \in E} \frac{\mathbf{i}(e)}{\sqrt{w_e}} | e \rangle, \tag{114}$$

which means that

$$-\frac{1}{2\lambda} = \sum_{e \in E} \frac{\mathbf{f}(e) \mathbf{i}(e)}{w_e} < 0, \tag{115}$$

then there exists a flow $\mathbf{f}' \neq \mathbf{i}$ consistent with $\mathbf{i}_{ext}$ such that $\mathcal{E}(\mathbf{f}') < \mathcal{E}(\mathbf{i})$. But this is contradictory to Lemma 2 which states that $\mathbf{i}$ has the minimum power among such flows! Consequently, we must have $|\Psi\rangle \not\perp |\Phi\rangle$.

Now we prove this claim. Let $\mathbf{f}' = \beta\mathbf{f} + (1-\beta)\mathbf{i}$, where $\beta \in (0,1)$ is to be chosen later. Obviously, $\mathbf{f}'$ is a flow consistent with $\mathbf{i}_{ext}$ for any choice of β. Let us consider the power of $\mathbf{f}'$.

$$\mathcal{E}(\mathbf{f}') = \sum_{e \in E} \frac{(\beta\mathbf{f}(e) + (1-\beta)\mathbf{i}(e))^2}{w_e} \tag{116}$$

$$= \beta^2 \sum_{e \in E} \frac{\mathbf{f}(e)^2}{w_e} + (1-\beta)^2 \sum_{e \in E} \frac{\mathbf{i}(e)^2}{w_e} + 2\beta(1-\beta) \sum_{e \in E} \frac{\mathbf{f}(e)\mathbf{i}(e)}{w_e} \tag{117}$$

$$< \beta^2 \mathcal{E}(\mathbf{f}) + (1-\beta)^2 \mathcal{E}(\mathbf{i}). \tag{118}$$

Now let $\gamma = \dfrac{\mathcal{E}(\mathbf{f})}{\mathcal{E}(\mathbf{i})}$ and $\beta = \dfrac{1}{1+\gamma}$. Then we get

$$\mathcal{E}(\mathbf{f}') < (\beta^2\gamma + (1-\beta)^2)\mathcal{E}(\mathbf{i}) = \frac{\gamma}{1+\gamma}\mathcal{E}(\mathbf{i}) < \mathcal{E}(\mathbf{i}), \tag{119}$$

as claimed. $\square$.

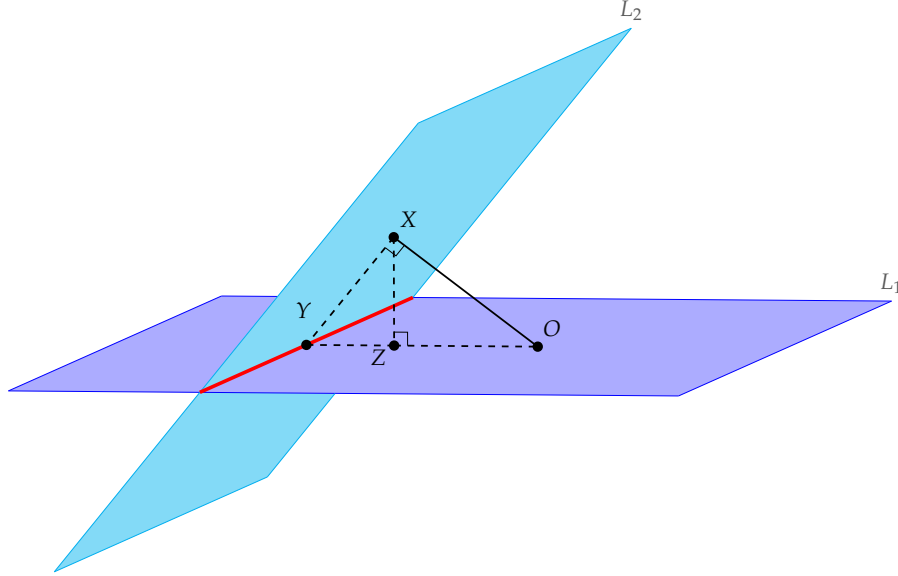**Proof.** [Proof 2 of Lemma 9] Consider the geometric picture shown in Fig.1.

Fig. 1.   Geometric proof of Lemma 9.   Here L_1 and L_2 are two hyperplanes in the space $\mathcal{H} = \mathrm{span}\{|e\rangle : e \in E'\}$.   They are defined as $L_1 = \mathrm{Ker}(C_{G'})$ and $L_2 = \{|\Psi\rangle \in \mathcal{H} : \langle e_0|\Psi\rangle = a\}$, where $a = 1/\sqrt{2\lambda}$. Moreover, O is the origin of $\mathcal{H}$, and X is the point in $\mathcal{H}$ such that $\overrightarrow{OX} = a|e_0\rangle$. Note that $\overrightarrow{OX}$ is perpendicular to L_2. The red line denotes the intersection of L_1 and L_2, and the points on this line correspond to the flows consistent with $\mathbf{i}_{ext}$. Let Y be an arbitrary point on the red line. Then we have $\overrightarrow{OX} \perp \overrightarrow{XY}$. Furthermore, let Z be the unique point on the line $\overline{OY}$ such that $\overrightarrow{XZ} \perp \overrightarrow{OY}$. Then we can show that $\|\overrightarrow{XZ}\|$ achieves the minimum value if and only if Y corresponds to the electrical flow consistent with $\mathbf{i}_{ext}$, and in this case, Z is exactly the projection of X onto L_1.

Let $\mathcal{H} := \text{span}\{|e\rangle : e \in E'\}$. For any $A \in \mathcal{H}$, we view A as both a vector and a point, and for any $A, B \in \mathcal{H}$, we define $\overrightarrow{AB}$ as the vector from the point A to the point B. Let O be the origin of $\mathcal{H}$, and let X be the point in $\mathcal{H}$ such that $\overrightarrow{OX} = a|e_0\rangle$, where $a := 1/\sqrt{2\lambda}$. Then $L_1 := \text{Ker}(C_{G'})$ is a hyperplane in $\mathcal{H}$. Moreover, let $L_2 := \{|\Psi\rangle \in \mathcal{H} : \langle e_0|\Psi\rangle = a\}$. Then, for any $|\psi\rangle \in L_2$, we can write it as $|\Psi\rangle = a|e_0\rangle + |\Psi'\rangle$ for some vector $|\Psi'\rangle \perp |e_0\rangle$. So L_2 is a hyperplane orthogonal to the vector $\overrightarrow{OX}$ and it also touches the point X.

Now consider $L_3 := L_1 \cap L_2$. One can see that

$$L_3 = \left\{ g(\mathbf{f}) = a|e_0\rangle + \sum_{e \in E} \frac{\mathbf{f}(e)}{\sqrt{\widetilde{w}_e}}|e\rangle : \mathbf{f} \text{ is a flow consistent with } \mathbf{i}_{ext} \right\}. \tag{120}$$

So the points in L_3 correspond to the flows consistent with $\mathbf{i}_{ext}$.

Let us pick arbitrary $Y \in L_3$. Then $\overrightarrow{OY} = g(\mathbf{f})$ for some flow $\mathbf{f}$ consistent with $\mathbf{i}_{ext}$. Note that $\overrightarrow{XY} = \overrightarrow{OY} - \overrightarrow{OX} = \sum_{e \in E} \frac{\mathbf{f}(e)}{\sqrt{\widetilde{w}_e}}|e\rangle$ and hence $\left\|\overrightarrow{XY}\right\|^2 = \mathcal{E}(\mathbf{f})$. Then, since $\overrightarrow{OX} \perp \overrightarrow{XY}$ and $\left\|\overrightarrow{OX}\right\| = a$, we get $\left\|\overrightarrow{OY}\right\|^2 = a^2 + \mathcal{E}(\mathbf{f})$. Now let Z be the unique point in the line $\overline{OY}$ that is closest to X. Then we have $\overrightarrow{XZ} \perp \overrightarrow{OY}$, and hence $\left\|\overrightarrow{XZ}\right\| = a\sqrt{\mathcal{E}(\mathbf{f})/(a^2 + \mathcal{E}(\mathbf{f}))}$. Note that $\left\|\overrightarrow{XZ}\right\|$ achieves the minimum value if and only if $\mathcal{E}(\mathbf{f})$ achieves the minimum value. By Lemma 2, the electrical flow $\mathbf{i}$ has the minimum power among all the flows consistent with $\mathbf{i}_{ext}$. So $\left\|\overrightarrow{XZ}\right\|$ achieves the minimum value if and only if Y corresponds to the electrical flow $\mathbf{i}$, i.e. $\overrightarrow{OY} = g(\mathbf{i})$. Then the corresponding Z is the point closest to X in L_1. In other words, this Z is exactly the projection of X onto L_1. So we have $\Pi|e_0\rangle = \overrightarrow{OZ} \propto \overrightarrow{OY} = g(\mathbf{i})$, as claimed. $\square$

4.2 Defining quantum walks

Now we define a quantum walk [38, 39] related to the matrix $C_{G'}$, analyze its spectral properties, and give its efficient implementations. This quantum walk will become a key component of the algorithms in the next subsection. It can be viewed as a generalization of those used for evaluating span programs [42, 43, 25].

Let us define two operators A and B as follows:

$$A := \sum_{v \in V} |\psi_v\rangle|v\rangle\langle v|, \tag{121}$$

$$B := \sum_{e \in E'} |e\rangle|\phi_e\rangle\langle e|, \tag{122}$$

where

$$|\psi_v\rangle := \frac{1}{\sqrt{\widetilde{\deg}'(v)}} \sum_{e \in E'(v)} \sqrt{\widetilde{w}_e}|e\rangle \tag{123}$$

$$= \frac{1}{\sqrt{\widetilde{\deg}(v) + \lambda}} \left(\sqrt{\lambda}|e_0\rangle + \sum_{e \in E(v)} \sqrt{\widetilde{w}_e}|e\rangle \right), \quad \forall v \in V, \tag{124}$$

and

$$|\phi_e\rangle \quad := \quad \frac{1}{\sqrt{2}}\left(|e^+\rangle - |e^-\rangle\right), \quad \forall e \in E, \tag{125}$$

$$|\phi_{e_0}\rangle \quad := \quad |\mathbf{i}_{ext}\rangle \;=\; \sum_{v \in V} \mathbf{i}_{ext}(v)|v\rangle. \tag{126}$$

Note that the $|\psi_v\rangle$'s and $|\phi_e\rangle$'s are all unit vectors (recall that $\|\mathbf{i}_{ext}\| = 1$). So A and B are both isometries, and

$$AA^\dagger \quad = \quad \Pi(A), \tag{127}$$

$$BB^\dagger \quad = \quad \Pi(B). \tag{128}$$

Furthermore, by a direct computation, one can check that

$$D(A,B) := A^\dagger B = -\frac{1}{\sqrt{2}}D_{G'}^{-1/2}C_{G'}. \tag{129}$$

This implies that $\mathrm{Ker}(D(A,B)) = \mathrm{Ker}(C_{G'})$, which is characterized by Lemma 8.

Now we define a unitary operator $U(A,B)$ as follows:

$$U(A,B) := \mathrm{Ref}(B) \cdot \mathrm{Ref}(A). \tag{130}$$

We can find the eigenvalues and eigenvectors of $U(A,B)$ by using Szegedy's spectral lemma:

Lemma 10 (Spectral Lemma, [39]) *Let $\mathcal{H}$ be a Hilbert space, and let A, B be two operators such that $AA^\dagger = \Pi(A)$ and $BB^\dagger = \Pi(B)$ are both projections onto subspaces of $\mathcal{H}$. Let $D(A,B) := A^\dagger B$ and let $U(A,B) := \mathrm{Ref}(B) \cdot \mathrm{Ref}(A)$. Then all the singular values of $D(A,B)$ are at most 1. Let $\{\cos\theta_j : 1 \le j \le k\}$ be the singular values of $D(A,B)$ that lie in the open interval $(0,1)$ (counted with multiplicity), and let $\{|w_j\rangle, |u_j\rangle : 1 \le j \le k\}$ be the associated left and right singular vectors. Then those eigenvalues of $U(A,B)$ that have nonzero imaginary part are exactly*

$$\{e^{-2i\theta_j}, e^{2i\theta_j} : 1 \le j \le k\}. \tag{131}$$

The (unnormalized) eigenvectors associated with these eigenvalues are

$$\{A|w_j\rangle - e^{-i\theta_j}B|u_j\rangle, \; A|w_j\rangle - e^{i\theta_j}B|u_j\rangle : 1 \le j \le k\}. \tag{132}$$

Furthermore,

1. *The $+1$ eigenspace of $U(A,B)$ is $(\mathrm{Range}(A) \cap \mathrm{Range}(B)) \oplus \left(\mathrm{Range}(A)^\perp \cap \mathrm{Range}(B)^\perp\right)$.*

2. *The -1 eigenspace of $U(A,B)$ is $\left(\mathrm{Range}(A) \cap \mathrm{Range}(B)^\perp\right) \oplus \left(\mathrm{Range}(A)^\perp \cap \mathrm{Range}(B)\right)$. In addition, $\mathrm{Range}(A)^\perp \cap \mathrm{Range}(B) = \{B|u\rangle : |u\rangle \in \mathrm{Ker}(D(A,B))\}$.*

The above is a complete description of the eigenvalues and eigenvectors of operator $U(A,B)$ acting on $\mathcal{H}$.

We are interested in the -1 eigenspace of $U(A, B)$. Let $\mathcal{H}'$ be this subspace, and let Π' be the projection onto this subspace. By Lemma 10, $\mathcal{H}'$ is the direct sum of $\mathcal{H}'' := \{B|u\rangle : |u\rangle \in \mathrm{Ker}(D(A, B))\}$ and another subspace which is orthogonal to $\mathrm{Range}(B)$. Then, since B is an isometry, Lemma 9 implies that

$$\Pi' B|e_0\rangle \propto B|\Phi\rangle = aB|e_0\rangle + BW_G^{-1/2}|\mathbf{i}\rangle = a|\phi_{e_0}\rangle|e_0\rangle + \sum_{e \in E} \frac{\mathbf{i}(e)}{\sqrt{w_e}}|\phi_e\rangle|e\rangle, \tag{133}$$

where $a := 1/\sqrt{2\lambda}$. This fact will be useful in the next subsection.

The following lemma gives a lower bound on the eigenphase gap around π of $U(A, B)$:

Lemma 11 *The eigenphase gap around π of $U(A, B)$ is at least $\sqrt{2\lambda/3}$.*

Proof. Since $\lambda_2(\overline{L}_G) \geq \lambda > 0$, by Lemma 7, we have $\lambda_2(\overline{L}_{G'}) \geq \lambda/3 > 0$. Then, by

$$D(A, B)D(A, B)^{\dagger} = \frac{1}{2}D_{G'}^{-1/2}C_{G'}C_{G'}^T D_{G'}^{-1/2} = \frac{1}{2}\overline{L}_{G'}, \tag{134}$$

we get $s_2(D(A, B)) \geq \sqrt{\lambda/6}$. Meanwhile, by Lemma 10, the singular value $s_j \in (0, 1)$ of $D(A, B)$ is mapped to the eigenvalues $e^{\pm 2i \arccos s_j}$ of $U(A, B)$. Let $\theta_j = \pi/2 - \arccos s_j$. Then we have

$$\theta_j \geq \sin \theta_j = s_j \geq \sqrt{\frac{\lambda}{6}}. \tag{135}$$

Therefore, the eigenphase gap around π of $U(A, B)$ is at least $2\sqrt{\lambda/6} = \sqrt{2\lambda/3}$, as claimed. $\square$.

The following lemmas give upper bounds on the cost of implementing $U(A, B)$ perfectly or approximately:

Lemma 12 $U(A, B)$ *can be implemented by a gate-efficient procedure that makes $O(d)$ uses of $\mathcal{P}_v$, $\mathcal{P}_e$ and $\mathcal{P}_i$.*

Proof. Since $U(A, B) = \mathrm{Ref}(B) \cdot \mathrm{Ref}(A)$, we only need to show that both $\mathrm{Ref}(A)$ and $\mathrm{Ref}(B)$ can be implemented by gate-efficient procedures that make $O(d)$ uses of $\mathcal{P}_v$, $\mathcal{P}_e$ and $\mathcal{P}_i$..

To implement $\mathrm{Ref}(A)$, we use the following method. Let Q_1 be a unitary operation that maps $|0^n\rangle|v\rangle$ to $|\psi_v\rangle|v\rangle$ for all $v \in V$, and let R_1 be the reflection about $\mathrm{span}\{|0^n\rangle|v\rangle : v \in V\}$, where $n = \Theta(\log(N))$. Then

$$\mathrm{Ref}(A) = Q_1 R_1 Q_1^{\dagger}. \tag{136}$$

Clearly, R_1 can be implemented in time $\mathrm{poly}(\log(N))$. We implement Q_1 using the following procedure. Given the state $|0^n\rangle|v\rangle$ for any $v \in V$, we first map it to

$$|0^n\rangle|v\rangle \left(\bigotimes_{e \in E(v)} |e\rangle|w_e\rangle \right) \tag{137}$$

by using $O(d)$ queries to $\mathcal{P}_v$ and $\mathcal{P}_e$ (recall that $|E(v)| \leq d$). Then we transform this state into

$$|\psi_v\rangle|v\rangle \left(\bigotimes_{e \in E(v)} |e\rangle|w_e\rangle \right), \tag{138}$$

where

$$|\psi_v\rangle = \frac{1}{\sqrt{\widetilde{\deg}(v) + \lambda}} \left(\sqrt{\lambda}|e_0\rangle + \sum_{e \in E(v)} \sqrt{w_e}|e\rangle \right). \tag{139}$$

Since $|\psi_v\rangle$ is a $(d+1)$-sparse vector in a $\text{poly}(N)$-dimensional space, this step can be accomplished by using $O(d \cdot \text{poly}(\log(N)))$ 2-qubit gates, as implied by Ref. [48]. Finally, we uncompute $\bigotimes_{e \in E(v)}|e\rangle|w_e\rangle$ by using $O(d)$ queries to $\mathcal{P}_v$ and $\mathcal{P}_e$. This implementation of Q_1 requires $O(d)$ uses of $\mathcal{P}_v$ and $\mathcal{P}_e$, and is gate-efficient. As a result, $\text{Ref}(A)$ can be implemented by a gate-efficient procedure that makes $O(d)$ uses of $\mathcal{P}_v$ and $\mathcal{P}_e$.

The implementation of $\text{Ref}(B)$ is similar. Let Q_2 be a unitary operation that maps $|e\rangle|0^m\rangle$ to $|e\rangle|\phi_e\rangle$ for all $e \in E'$, and let R_2 be the reflection about $\text{span}\{|e\rangle|0^m\rangle : e \in E'\}$, where $m = \Theta(\log(N))$. Then we have

$$\text{Ref}(B) = Q_2 R_2 Q_2^\dagger. \tag{140}$$

Clearly, R_2 can be implemented in time $\text{poly}(\log(N))$. We implement Q_2 using the following procedure. Given the state $|e\rangle|0^m\rangle$ for any $e \in E'$, if $e = e_0$, then we transform $|0^m\rangle$ into $|\phi_{e_0}\rangle = |\mathbf{i}_{ext}\rangle$ by calling $\mathcal{P}_i$ once; otherwise, we first map this state to

$$|e\rangle|0^m\rangle(|e^+\rangle|e^-\rangle|w_e\rangle) \tag{141}$$

by using $O(1)$ queries to $\mathcal{P}_e$, then transform it into

$$|e\rangle|\phi_e\rangle(|e^+\rangle|e^-\rangle|w_e\rangle), \tag{142}$$

where $|\phi_e\rangle = (|e^+\rangle - |e^-\rangle)/\sqrt{2}$, by using $\text{poly}(\log(N))$ 2-qubit gates, and finally uncompute $|e^+\rangle|e^-\rangle|w_e\rangle$ by using $O(1)$ queries to $\mathcal{P}_e$. This implementation of Q_2 requires $O(1)$ uses of $\mathcal{P}_e$ and $\mathcal{P}_i$, and is gate-efficient. As a consequence, $\text{Ref}(B)$ can be implemented by a gate-efficient procedure that makes $O(1)$ uses of $\mathcal{P}_e$ and $\mathcal{P}_i$. $\square$

Lemma 13 $U(A, B)$ *can be implemented with precision $\delta > 0$ by a gate-efficient procedure that makes*

$$O\left(\sqrt{\frac{c}{\lambda}} \cdot \text{poly}\left(\log\left(\frac{d}{\delta} \right) \right) \right)$$

uses of $\mathcal{P}_v$, $\mathcal{P}_e$ and $\mathcal{P}_i$.

Proof. Let us use the same notation as in the proof of Lemma 12. Recall that R_1, R_2 and Q_2 can be all implemented by gate-efficient procedures that make $O(1)$ uses of $\mathcal{P}_v$, $\mathcal{P}_e$ and $\mathcal{P}_i$. So we only need to show that Q_1 can be implemented with precision $\delta > 0$ by a gate-efficient procedure that makes $O(\sqrt{c/\lambda} \cdot \text{poly}(\log(d/\delta)))$ uses of $\mathcal{P}_v$ and $\mathcal{P}_e$.

Recall that Q_1 is the unitary operation mapping $|0^n\rangle|v\rangle$ to $|\psi_v\rangle|v\rangle$ for all $v \in V$, where $n = \Theta(\log(N))$ and

$$|\psi_v\rangle = \frac{1}{\sqrt{\widetilde{\deg}'(v)}} \sum_{e \in E'(v)} \sqrt{w_e}|e\rangle. \tag{143}$$

Given the state $|0^n\rangle|v\rangle$ for any $v \in V$, we first map it to

$$|0^n\rangle|v\rangle|d_v\rangle, \tag{144}$$

where $d_v := |E(v)| = \deg(v)$, by using $O(\log(d))$ queries to $\mathcal{P}_v$ and $\mathrm{poly}(\log(N))$ 2-qubit gates (via binary search). Then, we transform it into

$$\left(\frac{1}{\sqrt{d_v + 1}} \sum_{j=0}^{d_v} |j\rangle \right) |v\rangle |d_v\rangle \tag{145}$$

by using $\mathrm{poly}(\log(N))$ 2-qubit gates. Next, we convert this state into

$$\left(\frac{1}{\sqrt{d_v + 1}} \sum_{e \in E'(v)} |e\rangle \right) |v\rangle |d_v\rangle = \left[\frac{1}{\sqrt{d_v + 1}} \left(|e_0\rangle + \sum_{e \in E(v)} |e\rangle \right) \right] |v\rangle |d_v\rangle \tag{146}$$

by using $O(1)$ queries to $\mathcal{P}_v$ and $\mathrm{poly}(\log(N))$ 2-qubit gates. Then, we transform this state into

$$\left(\frac{1}{\sqrt{d_v + 1}} \sum_{e \in E'(v)} |e\rangle |w_e\rangle \right) |v\rangle |d_v\rangle = \left[\frac{1}{\sqrt{d_v + 1}} \left(|e_0\rangle |\lambda\rangle + \sum_{e \in E(v)} |e\rangle |w_e\rangle \right) \right] |v\rangle |d_v\rangle \tag{147}$$

by using $O(1)$ queries to $\mathcal{P}_e$ and $\mathrm{poly}(\log(N))$ 2-qubit gates. After that, we append an ancilla qubit in state $|0\rangle$, and perform the following controlled-rotation:

$$|w_e\rangle |0\rangle \rightarrow |w_e\rangle \left(\sqrt{\frac{w_e}{2c}} |0\rangle + \sqrt{1 - \frac{w_e}{2c}} |1\rangle \right). \tag{148}$$

This is a valid unitary operation, because $w_e \leq 2c$ for all $e \in E'(v)$ (note that $w_{e_0} = \lambda \leq 2 \leq 2c$). Then, we measure the ancilla qubit, and conditioning on the outcome being 0, we obtain the state

$$\left(\frac{\sum_{e \in E'(v)} \sqrt{w_e} |e\rangle |w_e\rangle}{\left\| \sum_{e \in E'(v)} \sqrt{w_e} |e\rangle |w_e\rangle \right\|} \right) |v\rangle |d_v\rangle. \tag{149}$$

The probability of this event happening is $\Omega(\lambda/c)$, since $w_e \geq \lambda/2$ for all $e \in E'(v)$ (note that $w_e \geq 1 \geq \lambda/2$ for all $e \in E$). Next, we uncompute $|d_v\rangle$ by using $O(\log(d))$ queries to $\mathcal{P}_v$ and $\mathrm{poly}(\log(N))$ 2-qubit gates. Finally, we uncompute $|w_e\rangle$ by using $O(1)$ queries to $\mathcal{P}_e$ and $\mathrm{poly}(\log(N))$ 2-qubit gates, and obtain the desired state $|\psi_v\rangle |v\rangle$.

The above procedure, denoted by $\mathcal{A}$, makes $O(\log(d))$ uses of $\mathcal{P}_v$ and $\mathcal{P}_e$, is gate-efficient, and has $\Omega(\lambda/c)$ success probability. We can raise the success probability to $\Omega(1)$ by using the standard amplitude amplification, which requires $O(\sqrt{c/\lambda})$ repetitions of $\mathcal{A}$. Let $\mathcal{A}'$ be this modified procedure with $\Omega(1)$ success probability. Then we can further boost the success probability to $1 - O(\delta')$ by using Grover's $\pi/3$ amplitude amplification (i.e. the generalization of fixed-point quantum search) [49], which requires $O(\log(1/\delta'))$ repetitions of $\mathcal{A}'$. Let us pick $\delta' = \Theta(\delta^2)$, and let $\mathcal{A}''$ be this procedure with $1 - O(\delta^2)$ success probability. Then $\mathcal{A}''$ makes $O(\sqrt{c/\lambda} \cdot \mathrm{poly}(\log(d/\delta)))$ uses of $\mathcal{P}_v$ and $\mathcal{P}_e$, is gate-efficient, and satisfies

$$\mathcal{A}'' |0^t\rangle |0^n\rangle |v\rangle = \sqrt{1 - \delta_v} |0^t\rangle |\psi_v\rangle |v\rangle + |\Phi_v^\perp\rangle, \tag{150}$$

where t is a positive integer, $\delta_v = O(\delta^2)$, $|\Phi_v^\perp\rangle$ is an unnormalized state satisfying $\langle \Phi_v^\perp | \Phi_v^\perp \rangle = \delta_v$ and $(|0^t\rangle \langle 0^t| \otimes I) |\Phi_v^\perp\rangle = 0$, for all $v \in V$. This implies that

$$\left\| \mathcal{A}'' |0^t\rangle |0^n\rangle |v\rangle - |0^t\rangle |\psi_v\rangle |v\rangle \right\|^2 = (1 - \sqrt{1 - \delta_v})^2 + \delta_v = O(\delta^2). \tag{151}$$

Meanwhile, since $\mathcal{A}''$ is a unitary operation, we have $\mathcal{A}''|0^t\rangle|0^n\rangle|u\rangle \perp \mathcal{A}''|0^t\rangle|0^n\rangle|v\rangle$ for any $u \neq v$. Then by Eq. (150), we know that $|0^t\rangle|\psi_u\rangle|u\rangle$, $|0^t\rangle|\psi_v\rangle|v\rangle$, $|\Phi_u^\perp\rangle$ and $|\Phi_v^\perp\rangle$ are mutually orthogonal for any $u \neq v$. As a result, for any normalized state $|z\rangle = \sum_{v \in V} z_v |v\rangle$, we have

$$\left\| \mathcal{A}''|0^t\rangle|0^n\rangle|z\rangle - |0^t\rangle(Q_1|0^n\rangle|z\rangle)) \right\|^2 \;=\; \left\| \sum_{v \in V} z_v \left(\mathcal{A}''|0^t\rangle|0^n\rangle|v\rangle - |0^t\rangle|\psi_v\rangle|v\rangle \right) \right\|^2 \quad (152)$$

$$\;=\; \sum_{v \in V} |z_v|^2 \left\| \mathcal{A}''|0^t\rangle|0^n\rangle|v\rangle - |0^t\rangle|\psi_v\rangle|v\rangle \right\|^2 \quad (153)$$

$$\;=\; O\left(\delta^2\right). \quad (154)$$

This means that

$$\left\| \langle 0^t | \mathcal{A}'' | 0^t \rangle - Q_1 \right\| = O(\delta), \quad (155)$$

as desired. □.

4.3   *Using quantum walks to analyze electrical networks*

Now we describe our quantum-walk-based algorithms for solving the ENA-P and ENA-ER problems. These algorithms require the following variant of phase estimation [50, 51], which determines whether the eigenphase corresponding to an eigenvector of a unitary operation is θ or far away from θ, for some given $\theta \in [0, 2\pi)$, succeeding with probability close to 1. (Similar procedures have been used in e.g. Refs. [52, 35].)

Lemma 14 *Let U be a unitary operation with eigenvectors $|\psi_j\rangle$ satisfying $U|\psi_j\rangle = e^{i\theta_j}|\psi_j\rangle$ for some $\theta_j \in [0, 2\pi)$. Let $\theta \in [0, 2\pi)$ and let $\Delta, \delta \in (0, 1)$. Then there is a unitary procedure $\mathcal{P}$ that requires $O((1/\Delta) \cdot \log(1/\delta))$ uses of U and $\mathrm{poly}(\log(1/(\Delta\delta)))$ additional 2-qubit gates, and satisfies*

$$\mathcal{P}|0\rangle\left|0^l\right\rangle|\psi_j\rangle = \left(\alpha_{j,0}|0\rangle|\eta_{j,0}\rangle + \alpha_{j,1}|1\rangle|\eta_{j,1}\rangle\right)|\psi_j\rangle, \quad (156)$$

where $l = O(\log(1/\Delta)\log(1/\delta))$, $|\alpha_{j,0}|^2 + |\alpha_{j,1}|^2 = 1$, $|\eta_{j,0}\rangle$ and $|\eta_{j,1}\rangle$ are two normalized states, and

- *If $\theta_j = \theta$, then $|\alpha_{j,0}|^2 \geq 1 - \delta$.*

- *If $|\theta_j - \theta| \geq \Delta$, then $|\alpha_{j,1}|^2 \geq 1 - \delta$.*

Proof. We can get a $\Delta/2$-additive approximation of θ_j by using the standard phase estimation, which requires $O(1/\Delta)$ uses of U and $\mathrm{poly}(\log(1/\Delta))$ additional 2-qubit gates. This is sufficient to distinguish between the two cases. However, it only succeeds with $\Omega(1)$ probability. To overcome this issue, we repeat this procedure $O(\log(1/\delta))$ times and check whether the median of the estimates is $\Delta/2$-close to θ. By a standard Chernoff bound, we can ensure that the failure proability is at most δ. Let $\mathcal{P}$ be this boosted procedure. Then $\mathcal{P}$ requires $O((1/\Delta) \cdot \log(1/\delta))$ uses of U and $\mathrm{poly}(\log(1/(\Delta\delta)))$ additional 2-qubit gates, and satisfies the desired properties. □.

Theorem 5 *The ENA-P problem can be solved by a gate-efficient quantum algorithm that makes*

$$O\left(\min\left\{ \frac{c^{0.5}d^{1.5}}{\epsilon\lambda}, \frac{cd^{0.5}}{\epsilon\lambda^{1.5}} \right\} \cdot \mathrm{poly}\left(\log\left(\frac{cd}{\epsilon\lambda} \right) \right) \right)$$

uses of $\mathcal{P}_v$, $\mathcal{P}_e$ and $\mathcal{P}_i$.

Proof. **Algorithm:** We estimate $\mathcal{E}(\mathbf{i})$ up to multiplicative error $O(\epsilon)$ by using the following algorithm:

- Let $\mathcal{P}$ be the unitary procedure in Lemma 14 for $U = U(A,B)$, $\theta = \pi$, $\Delta = \sqrt{\lambda/3}$ and $\delta = O(\epsilon\lambda/(cd))$. Suppose

$$\mathcal{P}|0\rangle_1 \big|0^l\big\rangle_2 (B|e_0\rangle)_3 = \mu_0|0\rangle_1|\varphi_0\rangle_{2,3} + \mu_1|1\rangle_1|\varphi_1\rangle_{2,3} \tag{157}$$

where $l = O(\log(1/\Delta)\log(1/\delta))$, $|\mu_0|^2 + |\mu_1|^2 = 1$, $|\varphi_0\rangle$ and $|\varphi_1\rangle$ are two normalized states. We use amplitude estimation to get an $O(\epsilon)$-multiplicative approximation $\hat{r}$ of $r := |\mu_1|^2$ (succeeding with probability at least $3/4$). Then we return

$$\hat{E} := \frac{\hat{r}}{1-\hat{r}} \cdot \frac{1}{2\lambda} \tag{158}$$

as our estimate of $\mathcal{E}(\mathbf{i})$. During this process, the unitary operation $U(A,B)$ is implemented either by the procedure in Lemma 12, or by the procedure in Lemma 13 with precision $O(\epsilon^2\lambda^2/(cd))$.

Correctness: Recall that $\mathcal{H}'$ is the -1 eigenspace of $U(A,B)$, and Π' is the projection onto this subspace. We have shown in the previous subsection that

$$\Pi' B|e_0\rangle \propto B|\Phi\rangle = aB|e_0\rangle + BW^{-1/2}|\mathbf{i}\rangle, \tag{159}$$

where $a = 1/\sqrt{2\lambda}$. Since B is an isometry, we have

$$\left\|BW^{-1/2}|\mathbf{i}\rangle\right\|^2 = \left\|W^{-1/2}|\mathbf{i}\rangle\right\|^2 = \mathcal{E}(\mathbf{i}) = \mathbf{i}_{ext}^T L_G^+ \mathbf{i}_{ext}. \tag{160}$$

Meanwhile, since $\lambda_2(\overline{L}_G) \geq \lambda$ and $1 \leq \widetilde{\deg}(v) \leq cd$ for all $v \in V$, by Eq. (12) and Lemma 1, we get

$$\lambda \leq \lambda_2(L_G) \leq \lambda_3(L_G) \leq \cdots \leq \lambda_N(L_G) \leq 2cd. \tag{161}$$

Then, since $\mathbf{i}_{ext} \in \text{Range}(L_G)$ and $\|\mathbf{i}_{ext}\| = 1$, we obtain

$$\frac{1}{2cd} \leq \mathcal{E}(\mathbf{i}) = \mathbf{i}_{ext}^T L_G^+ \mathbf{i}_{ext} \leq \frac{1}{\lambda} = 2a^2. \tag{162}$$

Now let

$$|\Psi\rangle := \frac{B|\Phi\rangle}{\|B|\Phi\rangle\|} = \frac{aB|e_0\rangle + BW^{-1/2}|\mathbf{i}\rangle}{\sqrt{a^2 + \mathcal{E}(\mathbf{i})}} \in \mathcal{H}', \tag{163}$$

and let $\{|\Psi_k^\perp\rangle : 1 \leq k \leq K\}$ be an orthonormal basis for $(\mathcal{H}')^\perp$. Then Eq. (159) implies

$$B|e_0\rangle = \beta|\Psi\rangle + \sum_{k=1}^K \beta_k\big|\Psi_k^\perp\big\rangle, \tag{164}$$

for some numbers $\beta, \beta_1, \beta_2, \ldots, \beta_K$. Since B is an isometry, we have

$$\beta = \langle \Psi | B | e_0 \rangle = \frac{a}{\sqrt{a^2 + \mathcal{E}(\mathbf{i})}}. \tag{165}$$

Let

$$r_1 := 1 - |\beta|^2 = \sum_{k=1}^{K} |\beta_k|^2 = \frac{\mathcal{E}(\mathbf{i})}{a^2 + \mathcal{E}(\mathbf{i})}. \tag{166}$$

Then we have

$$\mathcal{E}(\mathbf{i}) = \frac{r_1}{1 - r_1} \cdot a^2. \tag{167}$$

In addition, by Eqs. (162) and (166), we get

$$\frac{1}{\kappa + 1} \leq r_1 \leq \frac{2}{3}, \tag{168}$$

where $\kappa := cd/\lambda$.

Now, since $\Delta = \sqrt{\lambda/3}$ is smaller than the eigenphase gap around π of $U(A, B)$ by Lemma 11, $\mathcal{P}$ satisfies

$$\mathcal{P}|0\rangle \big| 0^l \big\rangle |\Psi\rangle = (\alpha_0 |0\rangle |\eta_0\rangle + \alpha_1 |1\rangle |\eta_1\rangle) |\Psi\rangle, \tag{169}$$

where $|\alpha_0|^2 \geq 1 - \delta$, $|\alpha_1|^2 \leq \delta$, $|\eta_0\rangle$ and $|\eta_1\rangle$ are normalized states, and

$$\mathcal{P}|0\rangle \big| 0^l \big\rangle \big| \Psi_k^{\perp} \big\rangle = (\alpha_{k,0} |0\rangle |\eta_{k,0}\rangle + \alpha_{k,1} |1\rangle |\eta_{k,1}\rangle) \big| \Psi_k^{\perp} \big\rangle, \tag{170}$$

where $|\alpha_{k,1}|^2 \geq 1 - \delta$, $|\alpha_{k,0}|^2 \leq \delta$, $|\eta_{k,0}\rangle$ and $|\eta_{k,1}\rangle$ are normalized states, for all k. As a result, we get

$$\mathcal{P}|0\rangle \big| 0^l \big\rangle (B|e_0\rangle) = |0\rangle \left(\alpha_0 \beta |\eta_0\rangle |\Psi\rangle + \sum_{k=1}^{K} \alpha_{k,0} \beta_k |\eta_{k,0}\rangle \big| \Psi_k^{\perp} \big\rangle \right) \tag{171}$$

$$+ |1\rangle \left(\alpha_1 \beta |\eta_1\rangle |\Psi\rangle + \sum_{k=1}^{K} \alpha_{k,1} \beta_k |\eta_{k,1}\rangle \big| \Psi_k^{\perp} \big\rangle \right). \tag{172}$$

This implies that

$$r = |\alpha_1 \beta|^2 + \sum_{k=1}^{K} |\alpha_{k,1} \beta_k|^2. \tag{173}$$

Note that

$$\left| r_1 - \sum_{k=1}^{K} |\alpha_{k,1} \beta_k|^2 \right| = \sum_{k=1}^{K} (1 - |\alpha_{k,1}|^2) |\beta_k|^2 \leq \delta r_1 = O(\epsilon r_1), \tag{174}$$

and

$$|\alpha_1 \beta|^2 \leq |\alpha_1|^2 \leq \delta = O(\epsilon r_1), \tag{175}$$

since $r_1 = \Omega(1/\kappa) = \Omega(\lambda/(cd))$ by Eq. (168). It follows that

$$|r - r_1| \leq \left| r_1 - \sum_{k=1}^{K} |\alpha_{k,1} \beta_k|^2 \right| + |\alpha_1 \beta|^2 = O(\epsilon r_1). \tag{176}$$

Namely, r is an $O(\epsilon)$-multiplicative approximation of r_1. Meanwhile, $\hat{r}$ is an $O(\epsilon)$-multiplicative approximation of r. Combining these two facts, we know that

$$|\hat{r} - r_1| \leq |\hat{r} - r| + |r - r_1| = O(\epsilon r) + O(\epsilon r_1) = O(\epsilon r_1), \tag{177}$$

and

$$|(1 - \hat{r}) - (1 - r_1)| = |\hat{r} - r_1| = O(\epsilon r_1) = O(\epsilon(1 - r_1)), \tag{178}$$

since $r_1 \leq 2/3$ by Eq. (168). This implies that

$$(1 - O(\epsilon)) \cdot \frac{r_1}{1 - r_1} \leq \frac{\hat{r}}{1 - \hat{r}} \leq (1 + O(\epsilon)) \cdot \frac{r_1}{1 - r_1}, \tag{179}$$

and hence

$$\left| \frac{\hat{r}}{1 - \hat{r}} - \frac{r_1}{1 - r_1} \right| = O\left(\epsilon \cdot \frac{r_1}{1 - r_1} \right). \tag{180}$$

As a consequence, by Eqs. (158) and (167), we get

$$|\hat{E} - \mathcal{E}(i)| = \left| \frac{\hat{r}}{1 - \hat{r}} \cdot a^2 - \frac{r_1}{1 - r_1} \cdot a^2 \right| = O\left(\epsilon \cdot \frac{r_1}{1 - r_1} \cdot a^2 \right) = O(\epsilon \cdot \mathcal{E}(\mathbf{i})), \tag{181}$$

as desired.

In the above argument, we have assumed that the unitary operation $U(A, B)$ is implemented perfectly. This is true if we use the procedure in Lemma 12 to implement $U(A, B)$. If we instead use the procedure in Lemma 13 to implement $U(A, B)$ with precision $O(\lambda^2 \epsilon^2 / (cd))$, then the algorithm still outputs a correct $\hat{E}$ with high probability. The reason is as follows. We will show below that this algorithm only makes $o(cd/(\lambda^2 \epsilon^2))$ uses of $U(A, B)$. Provided that each $U(A, B)$ is implemented with precision $O(\lambda^2 \epsilon^2 / (cd))$, the error in the final state (compared to the ideal case) is only $o(1)$. Therefore, the probability that this algorithm outputs a correct $\hat{r}$ (and hence a correct $\hat{E}$) is at least $3/4 - o(1)$.

Complexity: The state $B|e_0\rangle = |e_0\rangle |\mathbf{i}_{ext}\rangle$ can be prepared by making $O(1)$ uses of $\mathcal{P}_i$. Since $r = \Theta(r_1) = \Omega(1/\kappa)$ and we want to estimate it up to multiplicative error $O(\epsilon)$, amplitude estimation requires

$$O\left(\frac{1}{\epsilon \sqrt{r}} \right) = O\left(\frac{\sqrt{\kappa}}{\epsilon} \right) \tag{182}$$

repetitions of $\mathcal{P}$. By Lemma 14, the procedure $\mathcal{P}$ can be implemented with $O((1/\Delta) \cdot \log(1/\delta))$ uses of $U(A, B)$ and $\text{poly}(\log(1/(\Delta\delta)))$ additional 2-qubit gates. So this algorithm makes

$$O\left(\frac{\sqrt{\kappa}}{\epsilon} \cdot \frac{1}{\Delta} \cdot \log\left(\frac{1}{\delta} \right) \right) = O\left(\frac{\sqrt{cd}}{\epsilon\lambda} \cdot \log\left(\frac{cd}{\epsilon\lambda} \right) \right) \tag{183}$$

uses of $U(A, B)$. If we use the procedure in Lemma 12 to implement $U(A, B)$, the resulting algorithm will require

$$O\left(\frac{c^{0.5} d^{1.5}}{\epsilon\lambda} \cdot \text{poly}\left(\log\left(\frac{cd}{\epsilon\lambda} \right) \right) \right)$$

uses of $\mathcal{P}_v$, $\mathcal{P}_e$, $\mathcal{P}_i$, and is gate-efficient. Alternatively, if we use the procedure in Lemma 13 to implement $U(A,B)$ with precision $O(\lambda^2 \epsilon^2/(cd))$, the resulting algorithm will require

$$O\left(\frac{cd^{0.5}}{\epsilon \lambda^{1.5}} \cdot \mathrm{poly}\left(\log\left(\frac{cd}{\epsilon \lambda} \right) \right) \right)$$

uses of $\mathcal{P}_v$, $\mathcal{P}_e$, $\mathcal{P}_i$, and is also gate-efficient. Our claim follows from the combination of these two facts. □.

Corollary 3 *The ENA-ER problem can be solved by a gate-efficient quantum algorithm that makes*

$$O\left(\min\left\{ \frac{c^{0.5}d^{1.5}}{\epsilon \lambda}, \frac{cd^{0.5}}{\epsilon \lambda^{1.5}} \right\} \cdot \mathrm{poly}\left(\log\left(\frac{cd}{\epsilon \lambda} \right) \right) \right)$$

uses of $\mathcal{P}_v$ and $\mathcal{P}_e$.

Proof. Recall that $\mathrm{R}_{\mathrm{eff}}(s,t) = \mathcal{E}(\mathbf{i})$, where $\mathbf{i} = W_G B_G^T L_G^+ \chi_{s,t}$ is the electrical flow induced by the external current $\chi_{s,t} = |s\rangle - |t\rangle$. Clearly, we can prepare the state $(|s\rangle - |t\rangle)/\sqrt{2}$ in time $\mathrm{poly}(\log(N))$. Then we can run the algorithm in Theorem 5 to obtain an $O(\epsilon)$-multiplicative approximation of $\mathcal{E}(\tilde{\mathbf{i}}) = \mathcal{E}(\mathbf{i})/2$, where $\tilde{\mathbf{i}} = \mathbf{i}/\sqrt{2}$ is the electrical flow induced by the external current $\chi_{s,t}/\sqrt{2}$. Then we multiply this result by a factor of 2, and obtain an $O(\epsilon)$-multiplicative approximation of $\mathcal{E}(\mathbf{i}) = \mathrm{R}_{\mathrm{eff}}(s,t)$. By Theorem 5, this algorithm makes

$$O\left(\min\left\{ \frac{c^{0.5}d^{1.5}}{\epsilon \lambda}, \frac{cd^{0.5}}{\epsilon \lambda^{1.5}} \right\} \cdot \mathrm{poly}\left(\log\left(\frac{cd}{\epsilon \lambda} \right) \right) \right)$$

uses of $\mathcal{P}_v$ and $\mathcal{P}_e$, and is gate-efficient. □.

One can compare the algorithm in Theorem 5 (or Corollary 3) with the one in Theorem 4 (or Corollary 2). The quantum-walk-based one is unconditionally better if we use the procedure in Lemma 12 to implement $U(A,B)$. If we instead use the procedure in Lemma 13 to implement $U(A,B)$, then the quantum-walk-based one has much better dependence on d, but slightly worse dependence on $1/\lambda$. So it is more suitable in the case where d is larger than $1/\lambda$ (which is possible and common).

We remark that the algorithm in Theorem 5 can be modified to solve the ENA-C problem (and the ENA-V problem under the promise that s and t are adjacent vertices). Specifically, recall that

$$|\Psi\rangle = \frac{B|\Phi\rangle}{\|B|\Phi\rangle\|} = \frac{1}{\sqrt{a^2 + \mathcal{E}(i)}}\left(aB|e_0\rangle + \sum_{e \in E} \frac{\mathbf{i}(e)}{\sqrt{w_e}} B|e\rangle \right). \tag{184}$$

If we can create this state, then we can infer $|\mathbf{i}(e)|$ from it, for any given $e \in E$. To prepare the state $|\Psi\rangle$, we need to use a clean version of the procedure $\mathcal{P}$ in Lemma 14. That is, we need to replace the states $|\eta_{j,0}\rangle$ and $|\eta_{j,1}\rangle$ in Lemma 14 with $|0^l\rangle$ (namely, we want to reset the l ancilla qubits to their initial states after the computation). This can be approximately achieved by using the standard "do-copy-undo" trick. Then, when we apply this clean version of $\mathcal{P}$ on $|0\rangle|0^l\rangle B|e_0\rangle$ (for some positive integer t) and measure the first qubit, conditioning on the outcome being 0, we would obtain a state close to $|\Psi\rangle$, from which $|\mathbf{i}(e)|$ can be learned. However, this algorithm for solving ENA-C is not more efficient than the one in Theorem 3. So we will not present it in detail here.

5 Lower Bounds on the Complexity of Electrical Network Analysis

So far we have presented two classes of quantum algorithms for analyzing electrical networks. All of these algorithms have complexities polynomial in $1/\lambda$ (and other parameters), where λ is the spectral gap of the normalized Laplacian of the network. In this section, we show that this polynomial dependence on $1/\lambda$ is necessary. Specifically, we prove that in order to solve any of the ENA-V, ENA-C, ENA-P, ENA-ER problems, one has to make $\Omega\left(1/\sqrt{\lambda}\right)$ queries to the graph. This lower bound implies that our algorithms are optimal up to polynomial factors [f] and hence cannot be greatly improved.

Theorem 6 *For any positive integer N, there exists an unweighted connected graph $G = (V, E)$ with four distinguished vertices $s, t, u, v \in V$ such that $|V| = 10N$, $\deg(G) = 3$, $(u, v) \in E$, $\lambda_2(\overline{L}_G) = \Omega(1/N^2)$, $\lambda_2(\overline{L}_G) = O(1/N)$, and assuming a unit electric current is injected at s and extracted at t, one needs to make $\Omega(N)$ queries to G to solve any of the following problems (succeeding with probability at least $2/3$):*

1. *Estimate the voltage between u and v up to additive error 0.1.*

2. *Estimate the current on (u, v) up to additive error 0.1.*

3. *Estimate the power dissipated by G up to multiplicative error 0.1.*

4. *Estimate the effective resistance between s and t up to multiplicative error 0.1.*

Proof. We will build a graph such that, if one solves any of the above problems on this graph, then one has solved a corresponding PARITY problem. Recall that in the PARITY problem, one is given oracle access to an N-bit string $x = x_1 x_2 \ldots x_N$, and needs to determine the value of $\text{PARITY}(x) := x_1 \oplus x_2 \oplus \ldots \oplus x_N$. Our claims will follow from a known lower bound on the quantum query complexity of PARITY.

Now let us make this argument precise. Given an N-bit string $x = x_1 x_2 \ldots x_N$, we will map it to an unweighted graph $G(x) = (V(x), E(x))$ with $10N$ vertices. For convenience, we will label the vertices in this graph by (i, j) or (i^*, j) for some integers i and j. We start with $10N$ isolated vertices, which are labeled by (i, a) for $i \in \{1, 2, \ldots, N + 1\}$ and $a \in \{0, 1\}$, and (j^*, b) for $j \in \{1, 2, \ldots, 4N - 1\}$ and $b \in \{0, 1\}$. Then we add the following edges to this graph:

- For $i \in \{1, 2, \ldots, N\}$ and $a \in \{0, 1\}$, we add an edge between (i, a) and $(i + 1, a \oplus x_i)$. That is, if $x_i = 0$, we add an edge between $(i, 0)$ and $(i + 1, 0)$, and an edge between $(i, 1)$ and $(i + 1, 1)$; otherwise, we add an edge between $(i, 0)$ and $(i + 1, 1)$, and an edge between $(i, 1)$ and $(i + 1, 0)$).

- For $j \in \{1, 2, \ldots, 4N - 2\}$ and $b \in \{0, 1\}$, we add an edge between (j^*, b) and $((j+1)^*, b)$.

- For $b \in \{0, 1\}$, we add an edge between $(1, 0)$ and $(1^*, b)$.

- For $b \in \{0, 1\}$, we add an edge between $((4N - 1)^*, b)$ and $(N + 1, b)$.

---

[f] The polynomial dependence on the other parameters, including c, d, $\log(N)$ and $1/\epsilon$, is clearly necessary.

For example, Fig.2 shows the graph $G(x)$ for the string $x = 11010$ (where $N = 5$). Note that $G(x)$ consists of N crossing-type or parallel-type gadgets (where the i-th gadget's type depends the value of x_i) and two long paths, one connecting $(1,0)$ and $(N+1,0)$ and the other connecting $(1,0)$ and $(N+1,1)$. (Similar constructions have been used to prove lower bounds on the quantum query complexity of Hamiltonian simulation [4, 53].)
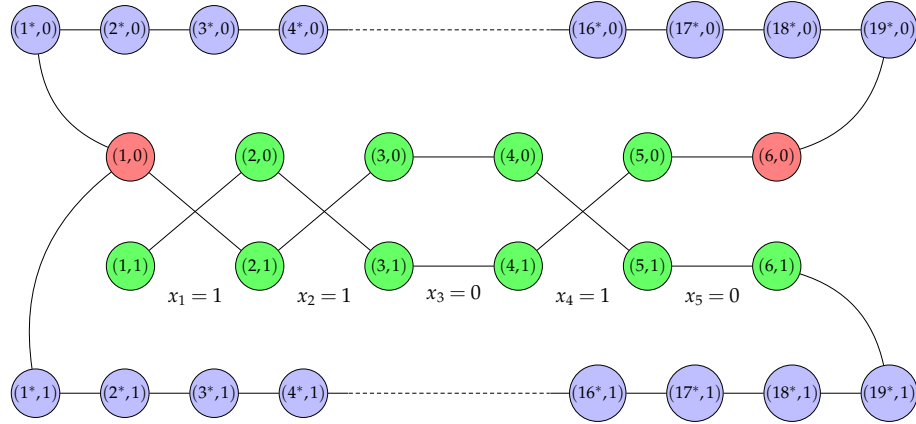

Fig. 2. The graph $G(x)$ for the string $x = 11010$.

Now we pick $s = (1,0)$, $t = (N+1,0)$, $u = ((2N-1)^*,0)$ and $v = ((2N)^*,0)$. Then the graph $G(x)$ satisfies the following property:

- If PARITY$(x) = 0$, then there are two paths between s and t in $G(x)$. One of them is

$$s = (1,0) \to (2,x_1) \to (3,x_1 \oplus x_2) \to \cdots \to (N+1, \oplus_{i=1}^N x_i) = (N+1,0) = t, \quad (185)$$

and the other is

$$s = (1,0) \to (1^*,0) \to (2^*,0) \to \cdots \to ((4N-1)^*,0) \to (N+1,0) = t. \quad (186)$$

- If PARITY$(x) = 1$, then there is only one path between s and t in $G(x)$, which is described by Eq. (186).

This implies that when a unit electric current is injected at s and extracted at t, we have:

- If PARITY$(x) = 0$, then there is an electrical flow of value 0.8 on the path described by Eq. (185), a flow of value 0.2 on the path described by Eq. (186), and no flow on other edges. Thus, the current on (u,v) is 0.2, and so is the voltage between u and v. Moreover, the power dissipated by $G(x)$ is $0.8^2 \times N + 0.2^2 \times 4N = 0.8N$, and so is the effective resistance between s and t.

- If PARITY$(x) = 1$, then there is an electrical flow of value 1 on the path described by Eq. (186), and no flow on other edges. Thus, the current on (u,v) is 1, and so is the voltage between u and v. Moreover, the power dissipated by $G(x)$ is $4N$, and so is the effective resistance between s and t.

It follows that we can distinguish these two cases by solving any of the following problems:

1. Estimate the voltage between u and v up to additive error 0.1.

2. Estimate the current on (u,v) up to additive error 0.1.

3. Estimate the power dissipated by G up to multiplicative error 0.1.

4. Estimate the effective resistance between s and t up to multiplicative error 0.1.

It is known that PARITY has $\Theta(N)$ bounded-error quantum query complexity [54, 55]. This implies that one needs to make $\Omega(N)$ queries to $G(x)$ to solve any of the above problems.

Finally, we show that $G(x)$ satisfies the other desired properties. Clearly, $G(x)$ is a connected graph with maximum degree 3. Moreover, it has conductance $\phi_{G(x)} = \Theta(1/N)$. To see this, consider the cut $(S, \bar{S})$, where $S = \{(i,a) : i \in \{1,2,\ldots,\lfloor N/2 \rfloor\}, a \in \{0,1\}\} \cup \{(j^*,b) : j \in \{1,2,\ldots,2N\}, b \in \{0,1\}\}$ and $\bar{S} = V \setminus S$. We have $\text{vol}(S)$, $\text{vol}(\bar{S}) = \Theta(N)$ and $|E(S,\bar{S})| = O(1)$. So $\phi_S = O(1/N)$, which implies $\phi_{G(x)} = O(1/N)$. On the other hand, since $G(x)$ is a connected graph with $O(N)$ edges, for any cut $(S, \bar{S})$, we have $\text{vol}(S)$, $\text{vol}(\bar{S}) = O(N)$, $|E(S,\bar{S})| = \Omega(1)$, and hence $\phi_S = \Omega(1/N)$. This implies $\phi_{G(x)} = \Omega(1/N)$. Combining these two facts, we obtain $\phi_{G(x)} = \Theta(1/N)$. Then by Cheeger's inequality (i.e. Eq. (19)), we have $\lambda_2(\bar{L}_{G(x)}) = \Omega(1/N^2)$ and $\lambda_2(\bar{L}_{G(x)}) = O(1/N)$. This concludes the proof. □.

6   Discussion

To summarize, we have proposed two classes of quantum algorithms for analyzing large sparse electrical networks. The first class is based on solving linear systems, and the second class is based on using quantum walks. These algorithms compute various electrical quantities, including voltages, currents, dissipated powers and effective resistances, in time $\text{poly}(d,c,\log(N),1/\lambda,1/\epsilon)$, where N is the number of vertices in the graph, d is the maximum unweighted degree of the vertices, c is the ratio of largest to smallest edge resistance, λ is the spectral gap of the normalized Laplacian of the graph, and ϵ is the accuracy. Furthermore, we prove that the polynomial dependence on $1/\lambda$ is necessary. Hence, our algorithms are optimal up to polynomial factors and cannot be significantly improved.

We have seen that a Laplacian system $Lx = b$ naturally arises when one wants to compute the voltages in an electrical network. Such systems also play an important role in other graph problems, such as graph partitioning (e.g. [56, 57, 58, 15]), graph sparsification (e.g. [11, 12, 14]) and maximum flows (e.g. [13, 16, 17]). As a result, much effort has been dedicated to studying the classical complexity of solving these systems (e.g. [33, 34]). It appears that Laplacian systems are easier to solve than general linear systems classically. In contrast, we do not know whether the quantum analogue of this statement is true. In particular, Harrow, Hassidim and Lloyd [36] showed that it is BQP-complete to solve a general sparse well-conditioned linear system $Ax = b$ (in certain sense). Does this theorem still hold under the restriction that A is a Laplacian? If so, there would be an interesting implication: Any problem in BQP can be reduced to the problem of computing certain voltages in an exponentially-large electrical network! This can be even viewed as a novel (but impractical) proposal for building a quantum computer! On the other hand, if Laplacian systems are

indeed easier to solve than general linear systems quantumly, then what is the exact quantum complexity of solving them? In particular, can they be solved in time sublinear in the Laplacian's finite condition number? These are left as interesting open questions.

In this paper, we have focused on direct-current (DC) circuits which consist of resistors and DC sources. It is also worth exploring alternating-current (AC) circuits which consist of resistors, capacitors, inductors and AC sources. Such electrical systems are governed by a set of second-order linear ordinary differential equations. But it is possible to transform these differential equations into a system of linear equations by applying the Fourier or Laplace transform. The resulting linear system can be viewed as a *complex* Laplacian system, and it can be solved by invoking a quantum linear system algorithm. However, the complexity of this algorithm is difficult to analyze, because it depends on the condition number of a complex Laplacian, and there are few known methods to bound this quantity (to our knowledge, there is no analogue of Cheeger's inequality for complex Laplacians). So it is unclear how much quantum advantage can be gained on such systems.

Spectral graph theory has become a powerful tool for the design and analysis of fast classical algorithms for various graph problems, such as graph partitioning (e.g. [56, 57, 58, 15]), maximum flows (e.g. [13, 16, 17]) and max cuts (e.g. [59]). Its application in quantum computation, nevertheless, is still scarce. It would be exciting to see more quantum algorithms (especially exponentially faster ones) developed based on this elegant theory.

Acknowledgments

References

1. Seth Lloyd. Universal quantum simulators. *Science*, 273(5278):1073–1078, 1996.
2. Daniel S. Abrams and Seth Lloyd. Simulation of many-body fermi systems on a universal quantum computer. *Phys. Rev. Lett.*, 79:2586–2589, Sep 1997.
3. Dorit Aharonov and Amnon Ta-Shma. Adiabatic quantum state generation and statistical zero knowledge. In *Proceedings of the Thirty-fifth Annual ACM Symposium on Theory of Computing*, STOC '03, pages 20–29, New York, NY, USA, 2003. ACM.
4. Dominic W. Berry, Graeme Ahokas, Richard Cleve, and Barry C. Sanders. Efficient quantum algorithms for simulating sparse hamiltonians. *Communications in Mathematical Physics*, 270(2):359–371, Mar 2007.
5. Stephen P. Jordan, Keith S. M. Lee, and John Preskill. Quantum algorithms for quantum field theories. *Science*, 336(6085):1130–1133, 2012.
6. Peter W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, SFCS '94, pages 124–134, Washington, DC, USA, 1994. IEEE Computer Society.
7. Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.*, 26(5):1484–1509, October 1997.
8. Sean Hallgren. Fast quantum algorithms for computing the unit group and class group of a num-

ber field. In *Proceedings of the Thirty-seventh Annual ACM Symposium on Theory of Computing*, STOC '05, pages 468–474, New York, NY, USA, 2005. ACM.

9. Sean Hallgren. Polynomial-time quantum algorithms for pell's equation and the principal ideal problem. *J. ACM*, 54(1):4:1–4:19, March 2007.

10. Andrew M. Childs and Wim van Dam. Quantum algorithms for algebraic problems. *Rev. Mod. Phys.*, 82:1–52, Jan 2010.

11. Daniel A. Spielman and Nikhil Srivastava. Graph sparsification by effective resistances. *SIAM J. Comput.*, 40(6):1913–1926, December 2011.

12. Daniel A. Spielman and Shang-Hua Teng. Spectral sparsification of graphs. *SIAM Journal on Computing*, 40(4):981–1025, 2011.

13. Paul Christiano, Jonathan A. Kelner, Aleksander Madry, Daniel A. Spielman, and Shang-Hua Teng. Electrical flows, laplacian systems, and faster approximation of maximum flow in undirected graphs. In *Proceedings of the Forty-third Annual ACM Symposium on Theory of Computing*, STOC '11, pages 273–282, New York, NY, USA, 2011. ACM.

14. Ioannis Koutis, Alex Levin, and Richard Peng. Faster spectral sparsification and numerical algorithms for sdd matrices. *ACM Trans. Algorithms*, 12(2):17:1–17:16, December 2015.

15. Nisheeth K. Vishnoi. Lx=b. *Foundations and Trends in Theoretical Computer Science*, 8(1–2):1–141, 2012.

16. Yin Tat Lee, Satish Rao, and Nikhil Srivastava. A new approach to computing maximum flows using electrical flows. In *Proceedings of the Forty-fifth Annual ACM Symposium on Theory of Computing*, STOC '13, pages 755–764, New York, NY, USA, 2013. ACM.

17. Aleksander Madry. Navigating central path with electrical flows: From flows to matchings, and back. In *Proceedings of the 2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, FOCS '13, pages 253–262, Washington, DC, USA, 2013. IEEE Computer Society.

18. Aleksander Madry, Damian Straszak, and Jakub Tarnawski. Fast generation of random spanning trees and the effective resistance metric. In *Proceedings of the Twenty-sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '15, pages 2019–2036, Philadelphia, PA, USA, 2015. Society for Industrial and Applied Mathematics.

19. Peter G. Doyle and J. Laurie Snell. *Random walks and electric networks*. Mathematical Association of America,, 1984.

20. Ashok K Chandra, Prabhakar Raghavan, Walter L Ruzzo, Roman Smolensky, and Prasoon Tiwari. The electrical resistance of a graph captures its commute and cover times. *Computational Complexity*, 6(4):312–340, 1996.

21. Aleksandrs Belovs. Span programs for functions with constant-sized 1-certificates: Extended abstract. In *Proceedings of the Forty-fourth Annual ACM Symposium on Theory of Computing*, STOC '12, pages 77–84, New York, NY, USA, 2012. ACM.

22. Frédéric Magniez Troy Lee and Miklos Santha. Learning graph based quantum query algorithms for finding constant-size subgraphs. *Chicago Journal of Theoretical Computer Science*, 2012(10), December 2012.

23. Aleksandrs Belovs. Learning-graph-based quantum algorithm for k-distinctness. In *Proceedings of the 2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, FOCS '12, pages 207–216, Washington, DC, USA, 2012. IEEE Computer Society.

24. Aleksandrs Belovs and Ansis Rosmanis. On the power of non-adaptive learning graphs. *Comput. Complex.*, 23(2):323–354, June 2014.

25. Aleksandrs Belovs and Ben W. Reichardt. Span programs and quantum algorithms for st-connectivity and claw detection. In *Proceedings of the 20th Annual European Symposium on Algorithms*, ESA'12, pages 193–204, Berlin, Heidelberg, 2012. Springer-Verlag.

26. Aleksandrs Belovs, Andrew M. Childs, Stacey Jeffery, Robin Kothari, and Frédéric Magniez. Time-efficient quantum walks for 3-distinctness. In *Proceedings of the 40th International Conference on Automata, Languages, and Programming - Volume Part I*, ICALP'13, pages 105–122, Berlin, Heidelberg, 2013. Springer-Verlag.

27. Tsuyoshi Ito and Stacey Jeffery. Approximate span programs. In *43rd International Colloquium*

on Automata, Languages, and Programming (ICALP 2016)*, volume 55 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 12:1–12:14, Dagstuhl, Germany, 2016. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

28. Stacey Jeffery and Shelby Kimmel. Nand-trees, average choice complexity, and effective resistance. *arXiv preprint arXiv:1511.02235*, 2015.

29. Titouan Carette, Mathieu Laurière, and Frédéric Magniez. Extended learning graphs for triangle finding. *arXiv preprint arXiv:1609.07786*, 2016.

30. Aleksandrs Belovs. Quantum walks and electric networks. *arXiv preprint arXiv:1302.3143*, 2013.

31. Ashley Montanaro. Quantum walk speedup of backtracking algorithms. *arXiv preprint arXiv:1509.02374*, 2015.

32. Andris Ambainis, Krišjānis Prūsis, Jevgēnijs Vihrovs, and Thomas G. Wong. Oscillatory localization of quantum walks analyzed by classical electric circuits. *Phys. Rev. A*, 94:062324, Dec 2016.

33. Daniel A. Spielman and Shang-Hua Teng. Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems. In *Proceedings of the Thirty-sixth Annual ACM Symposium on Theory of Computing*, STOC '04, pages 81–90, New York, NY, USA, 2004. ACM.

34. Daniel A. Spielman and Shang-Hua Teng. Nearly linear time algorithms for preconditioning and solving symmetric, diagonally dominant linear systems. *SIAM Journal on Matrix Analysis and Applications*, 35(3):835–885, 2014.

35. Andrew M Childs, Robin Kothari, and Rolando D Somma. Quantum linear systems algorithm with exponentially improved dependence on precision. *arXiv preprint arXiv:1511.02306*, 2015.

36. Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Phys. Rev. Lett.*, 103:150502, Oct 2009.

37. Andris Ambainis. Variable time amplitude amplification and quantum algorithms for linear algebra problems. In *29th International Symposium on Theoretical Aspects of Computer Science (STACS 2012)*, volume 14 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 636–647, Dagstuhl, Germany, 2012. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

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

39. Mario Szegedy. Quantum speed-up of markov chain based algorithms. In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '04, pages 32–41, Washington, DC, USA, 2004. IEEE Computer Society.

40. Frdric Magniez, Miklos Santha, and Mario Szegedy. Quantum algorithms for the triangle problem. *SIAM Journal on Computing*, 37(2):413–424, 2007.

41. Frdric Magniez, Ashwin Nayak, Jrmie Roland, and Miklos Santha. Search via quantum walk. *SIAM Journal on Computing*, 40(1):142–164, 2011.

42. Ben W. Reichardt. Span programs and quantum query complexity: The general adversary bound is nearly tight for every boolean function. In *Proceedings of the 2009 50th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '09, pages 544–551, Washington, DC, USA, 2009. IEEE Computer Society.

43. Ben W. Reichardt. Reflections for quantum query algorithms. In *Proceedings of the Twenty-second Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '11, pages 560–569, Philadelphia, PA, USA, 2011. Society for Industrial and Applied Mathematics.

44. Jeff Cheeger. A lower bound for the smallest eigenvalue of the laplacian. *Problems in analysis*, pages 195–199, 1970.

45. Fan R. K. Chung. *Spectral graph theory*. Number 92. American Mathematical Soc., 1997.

46. Ashley Montanaro and Sam Pallister. Quantum algorithms and the finite element method. *Phys. Rev. A*, 93:032324, Mar 2016.

47. Gilles Brassard, Peter Hoyer, Michele Mosca, and Alain Tapp. Quantum amplitude amplification and estimation. *Contemporary Mathematics*, 305:53–74, 2002.

48. Vivek V. Shende, Stephen S. Bullock, and Igor L. Markov. Synthesis of quantum-logic circuits. *Trans. Comp.-Aided Des. Integ. Cir. Sys.*, 25(6):1000–1010, June 2006.

49. Lov K. Grover. Fixed-point quantum search. *Phys. Rev. Lett.*, 95:150501, Oct 2005.

50. Alexei Y. Kitaev. Quantum measurements and the abelian stabilizer problem. *arXiv preprint quant-ph/9511026*, 1995.
51. Richard Cleve, Artur Ekert, Chiara Macchiavello, and Michele Mosca. Quantum algorithms revisited. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 454(1969):339–354, 1998.
52. Daniel Nagaj, Pawel Wocjan, and Yong Zhang. Fast amplification of qma. *Quantum Info. Comput.*, 9(11):1053–1068, November 2009.
53. Dominic W. Berry, Andrew M. Childs, Richard Cleve, Robin Kothari, and Rolando D. Somma. Exponential improvement in precision for simulating sparse hamiltonians. In *Proceedings of the Forty-sixth Annual ACM Symposium on Theory of Computing*, STOC '14, pages 283–292, New York, NY, USA, 2014. ACM.
54. Robert Beals, Harry Buhrman, Richard Cleve, Michele Mosca, and Ronald de Wolf. Quantum lower bounds by polynomials. *J. ACM*, 48(4):778–797, July 2001.
55. Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. Limit on the speed of quantum computation in determining parity. *Phys. Rev. Lett.*, 81:5442–5444, Dec 1998.
56. Subhransu Maji, Nisheeth K. Vishnoi, and Jitendra Malik. Biased normalized cuts. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '11, pages 2057–2064, Washington, DC, USA, 2011. IEEE Computer Society.
57. James R. Lee, Shayan Oveis Gharan, and Luca Trevisan. Multiway spectral partitioning and higher-order cheeger inequalities. *J. ACM*, 61(6):37:1–37:30, December 2014.
58. Michael W. Mahoney, Lorenzo Orecchia, and Nisheeth K. Vishnoi. A local spectral method for graphs: With applications to improving graph partitions and exploring data graphs locally. *J. Mach. Learn. Res.*, 13(1):2339–2365, August 2012.
59. Luca Trevisan. Max cut and the smallest eigenvalue. *SIAM Journal on Computing*, 41(6):1769–1786, 2012.

Appendix A

The following lemma is used in Section 3.1. It says that if two unnormalized states are close and one of them has a large norm, then their normalized versions are also close.

Lemma A.1 *Let $|\psi\rangle$ and $|\phi\rangle$ be two unnormalized states satisfying $\||\psi\rangle\| \geq \alpha > 0$ and $\||\psi\rangle - |\phi\rangle\| \leq \beta$. Then*

$$\left\| \frac{|\psi\rangle}{\||\psi\rangle\|} - \frac{|\phi\rangle}{\||\phi\rangle\|} \right\| \leq \frac{2\beta}{\alpha}. \tag{A.1}$$

Proof. Using the triangle inequality, we get

$$
\left\| \frac{|\psi\rangle}{\||\psi\rangle\|} - \frac{|\phi\rangle}{\||\phi\rangle\|} \right\| = \left\| \frac{|\psi\rangle}{\||\psi\rangle\|} - \frac{|\phi\rangle}{\||\psi\rangle\|} + \frac{|\phi\rangle}{\||\psi\rangle\|} - \frac{|\phi\rangle}{\||\phi\rangle\|} \right\| \tag{A.2}
$$

$$
\leq \left\| \frac{|\psi\rangle}{\||\psi\rangle\|} - \frac{|\phi\rangle}{\||\psi\rangle\|} \right\| + \left\| \frac{|\phi\rangle}{\||\psi\rangle\|} - \frac{|\phi\rangle}{\||\phi\rangle\|} \right\| \tag{A.3}
$$

$$
\leq \frac{\||\psi\rangle - |\phi\rangle\|}{\||\psi\rangle\|} + \||\phi\rangle\| \left| \frac{1}{\||\psi\rangle\|} - \frac{1}{\||\phi\rangle\|} \right| \tag{A.4}
$$

$$
= \frac{\||\psi\rangle - |\phi\rangle\|}{\||\psi\rangle\|} + \frac{\left| \||\psi\rangle\| - \||\phi\rangle\| \right|}{\||\psi\rangle\|} \tag{A.5}
$$

$$
\leq \frac{2\||\psi\rangle - |\phi\rangle\|}{\||\psi\rangle\|} \tag{A.6}
$$

$$
= \frac{2\beta}{\alpha}. \tag{A.7}
$$

□.