Effective Resistances in Non-Expander Graphs

Dongrun Cai
School of Computer Science and Technology, University of Science and Technology of China, Hefei, China

Xue Chen
School of Computer Science and Technology, University of Science and Technology of China, Hefei, China

Pan Peng
School of Computer Science and Technology, University of Science and Technology of China, Hefei, China

Abstract

Effective resistances are ubiquitous in graph algorithms and network analysis. For an undirected graph $G$, its effective resistance $R_G(s,t)$ between two vertices $s$ and $t$ is defined as the equivalent resistance between $s$ and $t$ if $G$ is thought of as an electrical network with unit resistance on each edge. If we use $L_G$ to denote the Laplacian matrix of $G$ and $L_G^\dagger$ to denote its pseudo-inverse, we have $R_G(s,t) = (1 - 1_s)^\top L_G^\dagger (1 - 1_t)$ such that classical Laplacian solvers [46] provide almost-linear time algorithms to approximate $R_G(s,t)$.

In this work, we study sublinear time algorithms to approximate the effective resistance of an adjacent pair $s$ and $t$. We consider the classical adjacency list model [41] for local algorithms. While recent works [4, 40, 31] have provided sublinear time algorithms for expander graphs, we prove several lower bounds for general graphs of $n$ vertices and $m$ edges:

1. It needs $\Omega(n)$ queries to obtain $1.01$-approximations of the effective resistance of an adjacent pair $s$ and $t$, even for graphs of degree at most 3 except $s$ and $t$.
2. For graphs of degree at most $d$ and any parameter $\ell$, it needs $\Omega(m/\ell)$ queries to obtain $c \cdot \min\{d, \ell\}$-approximations where $c > 0$ is a universal constant.

Moreover, we supplement the first lower bound by providing a sublinear time $(1 + \epsilon)$-approximation algorithm for graphs of degree 2 except the pair $s$ and $t$.

One of our technical ingredients is to bound the expansion of a graph in terms of the smallest non-trivial eigenvalue of its Laplacian matrix after removing edges. We discover a new lower bound on the eigenvalues of perturbed graphs (resp. perturbed matrices) by incorporating the effective resistance of the removed edge (resp. the leverage scores of the removed rows), which may be of independent interest.

2012 ACM Subject Classification Theory of computation → Streaming, sublinear and near linear time algorithms

Keywords and phrases Sublinear Time Algorithm, Effective Resistance, Leverage Scores, Matrix Perturbation

Digital Object Identifier 10.4230/LIPIcs.ESA.2023.29


Funding Pan Peng: Supported in part by NSFC grant 62272431 and “the Fundamental Research Funds for the Central Universities”. 

Acknowledgements We thank Jingcheng Liu (Nanjing University) for suggesting the proof of Lemma 5 using the characteristic polynomial method. Also, we thank all anonymous reviewers for the helpful comments.

© Dongrun Cai, Xue Chen, and Pan Peng; licensed under Creative Commons License CC-BY 4.0
1 Introduction

Effective resistance of a graph is a fundamental quantity to measure the similarity between two vertices. Given an unweighted graph \( G \) and two vertices \( s \) and \( t \), the \( s-t \) effective resistance, denoted by \( R_{G}(s,t) \), is defined as the electrical distance between \( s \) and \( t \) when \( G \) represents an electrical circuit with each edge \( e \) a resistor with electrical resistance 1. Together with the related concept electrical flows, effective resistances have played important roles in advancing the development of graph algorithms. They have been utilized for computing and approximating maximum flow [10, 34], generating random spanning tree [35, 43], designing faster algorithms for multicommodity flow [28], and graph sparsification [45, 15]. In addition, effective resistances have also found applications in machine learning and social network analysis. For example, it has been used for graph convolutional networks [2], for measuring the similarity of vertices in social networks [30, 40] and measuring robustness of networks [19].

Let \( L_G \) denote the Laplacian matrix of \( G \) and \( L_G^{-\dagger} \) denote its pseudo-inverse. Then the \( s-t \) effective resistance admits an elegant expression \( R_{G}(s,t) = (\mathbf{1}_s - \mathbf{1}_t)^\top \cdot L_G^{-\dagger} \cdot (\mathbf{1}_s - \mathbf{1}_t) \) where \( \mathbf{1}_u \in \mathbb{R}^n \) denotes the indicator vector at vertex \( u \). Hence classical Laplacian solvers [46, 13] provide almost-linear time algorithms to approximate \( R_{G}(s,t) \).

It has recently received increasing interest of designing sublinear-time (or local) algorithms for estimating effective resistances. In this setting, we are given query access to a graph and any specified vertex pair \( s \) and \( t \), our goal is to find a good approximation of the \( s-t \) effective resistance, by making as few queries as possible. Such algorithms are particularly motivated by the ubiquity of modern massive graphs, on which traditional polynomial-time algorithms are no longer feasible. Andoni et al. [4] gave an algorithm that \((1 + \epsilon)\)-approximates \( R_{G}(s,t) \) in \( O(\frac{1}{\epsilon^2} \text{poly log} \frac{1}{\epsilon}) \) time for \( d \)-regular expander graphs. Peng et al. [40] then generalized this algorithm to unbounded-degree expander graphs with an additive error \( \epsilon \) and similar query complexity. Li and Sachdeva [31] then gave one algorithm that \((1 + \epsilon)\)-approximates the \( s-t \) effective resistance in \( O(\text{poly}(\log n)) \) time on expander graphs (which is implicit in the proof of Theorem 3.1 in [31]). However, the question of whether it is possible to obtain sublinear-time estimation for effective resistances on non-expander or general graphs remains largely open.

Besides the aforementioned sublinear-time algorithms for effective resistances on expander graphs, one can observe an \( \Omega(n) \)-query lower bound for approximating the effective resistance of non-adjacent pair \( s \) and \( t \). Indeed, consider an \( n \)-vertex path, on which the \( s-t \) effective resistance is equivalent to the \( s-t \) shortest path length. Intuitively, for the latter problem, any algorithm with a constant approximation ratio needs to well estimate the number of edges on the path from \( s \) to \( t \), which takes \( \Omega(n) \) queries in the worst case.

In this paper, we consider the power and limitations of sublinear-time algorithms for \( s-t \) effective resistance such that \( s \) and \( t \) are adjacent, i.e., \((s,t) \in E(G)\). The adjacency case is already interesting in many applications. For example, in the seminal work on graph sparsification [45], it suffices to have good estimations of the effective resistances between the endpoints of edges, i.e., the adjacent pairs. It is also known that the effective resistance multiplied by the edge weight is equal to the probability that the edge belongs to a randomly generated spanning tree (see e.g. [18]), which has found applications in random spanning tree generation.

On the other hand, for an adjacency pair, the lower bound from the previous discussion does not hold any more, as for any pair \( s \) and \( t \) such that \((s,t) \in E\), their effective resistance is exactly 1 on a path. A priori, it could be true that a \((1 + \epsilon)\)-approximation of the \( s-t \) effective resistance for an adjacent pair can be found in sublinear time.
Now we state our results on sublinear-time algorithms for estimating the effective resistances. We will focus on unit-weighted graphs and the adjacency list model [41], in which the local algorithms can perform degree, neighbor queries and also sample vertices uniformly at random.

### Strong Lower Bounds.
First, we provide a strong lower bound for graphs of degree at most 3 except the given pair \( s \) and \( t \). For convenience, for a parameter \( C > 1 \), we say that value \( a \) is a \( C \)-approximation of value \( b \) if \( a \in [b/C, b] \). In the following, we always assume that the given pair \( s \) and \( t \) are adjacent, i.e., \((s, t) \in E\).

**Theorem 1.** There are infinitely many \( n \) and graphs of \( n \) vertices such that any local algorithm with success probability 0.6 and approximation ratio 1.01 on \( R_G(s, t) \) needs \( \Omega(n) \) queries. This holds even for graphs whose vertices are of degree at most 3 except the adjacent pair \( s \) and \( t \).

Next, observe that the trivial algorithm outputting 1 directly gives an approximation factor \( 2^{1/d(s)+1/d(t)} \) where \( d(s) \) and \( d(t) \) are the degrees of the adjacent pair \( s \) and \( t \). This is because \( R_G(s, t) \geq \frac{1}{2^{1/(d(s)+1/d(t))}} \) from the spectral graph theory [33]. This factor equals \( d \) if the graph is regular and \( d \) is between \( \min\{d(s), d(t)\} \) and \( 2 \min\{d(s), d(t)\} \) in general. If we consider large approximation factors instead of the \((1 + \epsilon)\)-approximation shown in Theorem 1, a natural question is can we design sublinear time algorithms that improve upon this trivial approximation ratio \( 2^{1/(d(s)+1/d(t))} \)? Our next theorem shows that it would take \( \Omega(n) \) queries to improve this ratio significantly.

**Theorem 2.** There exist a universal constant \( c > 0 \) and infinitely many \( n \) such that given any \( d \geq 4 \) and any \( \ell \geq 4 \), for graphs of \( n \) vertices and degree at most \( d \), any local algorithm to approximate \( R_G(s, t) \) with success probability 0.6 and approximation ratio \( 1 + c \cdot \min\{d, \ell\} \) needs \( \Omega(nd/\ell) \) queries.

If we set \( \ell = d \), this indicates that it takes \( \Omega(n) \) queries to obtain an approximation ratio \( o(d) \) for \( d \)-regular graphs since \( m = nd/2 \). In contrast, the corresponding trivial approximation ratio for adjacent pairs on \( d \)-regular graphs is \( d \). If we fix \( \ell \) and consider graphs with sufficiently large degree \( d \geq \ell \), Theorem 2 also gives a trade-off between the query complexity and approximation ratio. Next we remark that we could incorporate additive errors into the lower bounds of Theorem 1 and Theorem 2. This is because these two bounds consider the task of distinguishing between graphs with \( R_G(s, t) = 1 \) and graphs with \( R_G(s, t) < 1 \).

On the other hand, for graphs of degree at most 2 except the given pair \( s \) and \( t \), we provide a \((1 + \epsilon)\)-approximation algorithm of sublinear time in Section 5.

### Total effective resistance.
Another important measure of a network is the total effective resistance, which is defined as \( R_{tot}(G) = \sum_{u,v} R_G(u, v) \) and is also known as the Kirchhoff index of a graph (see e.g. [20, 19, 30] for numerous applications of \( R_{tot}(G) \)). Because of the elegant expression \( R_{tot}(G) = n \cdot \sum_{\lambda > 1} \frac{1}{\lambda(\lambda - 1)} \) where the sum is over all non-trivial eigenvalues of the Laplacian of \( G \), one may wonder whether there is a simpler approximation algorithm for the total effective resistance or not. However, we show that even for simple graphs with degree 2, any algorithm with approximation ratio \( 2 < 2 \) needs \( \Omega(n) \) queries.

**Theorem 3.** For any \( n \) and any \( \ell > 1 \), any local algorithm for computing the total effective resistance with success probability 0.6 and approximation ratio \( \ell \) needs \( \Omega(n) \) queries. In particular, this holds even for \( \ell = 2 - o(1) \) and graphs of degree at most 2.
Eigenvalues of graph perturbation. One technical ingredient in the proof of Theorem 1 is to show that an expander graph is still an expander after removing one edge. A natural approach would be to compare the smallest non-trivial eigenvalue of the Laplacian of the perturbed graph to the corresponding one of the original graph. Since deleting one edge in $G$ does not change its edge expansion too much, one may use Cheeger’s inequality (with other properties) to lower bound the new eigenvalue in terms of the original one. However, our key technical lemma provides a more direct bound by incorporating the effective resistance of the moved edge.

Lemma 4. Given a graph $G = (V, E)$ with $n$ vertices, let $\lambda_1(G) \leq \cdots \leq \lambda_n(G)$ be the eigenvalues of its Laplacian $L_G$. Given any edge $(u, v)$ in $G$, let $G' = (V, E \setminus \{(u, v)\})$ be the graph obtained from $G$ by removing the edge $(u, v)$ and let $\lambda_1(G') \leq \cdots \leq \lambda_n(G')$ be the eigenvalues of its Laplacian $L_{G'}$. Then it holds that

$$\forall i \in [n], \lambda_i(G') \geq (1 - R_G(u, v)) \cdot \lambda_i(G).$$

This lemma shows that after removing edge $(u, v)$ in an expander $G$ with $R_G(u, v)$ strictly less than 1, $G$ is still an expander. We remark that this requirement on the effective resistance $R_G(u, v)$ is necessary. Because there exists an expander graph $G$ with edges of effective resistance 1 and vertices of degree 1 (see Claim 10 in Section 3), one can not delete an arbitrary edge in $G$ while maintaining the expansion property.

We could consider the general problem of bounding the eigenvalues of a matrix after removing a few rows. Specifically, given a matrix $A \in \mathbb{R}^{m \times n}$ and $k$ distinct rows $a_{\ell_1}, \ldots, a_{\ell_k}$ from $A$, the question is to compare all eigenvalues of $(A')^\top A'$, where $A' \in \mathbb{R}^{(m-k) \times n}$ removes row $a_{\ell_1}, \ldots, a_{\ell_k}$ from $A$, to the eigenvalues of $A^\top A$. Since $(A')^\top A' \preceq A^\top A$, each eigenvalue $\lambda_i((A')^\top A') \leq \lambda_i(A^\top A)$. We give a lower bound for every $\lambda_i((A')^\top A')$ by incorporating the leverage scores of those rows.

The (statistical) leverage scores of a matrix $A \in \mathbb{R}^{m \times n}$ provide a nonuniform importance sampling distribution over the $m$ rows of $A$, which plays a crucial role in randomized matrix algorithms (see [47] for a list of applications). For each row $a_i$ in $A$, its leverage score $\tau_i$ is defined to be $a_i^\top \cdot (A^\top A)^{\dagger} \cdot a_i$. In fact, the leverage scores of a matrix are the analogues of the effective resistances of a graph [16]. More formally, let $B \in \mathbb{R}^{m \times n}$ denote the edge-incidence matrix of a graph $G$ as follows: Each edge $(u, v)$ of $G$ gives a row $1_u - 1_v$ in $B$. Furthermore, the Laplacian matrix $L$ of $G$ equals $B^\top B$. If a row of $B$ corresponds to edge $(u, v)$, its leverage score $(1_u - 1_v)^\top \cdot (B^\top B)^{\dagger} \cdot (1_u - 1_v) = (1_u - 1_v)^\top \cdot L^{\dagger} \cdot (1_u - 1_v)$ turns out to be the effective resistance of $(u, v)$ in $G$.

Now we state Lemma 5 for the general problem. So Lemma 4 is a direct corollary of Lemma 5 by setting $A$ to the incidence matrix of $G$ as discussed above.

Lemma 5. Given a matrix $A \in \mathbb{R}^{m \times n}$, let $\lambda_1 \leq \cdots \leq \lambda_n$ be the eigenvalues of $A^\top A$. Moreover, for each $\ell \in [m]$, let $a_{\ell}$ be row $\ell$ of $A$ and $\tau_{\ell} = a_{\ell}^\top (A^\top A)^{\dagger} a_{\ell}$ be its leverage score.

For any $k$ distinct indices $\ell_1, \ldots, \ell_k \in [m]$, let $A' \in \mathbb{R}^{(m-k) \times n}$ be the matrix obtained from $A$ by removing the corresponding $k$ rows $a_{\ell_1}, \ldots, a_{\ell_k}$; and let $\lambda'_1 \leq \cdots \leq \lambda'_n$ be the eigenvalues of $(A')^\top A'$. It holds that

$$\forall i \in [n], \lambda'_i \in [(1 - \tau_{\ell_1} - \tau_{\ell_2} - \cdots - \tau_{\ell_k}) \cdot \lambda_i, \lambda_i].$$

1.1 Related Work

Previous research has studied the problem of how to quickly compute and approximate the effective resistances in the regime of polynomial-time algorithms, as such algorithms can be used as a crucial subroutine for other graph algorithms. For example, for any two vertices $s$
and \( t \) in a \( n \)-vertex \( m \)-edge graph, one can \((1 + \epsilon)\)-approximate the \( s-t \) effective resistance in \( \tilde{O}(m + nc^{-2}) \) [18] and \( O(m \log(1/\epsilon)) \) [13] time, respectively. To \((1 + \epsilon)\)-approximate the effective resistances between \( s \) given pairs, Chu et al. [11] provided an algorithm in time \( O(m^{1+\Theta(1)} + (n + s)n^{o(1)}.c^{-1.5}) \). There are also algorithms that find \((1 + \epsilon)\)-approximations to the effective resistance between every pair of vertices in \( \tilde{O}(n^2/\epsilon) \) time [27]. In order to exactly compute the \( s-t \) or all-pairs effective resistance, the current fastest algorithms run in times \( O(n^\omega) \) (by using the fastest matrix inversion algorithm [8, 24]), where \( \omega < 2.373 \) is the matrix multiplication exponent [3]. Faster algorithms are known for planar graphs by using the nested dissection method [32].

There exists a line of works on how to efficiently maintain the effective resistances dynamically [42, 1, 21, 22, 17, 9], i.e., if the graph undergoes edge insertions and/or deletions, and the goal is to support the update operations and query for the effective resistances as quickly as possible, rather than having to recompute it from scratch each time.

For the total effective resistance, Ghosh et al. [20] studied algorithms for allocating edge weights on a given graph in order to minimize the total effective resistance. Li and Zhang [30] used Kirchhoff index (i.e., total effective resistance) as the measure of edge centrality in weighted networks and gave efficient algorithms for the measure.

Matrix perturbation considers the eigenvalues (singular values) of \( A \) after adding a matrix \( E \) of the same order. Various bounds on the error of matrix perturbation, such as absolute errors and relative errors, has been studied. We refer to the survey [26] and the reference therein for a complete overview. Even though Lemma 5 is an instantiation of Theorem 2.8 in [26] with (statistical) leverage scores, we provide two simpler proofs in this work which are more intuitive. Also, different perturbation bounds in terms of the leverage scores were provided in [25, 23].

Leverage scores, analogue to effective resistances of graphs, have wide applications in randomized matrix algorithms and large-scale data algorithms. The most notable property is that sampling the rows of a matrix \( A \) via its leverage scores gives an efficient construction of the subspace embedding of \( A \) [45]. This fact is extremely useful in designing ultra-efficient algorithms for linear regression and low rank approximations [12]. We refer to the survey [47] for a list of applications. Since bounding eigenvalues are sufficient for \( \ell_2 \)-subspace embeddings, there is a line of research on the connection between eigenvalues and leverage scores, such as spectral sparsifications [45, 5].

Sublinear-time algorithms for the related graph problems has also been investigated. For example, Lee [29] gave an algorithm for producing a probabilistic \((\epsilon, \delta)\)-spectral sparsifier with \( O(n \log n/\epsilon^2) \) edges in \( \tilde{O}(n/\epsilon^2\delta) \) time for unweighted undirected graph. Note that its running time is sublinear if the number of edges in the graph is large enough. For spectral approximations in sublinear time, various approximation guarantees have been studied in [14, 39, 7].

**Organization.** In Section 2, we provide the basic definitions and notations of this work. Next we discuss about eigenvalues of perturbed matrices and graphs in Section 3. Then we prove the lower bounds of Theorem 1 in Section 4. Finally, we discuss the approximation algorithm for degree-2 graphs in Section 5. Due to the space constraint, we leave the proof of Theorem 2 and 3 to the full version of this paper in arXiv.

## 2 Preliminaries

For any integer \( k \geq 1 \), let \( [k] := \{1, \cdots, k\} \). We use \( a = b \pm c \) to denote \( a \in [b - c, b + c] \) and \( 1 \) to denote the all 1 vector.
Basic definitions from graph theory. In this work, we only consider undirected graphs with unit weights on each edge. Given an undirected graph $G = (V,E)$ with $n := |V|$ vertices and $m := |E|$ edges, let $A_G \in \mathbb{R}^{n \times n}$ denote the adjacency matrix of $G$ and $D_G \in \mathbb{R}^{n \times n}$ denote its degree diagonal matrix. We use $L_G \in \mathbb{R}^{n \times n}$ to denote its Laplacian, i.e., $L_G = D_G - A_G$. Also, we use $V(G)$ and $E(G)$ to denote the vertex set and edge set of a graph $G$.

In this work, we use $\tilde{L}_G$ to denote the normalized Laplacian of $G$, i.e., $\tilde{L}_G := D_G^{-1/2} \cdot L_G \cdot D_G^{-1/2}$. When the graph is clear, we hide the notation $G$. Also, we use $V(G)$ and $E(G)$ to denote the vertex and edge set of $G$. Moreover, we use $d$ to denote the maximum degree of $G$. For a vertex $u$, let $d(u)$ denote its degree in $G$ and $1_u \in \mathbb{R}^n$ denote the indicator vector of $u$, i.e., $1_u(v) = 1$ if $v = u$ and 0 otherwise.

Now we define the adjacency list model for sublinear time graph algorithms [41]. There are three types of operations in constant time:

1. degree query: the algorithm queries the degree of a fixed vertex $v \in V$;
2. neighbor query: the algorithm queries the $i$-th neighbor of vertex $v$ given $v$ and $i$;
3. uniform sampling: the algorithm receives a random vertex in $V$.

Basic definitions about matrices and expander graphs. We use $\psi, \phi$, and bold letter $\mathbf{a}$ to denote vectors and $\| \cdot \|$ to denote their $L_2$ (Euclidean) norms. For a vector $\mathbf{a} \in \mathbb{R}^V$ and a subset $U \subset V$, let $\mathbf{a}(U)$ denote the vector in $\mathbb{R}^U$ which contains the corresponding entries in $\mathbf{a}$.

Given a symmetric matrix $A$, we always use $\lambda_1(A) \leq \lambda_2(A) \leq \cdots \leq \lambda_n(A)$ to denote its eigenvalues in the non-decreasing order. Furthermore, let its eigendecomposition be $A = \sum_i \lambda_i(A) \cdot \psi_i \psi_i^T$, where $\psi_i$ is the eigenvector corresponding to the eigenvalue $\lambda_i(A)$.

We say $G$ is an expander if the second smallest eigenvalue $\lambda_2(\tilde{L})$ of $\tilde{L}$ is at least $c_1$, for some universal $c_1 > 0$. This is equivalent to $\lambda_2(L_G) \geq c_2$ for some $c_2 > 0$ when the degree of $G$ is bounded. We will use Ramanujan graphs [36, 38] of degree 3 and near-Ramanujan graphs for every degree $d \geq 4$ [37]. The guarantee of a Ramanujan graph $G$ of regular degree $d$ is that $\lambda_2(L_G) \geq d - 2\sqrt{d-1}$. For near-Ramanujan graphs, we only need $\lambda_2(L_G) \geq d - 2.01\sqrt{d-1}$.

Basic definitions about effective resistances. Then we define the Moore-Penrose pseudo-inverse and effective resistances. For a symmetric matrix $M \in \mathbb{R}^{n \times n}$ whose eigendecomposition is $M = \sum_i \psi_i \psi_i^T$, its Moore-Penrose pseudo-inverse $M^\dagger = \sum_{i : \lambda_i \neq 0} \frac{1}{\lambda_i} \psi_i \psi_i^T$.

**Definition 6 (Effective Resistances).** Given a graph $G = (V,E)$, for any two vertices $s, t \in V$, the $s - t$ effective resistance is defined as $R_G(s,t) := (1_s - 1_t)^\top \cdot L_G^{-1} \cdot (1_s - 1_t)$. Moreover, the total effective resistance of $G$ is defined as $R_{tot}(G) = \sum_{i<j} R_G(i,j)$.

In this work, we will extensively use the following facts about effective resistances (see [33, 44] for their proofs).

**Lemma 7.** Given a graph $G = (V,E)$, the effective resistances in $G$ satisfy the following properties:

1. $\sum_{(u,v) \in E} R_G(u,v) = n - 1$ and $R_{tot}(G) = n \cdot \sum_{i=1}^n \frac{1}{\lambda_i(L_G)}$.
2. $2m \cdot R_G(u,v) = \kappa_G(u,v)$, where $\kappa_G(i,j)$ is the commute time of a simple random walk from vertex $i$ to $j$ in $G$, i.e., the expected number of steps in a random walk starting at $i$, after vertex $j$ is visited and then vertex $i$ is reached again.
3. $\frac{1}{2} \left( \frac{1}{\lambda_1(L_G)} + \frac{1}{\lambda_2(L_G)} \right) \leq R_G(u,v) \leq \left( 1/\lambda_2(\tilde{L}_G) \right) \cdot \left( \frac{1}{\lambda_1(L_G)} + \frac{1}{\lambda_2(L_G)} \right)$ where $\lambda_2(\tilde{L}_G)$ is the 2nd smallest eigenvalue of the normalized Laplacian $\tilde{L}_G$. 

29:6 Effective Resistances in Non-Expander Graphs
4. Given any $(s,t)$, consider all functions $\phi \in \mathbb{R}^V$ such that $\phi(s) = 1$ and $\phi(t) = 0$, then

$$R_G(s, t) = \min_{\phi \in \mathbb{R}^V: \phi(s) = 1, \phi(t) = 0} \frac{1}{\sum_{(u,v) \in E} (\phi(u) - \phi(v))^2}.$$ 

In fact, the minimum value above is achieved when $\phi$ is harmonic, namely the unique solution satisfying $\phi(s) = 1, \phi(t) = 0$, and $\phi(v) = \frac{1}{d_G(v)} \sum_{(u,v) \in E} \phi(u)$ for $v \in V \setminus \{s,t\}$.

An equivalent definition is $R_G(s, t) = \max_{\phi \in \mathbb{R}^n: \phi(1) = 1} \frac{(1,\ldots,1,\phi)}{\phi \cdot L_G \phi}$. In particular, the leverage score $\tau_i$ of row $a_i$ in $A$ also equals $\tau_i = \max_{\phi \in \mathbb{R}^n: \phi(1) = 1} \frac{\|a_i\|_2^2}{\phi \cdot a_i^\top}$.

5. Let $T(G)$ denote all spanning trees in $G$. Let us pick $T \in T$ uniformly at random. Then $R_G(u,v)$ is the probability $(u,v)$ is in $T$, i.e., $R_G(u,v) = \Pr_{T \sim T(G)}[(u,v) \in T]$.

Note that the first equation $\sum_{(u,v) \in E} R_G(u,v) = n - 1$ in Lemma 7 considers the summation over all edges in $E$ where the total effective resistance is the summation over all pairs.

3 Eigenvalues of Perturbed Graphs and Matrices

We discuss Lemma 4 and Lemma 5 in this section. We give two different proofs for Lemma 5 such that Lemma 4 is a direct corollary by setting $A$ to be the incidence matrix. Then we show expander graphs with edges of effective resistance 1 to illustrate that we have to remove edges with $R_G(u,v) < 1$ to keep the perturbed graph as an expander in Lemma 4.

We restate Lemma 5 again. One remark is that both bounds could be tight in Lemma 5. For example, consider the case $A^\top A = I$ whose $\tau_i = \|a_i\|_2^2$ for all $i$ and $\lambda_i \equiv 1$ for all $i$. Then after removing any $a_i$ (hence $k = 1$), $(A^\top A - a_i a_i^\top)$ has $\lambda'_1 = 1 - \|a_i\|_2^2$ with eigenvector $a_i / \|a_i\|_2$. So $\lambda'_1$ matches the lower bound $(1 - \tau_i) \cdot \lambda_i$; and all the rest eigenvalues of $(A^\top A - a_i a_i^\top)$ are 1, the same as those of $A^\top A$.

**Lemma 5.** Given a matrix $A \in \mathbb{R}^{m \times n}$, let $\lambda_1 \leq \cdots \leq \lambda_n$ be the eigenvalues of $A^\top A$. Moreover, for each $\ell \in [m]$, let $a_\ell$ be row $\ell$ of $A$ and $\tau_\ell = a_\ell^\top (A^\top A) a_\ell$ be its leverage score.

For any $k$ distinct indices $\ell_1, \ldots, \ell_k \in [m]$, let $A' \in \mathbb{R}^{(m-k) \times n}$ be the matrix obtained from $A$ by removing the corresponding $k$ rows $a_{\ell_1}, \ldots, a_{\ell_k}$; and let $\lambda'_1 \leq \cdots \leq \lambda'_n$ be the eigenvalues of $(A')^\top A'$. It holds that

$$\forall i \in [n], \lambda'_i \in [(1 - \tau_{\ell_1} - \cdots - \tau_{\ell_k}) \cdot \lambda_i, \lambda_i].$$

The first proof is based on the characteristic polynomial of $A^\top A$, which is motivated by the potential function method of classical work [5]. One ingredient in this proof is the matrix determinant lemma [5].

**Lemma 8.** If $A$ is nonsingular and $v$ is a vector, then $\det(A + vv^\top) = \det(A) \cdot (1 + v^\top A^{-1} v)$.

**Proof of Lemma 5.** For ease of exposition, we start with $k = 1$. Namely, we remove one row $a_\ell$ from $A$ to obtain $A'$. Consider the characteristic polynomial of $(A')^\top A'$:

$$\det(xI - (A')^\top A') = \det((xI - A^\top A) + a_\ell a_\ell^\top)$$

$$= \det((xI - A^\top A) \cdot (I + (xI - A^\top A)^\top a_\ell a_\ell^\top))$$

$$= \det((xI - A^\top A) \cdot (I + (xI - A^\top A)^\top a_\ell a_\ell^\top).$$

Let $\psi_\ell$ be the corresponding eigenvector of $\lambda_\ell$ in $A$ such that $(xI - A^\top A)^\top a_\ell a_\ell^\top = \left(\sum_{i \in [n], i \neq \ell} \frac{1}{x - \lambda_i} \psi_\ell \psi_i^\top\right) a_\ell a_\ell^\top$. Then we apply the matrix determinant lemma (Lemma 8) to

$$\det((xI + (xI - A^\top A)^\top a_\ell a_\ell^\top)$$

such that
\[
det(xI - (A^\top A)' = det(xI - A^\top A) \cdot \left(1 + a_\ell^\top \cdot \sum_{i=1}^{n} \frac{1}{x - \lambda_i} \psi_i \psi_i^\top \cdot a_\ell \right) \\
= det(xI - A^\top A) \cdot \left(1 - \sum_{i=1}^{n} \frac{1}{\lambda_i - x} \langle \psi_i, a_\ell \rangle^2 \right).
\]

Let \(\lambda_1', \lambda_2', \ldots, \lambda_n'\) be the roots of \(det(xI - (A^\top A)')\). Similar to the argument in [5]: If \(\langle \psi_i, a_\ell \rangle = 0\), then \(\lambda_i\) is a root of \(det(xI - A^\top A)\), i.e., \(\lambda_i' = \lambda_i\). Otherwise, \(\lambda_i' \in (\lambda_{i-1}, \lambda_i)\) satisfies \(\sum_{j=1}^{n} \frac{1}{\lambda_j - \lambda_i} \langle \psi_j, a_\ell \rangle^2 = 1\). This is because \(\lim_{x \to \lambda_{i-1} + \sum_{i=1}^{n} \frac{1}{\lambda_i - x} \langle \psi_i, a_\ell \rangle^2 = -\infty\)

and \(\lim_{x \to \lambda_i - \sum_{i=1}^{n} \frac{1}{\lambda_i - x} \langle \psi_i, a_\ell \rangle^2 = +\infty\). For the 2nd case, we show \(\lambda_i' \geq (1 - \tau_\ell)\lambda_i\).

Let the function \(p(x) := \sum_{j=1}^{n} \frac{1}{x - \lambda_j} \langle \psi_j, a_\ell \rangle^2\). If \(\lambda_{i-1} \geq (1 - \tau_\ell)\lambda_i\), then we have proved \(\lambda_i' > (1 - \tau_\ell)\lambda_i\). Otherwise we show \(\lambda_i' > (1 - \tau_\ell)\lambda_i\) by considering \(p(x)\) in the continuous interval \([(1 - \tau_\ell)\lambda_i, \lambda_i]\).

\[
p((1 - \tau_\ell)\lambda_i) = \sum_{j=1}^{n} \frac{1}{\lambda_j - (1 - \tau_\ell)\lambda_i} \langle \psi_j, a_\ell \rangle^2 \\
\leq \sum_{j=1}^{n} \frac{1}{\lambda_j - (1 - \tau_\ell)\lambda_i} \langle \psi_j, a_\ell \rangle^2
\]

(Since \(\lambda_1 < \cdots < \lambda_{i-1} < (1 - \tau_\ell)\lambda_i\) from the assumption, their corresponding terms are negative.)

\[
\leq \sum_{j=1}^{n} \frac{1}{\lambda_j - (1 - \tau_\ell)\lambda_j} \langle \psi_j, a_\ell \rangle^2 = \frac{1}{\tau_\ell} \sum_{j=1}^{n} \frac{1}{\lambda_j} \langle \psi_j, a_\ell \rangle^2
\]

From the definition, \(\tau_\ell = a_\ell^\top (A^\top A)^{\dagger} a_\ell = a_\ell^\top \left(\sum_{i=1}^{n} \frac{1}{\lambda_i} \psi_i \psi_i^\top\right) a_\ell = \sum_{i=1}^{n} \frac{1}{\lambda_i} \langle \psi_i, a_\ell \rangle^2\). So \(p((1 - \tau_\ell)\lambda_i) \leq 1\).

On the other hand, \(\lim_{x \to \lambda_i} \sum_{j=1}^{n} \frac{1}{x - \lambda_j} \langle \psi_j, a_\ell \rangle^2 = +\infty\). So \(p((1 - \tau_\ell)\lambda_i) \leq 1\) and \(p(\lambda_i - \epsilon) > 1\) infer that there exists a \(x \in [(1 - \tau_\ell)\lambda_i, \lambda_i]\) such that \(p(x) = 1\), which also means that \(det(xI - (A^\top A)')\) has a root \(\lambda_i' \in [(1 - \tau_\ell)\lambda_i, \lambda_i]\).

Next we prove \(\lambda_i' \geq (1 - \tau_\ell - \cdots - \tau_k)\lambda_i\) by induction on \(k\). The above calculation proves the base case of \(k = 1\).

For the inductive step, let \(\tilde{A}\) denote the matrix after removing \(a_{\ell_{i+1}}, \ldots, a_{\ell_k}\) and \(A'\) denote the matrix by removing one more edge \(a_{\ell_{i+1}}\). By the induction hypothesis, \(\lambda_i(\tilde{A}) \in \left(1 - \sum_{j=1}^{q} \tau_j\right) \cdot \lambda_i, \lambda_i\) and \(1 - \sum_{j=1}^{q} \tau_j\) \(A^\top A \preceq \tilde{A}^\top \tilde{A}\). This implies \((\tilde{A}^\top \tilde{A})^\dagger \preceq (1 - \sum_{j=1}^{q} \tau_j)^{-1}\).

\[
\tilde{\tau}_{i+1} = a_{\ell_{i+1}}^\top (\tilde{A}^\top \tilde{A})^\dagger a_{\ell_{i+1}} \leq \left(1 - \sum_{j=1}^{q} \tau_j\right)^{-1} a_{\ell_{i+1}}^\top (A^\top A)^\dagger a_{\ell_{i+1}} = \left(1 - \sum_{j=1}^{q} \tau_j\right)^{-1} \tau_{i+1}.
\]

Using the perturbation bound for \(k = 1\) on \(\tilde{A}\) and \(A'\),

\[
\lambda_i(\tilde{A}) \in [(1 - \tilde{\tau}_{i+1})\lambda_i(\tilde{A}), \lambda_i(\tilde{A})] \\
\subseteq \left[\left(1 - \left(1 - \sum_{j=1}^{q} \tau_j\right)^{-1} \tilde{\tau}_{i+1}\right) \left(1 - \sum_{j=1}^{q} \tau_j\right)^{-1} \lambda_i, \lambda_i\right] = \left[\left(1 - \sum_{j=1}^{q} \tau_j\right) \cdot \lambda_i, \lambda_i\right].
\]
Next we present the 2nd proof, which is based on Property 4 of the leverage score in Lemma 7. One advantage of this proof is that it works directly on multiple edges.

**Proof of Lemma 5.** We will apply the Courant-Fischer theorem below, which shows \( \lambda_1 = \min_{S \subseteq \mathbb{R}^n, \dim(S) = i} \max_{\phi \in S} \frac{\phi^T (A')^T A' \phi}{\phi^T \phi}. \)

\[ \lambda_k(H) = \min_{\dim(S) = k} \max_{x \in S} \frac{x^T H x}{x^T x} \quad \text{and} \quad \max_{\dim(S) = n-k+1} \min_{x \in S} \frac{x^T H x}{x^T x}. \]

We compare the Rayleigh quotient \( \frac{\phi^T (A')^T A' \phi}{\phi^T \phi} \) with \( \frac{\phi^T A^T A \phi}{\phi^T \phi} \).

\[ \frac{\phi^T (A')^T A' \phi}{\phi^T \phi} = \frac{\|A' \phi\|^2_2}{\|\phi\|^2_2} = \frac{\|A \phi\|^2_2 - \sum_{j=1}^{k} \left( a_{i,j}^T \phi \right)^2}{\|\phi\|^2_2} = \frac{\|A \phi\|^2_2}{\|\phi\|^2_2} \left( 1 - \sum_{j=1}^{k} \frac{(a_{i,j}^T \phi)^2}{\|A \phi\|^2_2} \right). \]

Since \( \tau_\ell = \max_{\phi \in \mathbb{R}^n} \frac{(a_{i,j}^T \phi)^2}{\|A \phi\|^2_2} \) from Property 4 of Lemma 7, \( 1 - \sum_{j=1}^{k} \frac{(a_{i,j}^T \phi)^2}{\|A \phi\|^2_2} \geq 1 - \sum_{j=1}^{k} \tau_\ell. \)

Using Lemma 9 again,

\[ \lambda_i' = \min_{\dim(S) = i} \max_{\phi \in S} \frac{\|A \phi\|^2_2}{\|\phi\|^2_2} \left( 1 - \sum_{j=1}^{k} \frac{(a_{i,j}^T \phi)^2}{\|A \phi\|^2_2} \right) \geq \left( 1 - \sum_{j=1}^{k} \tau_\ell \right) \min_{\dim(S) = i} \max_{\phi \in S} \frac{\|A \phi\|^2_2}{\|\phi\|^2_2} = (1 - \sum_{j=1}^{k} \tau_\ell) \lambda_i. \]

As \( 1 - \sum_{j=1}^{k} \frac{(a_{i,j}^T \phi)^2}{\|A \phi\|^2_2} \leq 1 \), we can get \( \lambda_i' \leq \lambda_i \) in a similar way. Combining the two inequalities,

\[ \lambda_i' \in [(1 - \sum_{j=1}^{k} \tau_\ell) \lambda_i, \lambda_i]. \]

We remark that Lemma 5 is also an instantiation of Theorem 2.8 in [26] with leverage scores. However, we believe the above two proofs shed more insights on the structure of perturbed matrices and are simpler (without Weyl’s interlacing inequality). But for completeness, we provide that proof in the full version.

Next we show that there exist edges in expander graphs with unit effective resistance.

**Claim 10.** For any \( c_0 > 0 \) and infinitely many \( n \), there exists an expander graph \( G \) of constant degree such that
1. The smallest non-trivial eigenvalue of its Laplacian is at least \( c_0 \);  
2. There exists an edge \( e \) in \( G \) with effective resistance 1.

Hence, after removing \( e \), \( \lambda_2(L_G) = 0 \) in the perturbed graph \( G' \) such that \( G' \) is no longer an expander.

**Proof.** Given an expander \( G' = (V', E') \) with constant degree and \( \lambda_2(L_G) = \Omega(1) \), we add an extra vertex of degree 1 to it. More formally, let \( V' = \{1, \ldots, n-1\} \) and \( G = (V, E) = (V' \cup \{1\}, E' \cup \{(n-1, n)\}) \). Then \( R_G(n-1, n) = 1 \). We will show \( \lambda_2(L_{G'}) = \Omega(1) \) such that \( G' \) is also an expander (its degree is still a constant). So \( L_G(i, j) = L_{G'}(i, j) \) except \( i \in \{n-1, n\} \) and \( j \in \{n-1, n\} \) where those four entries have \( L_G(n-1, n-1) = L_G(n-1, n) + 1, L_G(n-1, n) = L_G(n, n-1) = -1, \) and \( L_G(n, n) = 1 \).
By the Courant-Fischer Theorem, $\lambda_2(L_G) = \min_{\phi \perp 1, ||\phi||_2 = 1} \phi^T L_G \phi$. Then we prove $\phi^T L_G \phi \Omega(1)$ for all $\phi$ with $\phi \perp 1$ and $||\phi||_2 = 1$ by considering $\phi(n)$ in two cases.

If $|\phi(n)| \geq 0.8$, then $|\phi(n-1)| \leq 0.6$ because $||\phi||_2 = 1$. So $\phi^T L_G \phi = \sum_{(u,v) \in E} (\phi(u) - \phi(v))^2 \geq (\phi(n) - \phi(n-1))^2 \geq 0.04$.

Otherwise $|\phi(n)| < 0.8$. For convenience, we assume $\phi(n) \in [0,0.8)$. Let $\phi([n-1]) \in \mathbb{R}^{[n-1]}$ denotes the sub-vector on $V'$ and $1([n-1])$ denotes the all 1-vector on $V'$. Then $||\phi([n-1])|| \geq 0.6$ and

\[ \phi^T L_G \phi = \sum_{(u,v) \in E} (\phi(u) - \phi(v))^2 \geq \sum_{(u,v) \in E'} (\phi(u) - \phi(v))^2 = \phi([n-1])^T \cdot L_G' \cdot \phi([n-1]). \]

Since $L_G'$ has an eigenvalue 0 with eigenvector $1([n-1])$, we consider $\phi([n-1])$ after orthonormalization: $\phi([n-1]) - \langle [n-1], 1([n-1]) \rangle \cdot 1([n-1])$.

Note that $\langle [n-1], 1([n-1]) \rangle = -\phi(n)$. We calculate its $L_2$ norm after orthonormalization as

\[ ||\phi([n-1]) - \langle [n-1], 1([n-1]) \rangle \cdot 1([n-1])||^2 = ||\phi([n-1]) + \frac{\phi(n)}{n-1} \cdot 1([n-1])||^2 = ||\phi([n-1])||^2 + \frac{2\phi(n)}{n-1} \cdot \langle [n-1], 1([n-1]) \rangle \geq \frac{2\phi(n)}{n-1} \cdot \phi(n) \geq 0.36 - \frac{2}{n-1} \geq 0.3. \]

So $\phi^T L_G \phi \geq \lambda_2(L_G') \cdot ||\phi([n-1]) - \langle [n-1], 1([n-1]) \rangle \cdot 1([n-1])||^2 = \Omega(1)$.

From all discussion above, $\lambda_2(L_G) = \Omega(1)$. Since $G$ has a constant degree, $\lambda_2(L_G) / d(G) = \Omega(1)$, which means $G$ is also an expander.

4 Lower Bound for Degree 3

In this section, we prove the lower bound for graphs with degrees at most 3 (except the given pair $s$ and $t$). Recall the statement of Theorem 1 in Section 1.

\begin{theorem}
There are infinitely many $n$ and graphs of $n$ vertices such that any local algorithm with success probability 0.6 and approximation ratio 1.01 on $R_G(s,t)$ needs $\Omega(n)$ queries. This holds even for graphs whose vertices are of degree at most 3 except the adjacent pair $s$ and $t$.
\end{theorem}

One ingredient in the proof is Lemma 4, which bounds the expansion of a perturbed graph $G'$ obtained from removing one edge $e$ of the original graph $G$, in terms of the eigenvalues of the Laplacian of $G$ and the effective resistance $R_G(e)$.

In the rest of this section, we finish the proof of Theorem 1. The high level idea is to consider a graph $G$ of degree 3 (see Figure 2 for an illustration) constituted by two disjoint expanders $H_s$ and $H_t$ with one extra edge between $s$ and $t$ in these two expanders separately.

Then we produce a random graph $G'$ as follows (see Figure 1 for an illustration): remove one random edge $(u,v)$ in $H_s$ and another one $(v,v')$ in $H_t$ separately; then add two edges $(u,v)$ and $(v',v)$ to $G'$. 

\begin{figure}[h]
So the number of vertices in $G$ and $G'$ are the same; moreover all vertices have the same degrees. Thus the only way to distinguish between $G$ and $G'$ is to figure out whether any of these four edges $(u, u')$, $(v, v')$, $(u, v)$, $(u', v')$ is in the graph or not. Since there are $3n/2$ edges, this needs $\Omega(n)$ queries. Finally we bound $R_{G'}(s, t)$ (and the approximation ratio) via Lemma 4.

Proof of Theorem 1. We provide a distribution $\mathcal{G}$ of graphs with $n$ vertices and degree at most 3 (except $s$ and $t$). Then by Yao’s minimax principle, we only need to consider deterministic algorithms of $q_n$ neighbor queries and 1.01-approximation ratio whose success probability is at least 0.6 over $\mathcal{G}$. Our goal is to prove $q_n = \Omega(n)$.

Consider any $n$ such that there exists a 3-regular Ramanujan graph $H$ of size $n/2$ [36, 38]. Then we construct $G$ with the given pair $(s, t)$ as follows:

1. Take two vertex-disjoint copies of $H$, denoted by $H_s$ and $H_t$, such that $H_s$ contains vertex $s$ and $H_t$ contains vertex $t$.
2. Define the vertex set $V(G)$ of $G$ to be the union of the vertex sets of $H_s$ and $H_t$.
3. Define the edge set $E(G)$ of $G$ to be the union of $(s, t)$, the edge set $E(H_s)$ of $H_s$ and the edge set $E(H_t)$ of $H_t$.

Note that the effective resistance $R_{G}(s, t) = 1$ in $G$. 

Figure 1 $G'$ after modification. 

Figure 2 Construction of $G$ with two expanders $H_s$ and $H_t$. 

Figure 2
Next, we construct a random graph $G'$ based on $G$. Let $E_{3/4}$ be the set of edges in $H_s$ with effective resistance at most $3/4$, i.e., $E_{3/4} = \{(u, v) \in E(H_s) | R_G(u, v) \leq 3/4\}$ and we define $E_{1/3}$ analogously. We use the following claim to lower bound the sizes of these two sets whose proof is deferred to Section 4.1.

\[ \text{Claim 11.} \quad \text{It holds that } |E_{3/4}| \geq n/12 \text{ and } |E_{1/3}| \geq n/12. \]

We give the construction of $G'$ which is almost identical to $G$ except four edges.

1. Choose one edge $(u, u')$ uniformly at random from $E_{3/4}$ and remove it from $G$. Similarly, remove another random edge $(v, v')$ from $E_{1/3}$. For convenience, let $H_s'$ be the subgraph of $H_s$ obtained by removing $(u, u')$ from $H_s$ and let $H_t'$ be the subgraph obtained by removing $(v, v')$ from $H_t$.

2. Add $(u, v)$ and $(u', v')$ to $G'$.

By our choices of $(u, u'), (v, v') \in E_{3/4}$ and $(v, v') \in E_{1/3}$, $H_s'^'$ and $H_t'^'$ are still expander graphs from Lemma 4. Based on this property, we show the effective resistance between $s$ and $t$ in $G'$ is strictly less than 1 in Claim 12, whose proof is deferred to Section 4.1.

\[ \text{Claim 12.} \quad \text{It holds that } R_{G'}(s, t) \leq 0.99. \]

Next we fix a deterministic algorithm $A$ with approximation ratio at most 1.01 and consider the underlying distribution $G$, which is $G$ or $G'$ with probability 1/2 separately. Observe that whenever $A$ succeeds, $A$ is able to distinguish between $G$ and $G'$, since the ratio between $R_G(s, t) = 1$ and $R_{G'}(s, t) \leq 0.99$ is more than 1.01. For convenience, let us modify $A$ so that its output is an assertion about whether the input graph is $G$ or $G'$. By Yao’s minimax principle, it holds that

$$0.6 \leq \Pr_{G'}[A \text{ succeeds}] = \frac{\Pr[A(G) = G]}{2} + \frac{\Pr[A(G') = G']}{2}. \quad (1)$$

Next we consider all neighbor queries made by $A$ when the underlying graph is $G$. Since $A$ and $G$ are fixed, say $A$ makes $q_n$ fixed neighbor queries on $G$. If $G'$ and $G$ provide the same answers on these neighbor queries, $A$ fails to distinguish them. But $G'$ is obtained from $G$ by removing one random edge in $E_{3/4}$ and another one in $E_{1/3}$ separately. Hence at most $q_n$ edges in $E_{3/4}$ will be queried; and similarly for $E_{1/3}$. So we bound

$$\Pr[A(G') = G'] \leq \Pr[A(G) = G'] + \Pr[\text{One neighbor query returns different values}] \leq 1 - \Pr[A(G) = G] + q_n/|E_{3/4}| + q_n/|E_{1/3}|.$$ 

Plugging this into (1) with the two bounds in Claim 11, we obtain $q_n \geq 0.1 \cdot \frac{n}{12}$.

We remark that replacing $H$ by an expander of degree $\lceil m/n \rceil$ would give a lower bound $\Omega(m)$ instead of $\Omega(n)$, but this result is covered by Theorem 2 with $\ell = 1$.

### 4.1 Proofs of Claim 11 and Claim 12

We now give the proof of Claim 11.

**Proof of Claim 11.** Recall that $E_{3/4} = \{(u, v) \in E(H_s) | R_G(u, v) \leq 3/4\}$. First, for any edge $(u, v)$ in $H_s$, $R_{H_s}(u, v) = R_G(u, v)$. Since $(s, t)$ is the unique edge between $H_s$ and $H_t$, the set of spanning trees $\mathcal{T}(G)$ is generated by picking one spanning tree $T_1 \in \mathcal{T}(H_s)$ and one $T_2 \in \mathcal{T}(H_t)$ then connecting them by $\{(s, t)\}$. Then by Property 5 of Lemma 7, $R_{H_s}(u, v) = R_{H_s}(u, v)$. Thus, $E_{3/4} = \{(u, v) \in E(H_s) | R_{H_s}(u, v) \leq 3/4\}$. 


From Property 1 of Lemma 7, \(\sum_{(u,v) \in E(H_s)} R_{H_s}(u,v) = \frac{n}{2} - 1\). By the definition of \(E_{s,3/4}\), we have \(|E(H_s) \setminus E_{s,3/4}| = \frac{\lambda_2(L_{H_s})}{\lambda_1(L_{H_s})} \leq \frac{n}{3} - \frac{\lambda_2(L_{H_s})}{\lambda_1(L_{H_s})} \leq \frac{n/2 - 1}{3/4}\).

Since \(|E(H_s)| = \frac{n}{2} - \frac{n}{2} - \frac{n}{3} \geq n/12\),

While it is possible to use the Cayley graph construction of Ramanujan graphs to obtain a better bound, we did not attempt to optimize those constants in this work. Then we finish the proof of Claim 12.

Proof of Claim 12. Recall that \(H'_s\) and \(H'_{t}\) are subgraphs in \(G'\) obtained by removing \((u, u')\) in \(H_s\) and \((v, v')\) in \(H_t\) separately. We show that the 2nd eigenvalue of \(L_{H'_s}\) has \(\lambda_2(L_{H'_s}) \geq \frac{3 - 2\sqrt{2}}{4}\). As a Ramanujan graph \(H_s\), \(\lambda_2(L_{H_s}) \geq 3 - 2\sqrt{2}\). Then we apply Lemma 4 to \(\lambda_2: \) since \(R_{H'_s}(u, u') = R_{G}(u, u') \leq 3/4\) (from the proof of Claim 11), \(\lambda_2(L_{H'_s}) \geq \lambda_2(L_{H'_t}) \cdot (1 - R_{H'_t}(u, u')) \geq \frac{3 - 2\sqrt{2}}{4}\).

As removing other edges doesn’t decrease the effective resistance of \((s, t)\), we ignore \((u', v')\) to give a upper bound of \(R_{G}(s, t)\). Consider another path \(s - u - v - t\), we have

\[R_{H'_s}(s, u) = (1_s - 1_u) \cdot L_{H'_s}(1_s - 1_u) = \frac{1}{\lambda_2(L_{H'_s})} \cdot \frac{1}{\lambda_2(L_{H'_s})} = \frac{4}{3 - 2\sqrt{2}} \cdot 2 = 24 + 16\sqrt{2};\]

and the same bound holds for \(R_{H'_t}(v, t)\). Since \((s, t)\) and the path passing by \(s - u - v - t\) are in parallel, we have \(R_{G}(s, t) \leq \frac{1}{\max(R_{H'_s}(s, u) + R_{H'_t}(v, t))} \leq 0.99\).

More formally, \(1/R_{G}(s, t) = \min_{\phi \in R^{V}: \phi(s) = 1, \phi(t) = 0} \sum_{(a, b) \in E} (\phi(a) - \phi(b))^2\) from Property 4 of Lemma 7. Since \(\phi(s)\) and \(\phi(t)\) are fixed,

\[\min_{\phi \in R^{V}: \phi(s) = 1, \phi(t) = 0} \sum_{(a, b) \in E} (\phi(a) - \phi(b))^2 = 1 + \min_{\phi \in R^{V}: \phi(s) = 1, \phi(t) = 0} \sum_{(a, b) \in E(G') \setminus (s, t)} (\phi(a) - \phi(b))^2.\]

Because there is only one path \(s - u - v - t\) between \(s\) and \(t\) in \(G'\) if we ignore the two edges \((s, t)\) and \((u', v')\), we simplify the 2nd term as follows.

\[
\min_{\phi \in R^{V}: \phi(s) = 1, \phi(t) = 0} \sum_{(a, b) \in E(G') \setminus (s, t)} (\phi(a) - \phi(b))^2
= \min_{\phi(u), \phi(v) \in R} \left[ (\phi(u) - \phi(v))^2 + \min_{\phi(s) = 1} \sum_{(a, b) \in E(H'_s)} (\phi(a) - \phi(b))^2 + \min_{\phi(t) = 0} \sum_{(a, b) \in E(H'_t)} (\phi(a) - \phi(b))^2 \right]
= \min_{\phi(u), \phi(v) \in R} \left[ (\phi(u) - \phi(v))^2 + \frac{(1 - \phi(u))^2}{R_{H'_s}(s, u)} + \frac{\phi(v)^2}{R_{H'_t}(v, t)} \right]
= \frac{1}{1 + R_{H'_s}(s, u) + R_{H'_t}(v, t)}
\]

where the second equation follows from Property 4 of Lemma 7 and the last equation holds when \(\phi^*(u) = 1 + R_{H'_s}(s, u) + R_{H'_t}(v, t)\) and \(\phi^*(v) = 1 + R_{H'_s}(s, u) + R_{H'_t}(v, t)\).

\[\square\]

5 Approximation Algorithm for Degree 2

In Section 4, we proved a lower bound for degree-3 graphs (except \(s\) and \(t\)). Now we give an approximation algorithm to solve the case of degree 2 in sublinear time. Since all vertices are of degree 2 except \(s\) and \(t\), essentially, \(G\) is constituted by several disjoint paths (some of them are disconnected) between \(s\) and \(t\).
Theorem 13. Given any $\epsilon < 0.1$, for any graph with degree at most 2 except the adjacent pair $s$ and $t$, there is a local algorithm such that with probability 0.99, it outputs a $(1 + \epsilon)$-approximation of $R_G(s, t)$ in time $O\left(\min\{d(s)^2, d(t)^2\} \cdot \log^2 n \cdot \log \log n \cdot \epsilon^{-2}\right)$.

Note that as long as $\min\{d(s), d(t)\}$ is not too large, the running time in the above theorem is sublinear for any constant $\epsilon > 0$. In the rest of this section, we finish the proof of Theorem 13. In fact, our algorithm could output an additive-error approximation $\tilde{R}_G(s, t)$ in time $O\left(\min\{d(s), d(t)\} \cdot \log^2 n \cdot \log \log n \cdot \delta^{-1} \cdot \epsilon^{-1}\right)$ such that $R_G(s, t) = (1 \pm \epsilon) \cdot \tilde{R}_G(s, t) + \delta$. Since we can start from either $s$ or $t$, the expected time is sublinear for any constant $\epsilon > 0$. In Algorithm 1, we describe the additive-error approximation algorithm motivated by the chaining argument whose idea is to sample longer paths with smaller probability. Its guarantee is

$$\hat{\rho} = (1 \pm \epsilon) \cdot \rho \pm \delta$$

for the output $\hat{\rho} = 1/\tilde{R}_G(s, t)$ and $\rho = 1/R_G(s, t)$,

which gives the additive error approximation $\tilde{R}_G(s, t) = (1 \pm 2\epsilon) \cdot R_G(s, t) \pm 2\delta$ by Claim 14.

Algorithm 1 Algorithm for estimating effective resistance with additive error.

1: function ADDITIVEAPPROXIMATEEffectiveResistance($G, \epsilon, \delta, s, t$)
2: If $d(s) > d(t)$ then swap $s$ and $t$
3: $\tilde{\rho} \leftarrow 0$, $p_0 \leftarrow 1$
4: $a \leftarrow 20 \log n \cdot \log \log n$
5: for $k$ in $[0, \ldots, \log n]$ do
6:   for each path $i$ from $s$ do
7:     With prob. $p_k$, take at most $2^{k+1} \cdot a$ steps in path $i$ to find out its length $\ell_i$
8:     if it reaches $t$ and $\ell_i > 2^k \cdot a$, then $\tilde{\rho} \leftarrow \tilde{\rho} + 1/(p_k \cdot \ell_i)$
9:   $p_k \leftarrow p_{k-1}/2$
10: return $1/\tilde{\rho}$ as the effective resistance (and $\tilde{\rho}$ as the conductance)

In the rest of this section, we prove the correctness of Algorithm 1. We remark that $p_k = 2^{-k}$ in this section. Let us consider its expected time. For a path of length $\ell_i$, if $j = \lfloor \log_2 \ell_i/a \rfloor$, the number of expected steps on this path is at most

$$p_0 \cdot 2a + p_1 \cdot 4a + \cdots + p_j \cdot 2^ja + p_{j+1} \cdot \ell_i + \cdots + p_{\log n} \cdot \ell_i = 2a \cdot j + \ell_i \cdot (p_{j+1} + \cdots + p_{\log n}) = O(a \cdot \log n).$$

Since we can start from either $s$ or $t$, the expected time is $O(\min\{d(s), d(t)\} \cdot a \cdot \log n)$. 
Next we show the approximation guarantee in (2). For all paths whose lengths are at most 2a, the algorithm gets the exact conductance \(1/\ell_i\) of them. The rest paths are divided into \(\log n\) groups \(G_1, \ldots, G_{\log n}\). The paths in group \(G_k\) have lengths between \((2^k \cdot a, 2^{k+1} \cdot a)\). Assuming path \(i\) is in group \(k\), we define the random variable \(X_i\) for estimating \(1/\ell_i\). With probability \(pk\), \(X_i = \frac{1}{pk\ell_i} - \frac{1}{\ell_i}\); otherwise, \(X_i = -\frac{1}{\ell_i}\). So \(\mathbb{E} X_i = 0\) and \(\text{Var} X_i = \frac{1}{pk \cdot \ell_i^2} \leq \frac{1}{pk \cdot \ell_i^2}\). And the result of Algorithm 1 is \(\hat{\rho} = \sum 1/\ell_i + X_i = \rho + \sum X_i\).

Now, we calculate the approximation of group \(G_k\) by Bernstein’s inequality. There are two cases of \(G_k\), i.e., \(\sum \frac{1}{G_k \ell_i} > b\) and \(\sum \frac{1}{G_k \ell_i} \leq b\) for a threshold value \(b = \frac{\delta}{e \cdot \log n}\).

\section*{Lemma 15 (Bernstein’s inequality [6])}

Let \(X_1, \cdots, X_n\) be independent zero-mean random variables. Suppose that \(|X_i| \leq M\) almost surely, for all \(i\). Then, for all positive \(t\),

\[
\Pr \left[ \sum_{i=1}^{n} X_i \geq t \right] \leq 2 \exp \left[ -\frac{1}{4} \min \left\{ \frac{t^2}{\sum_{i=1}^{n} \mathbb{E} X_i^2}, \frac{t}{M} \right\} \right].
\]

In the previous case, \(\sum \frac{1}{G_k \ell_i} \geq \frac{|G_k| \cdot \min \{G_k, \ell_i\}^2}{|G_k| \cdot \max \{G_k, \ell_i\}^2} \geq |G_k|/4 \geq 2^{k-2} \cdot ab\). The first inequality comes from \(\max \{G_k, \ell_i\} < 2 \min \{G_k, \ell_i\}\). And the last step is implied by \(b < \sum \frac{1}{G_k \ell_i} < |G_k|/(2^k \cdot a)\) since each \(\ell_i > 2^k \cdot a\) in \(G_k\).

Then \(\sum \mathbb{E} X_i^2 \leq \sum \frac{1}{G_k \ell_i} \leq \frac{4}{ab} \left( \sum \frac{1}{G_k \ell_i} \right)^2\), and \(|X_i| \leq \frac{2^{k-1}}{2^k - a} \leq \frac{1}{a}\) for all \(i \in G_k\). From the above discussion, Bernstein’s inequality infers that for \(t = \epsilon \sum \frac{1}{G_k \ell_i}\),

\[
\Pr \left[ \sum_{G_k} X_i \geq \epsilon \sum \frac{1}{G_k \ell_i} \right] \leq 2 \exp \left[ -\frac{1}{4} \min \left\{ \frac{a \epsilon^2}{4}, \frac{a \epsilon}{\log n} \right\} \right].
\]

The second step is by the assumption \(\sum \frac{1}{G_k \ell_i} > b\).

In the latter case, \(\sum \mathbb{E} X_i^2 \leq \sum \frac{1}{G_k \ell_i} \leq \sum \frac{1}{G_k} \cdot \ell_i \leq \frac{b}{a}\) with \(\ell_i \geq 2^k \cdot a\) and \(\sum \frac{1}{G_k \ell_i} \leq b\). So

\[
\Pr \left[ \sum_{G_k} X_i \geq \frac{\delta}{\log n} \right] \leq 2 \exp \left[ -\frac{1}{4} \min \left\{ \frac{a \delta^2}{4 \log^2 n}, \frac{a \delta}{\log n} \right\} \right].
\]

Take a union bound for all \(G_k\) with \(b = \frac{\delta}{e \log n}\), \(\Pr \left[ \sum_{G_k} X_i \geq \epsilon \cdot \sum \frac{1}{G_k \ell_i} + \delta \right] \) is at most \(2 \log n \cdot \exp \left[ -\frac{1}{4} \min \left\{ \frac{a \epsilon \delta}{4 \log^2 n}, \frac{a \delta}{\log n} \right\} \right] + 2 \log n \cdot \exp \left[ -\frac{1}{4} \min \left\{ \frac{a \epsilon \delta}{4 \log^2 n}, \frac{a \delta}{\log n} \right\} \right] \). When \(a = \frac{2 \log n \cdot \log \log n}{\epsilon \delta}\), \(\tilde{\rho}\) is a \((1 + \epsilon, \delta)\)-approximation of \(\rho\) with probability 0.99.

\section*{References}


Effective Resistances in Non-Expander Graphs


