

Finding Small Sparse Cuts by Random Walk

Tsz Chiu Kwok, Lap Chi Lau

The Chinese University of Hong Kong

Abstract. We study the problem of finding a small sparse cut in an undirected graph. Given an undirected graph $G = (V, E)$ and a parameter $k \leq |E|$, the small sparsest cut problem is to find a set $S \subseteq V$ with minimum conductance among all sets with volume at most k . Using ideas developed in local graph partitioning algorithms, we obtain the following bicriteria approximation algorithms for the small sparsest cut problem:

- If there is a set $U \subseteq V$ with conductance ϕ and $\text{vol}(U) \leq k$, then there is a polynomial time algorithm to find a set S with conductance $O(\sqrt{\phi/\epsilon})$ and $\text{vol}(S) \leq k^{1+\epsilon}$ for any $\epsilon > 1/k$.
- If there is a set $U \subseteq V$ with conductance ϕ and $\text{vol}(U) \leq k$, then there is a polynomial time algorithm to find a set S with conductance $O(\sqrt{\phi \log k/\epsilon})$ and $\text{vol}(S) \leq (1 + \epsilon)k$ for any $\epsilon > 2 \log k/k$.

These algorithms can be implemented locally using truncated random walk, with running time almost linear to k .

1 Introduction

For an undirected graph $G = (V, E)$, the conductance of a set $S \subseteq V$ is defined as $\phi(S) = |\delta(S)|/\text{vol}(S)$, where $\delta(S)$ is the set of edges with one endpoint in S and another endpoint in $V - S$, and $\text{vol}(S) = \sum_{v \in S} d(v)$ where $d(v)$ is the degree of v in G . Let $n = |V|$ and $m = |E|$. The conductance of G is defined as $\phi(G) = \min_{S: \text{vol}(S) \leq m} \phi(S)$. The conductance of a graph is an important parameter that is closely related to the expansion of a graph and the mixing time of a random walk [9]. Finding a set of small conductance, called a sparse cut, is a well-studied algorithmic problem that has applications in different areas. Several approximation algorithms are known for the sparsest cut problem. The spectral partitioning algorithm by Cheeger's inequality [7, 1] finds a set of conductance $\sqrt{\phi(G)}$ with volume at most m . The linear programming rounding algorithm by Leighton and Rao [11] finds a set of conductance $O(\phi(G) \log(n))$ with volume at most m . The semidefinite programming rounding algorithm by Arora, Rao and Vazirani [5] finds a set of conductance $O(\phi(G) \sqrt{\log(n)})$ with volume at most m .

Recently there has been much interest in studying the small sparsest cut problem, to determine $\phi_k(G) = \min_{S: \text{vol}(S) \leq k} \phi(S)$ for a given k , and to find a set of smallest conductance among all sets of volume at most k . This is also known as the expansion profile of the graph [13, 16]. There are two main motivations for this problem. One is the small set expansion conjecture [15], which states that for

every constant $\epsilon > 0$ there exists a constant $\delta > 0$ such that it is NP-hard to distinguish whether $\phi_{\delta m}(G) \leq \epsilon$ or $\phi_{\delta m}(G) \geq 1 - \epsilon$. This conjecture is shown to be closely related to the unique games conjecture [15], and so it is of interest to understand what algorithmic techniques can be used to estimate $\phi_k(G)$. There are bicriteria approximation algorithms for this problem using semidefinite programming relaxations: Raghavendra, Steurer and Tetali [16] obtained an algorithm that finds a set S with $\text{vol}(S) \leq O(k)$ and $\phi(S) \leq O(\sqrt{\phi_k(G) \log(m/k)})$, and Bansal et.al. [6] obtained an algorithm that finds a set S with $\text{vol}(S) \leq (1 + \epsilon)k$ and $\phi(S) \leq O(f(\epsilon) \phi_k(G) \sqrt{\log n \log(m/k)})$ for any $\epsilon > 0$ where $f(\epsilon)$ is a function depends only on ϵ .

There are also algorithms for finding a small sparse cut using the eigenvalues and eigenvectors of the Laplacian matrix: Arora, Barak and Steurer [4] gave a random walk based algorithm that returns a set of conductance $O(\sqrt{\lambda_{n^{100\epsilon}}}/\epsilon)$ with size $O(n^{1-\epsilon})$ for $0 < \epsilon < 1$, where λ_i is the i -th smallest eigenvalue of the Laplacian matrix of the graph. Lee, Oveis Gharan and Trevisan [10] and Louis, Raghavendra, Tetali and Vempala [12] gave spectral algorithms that return a set of conductance $O(\sqrt{\lambda_{n/k} \log(n/k)})$ with size $O(k)$. We note that these results give sufficient conditions to find a small sparse cut efficiently, but they do not imply a bicriteria approximation algorithm for the small sparsest cut problem. Recently there is also a random walk based algorithm for finding a balanced separator with conductance $O(\sqrt{\phi})$ in nearly linear time [19] where ϕ is the conductance of the optimal balanced separator, but this result does not apply directly for the small sparsest cut problem.

Another motivation is the design of local graph partitioning algorithms in massive graphs. In some situations, we have a massive graph $G = (V, E)$ and a vertex $v \in V$, and we would like to identify a small set S with small conductance that contains v (if it exists). The graph may be too big that it is not feasible to read the whole graph and run some nontrivial approximation algorithms. So it would be desirable to have a local algorithm that only explores a small part of the graph, and outputs a set S with small conductance that contains v , and the running time of the algorithm depends only on $\text{vol}(S)$ and $\text{polylog}(n)$. All local graph partitioning algorithms are based on some random walk type processes. The efficiency of the algorithm is measured by the work/volume ratio, which is defined as the ratio of the running time and the volume of the output set. Spielman and Teng [18] proposed the first local graph partitioning algorithm using truncated random walk, that returns a set S' with $\phi(S') = O(\phi^{1/2}(S) \log^{3/2} n)$ if the initial vertex is a random vertex in S , and the work/volume ratio of the algorithm is $O(\phi^{-2}(S) \text{polylog}(n))$. Anderson, Chung, Lang [2] used local pagerank vectors to find a set S' with $\phi(S') = O(\sqrt{\phi(S) \log k})$ and work/volume ratio $O(\phi^{-1}(S) \text{polylog}(n))$, if the initial vertex is a random vertex in a set S with $\text{vol}(S) = k$. Anderson and Peres [3] used the volume-biased evolving set process to obtain a local graph partitioning algorithm with work/volume ratio $O(\phi^{-1/2} \text{polylog}(n))$ and a similar conductance guarantee as in [2]. Note that the running time of these algorithms would be sublinear if the volume of the output set is small, which is the case of interest in massive graphs.

1.1 Main Results

We show that the techniques developed in local graph partitioning algorithms [18, 8] can be used to obtain bicriteria approximation algorithms for the small sparsest cut problem. We note that the algorithm in Theorem 1 is the same as the algorithm of Arora, Barak and Steurer [4], but we adapt the analysis in local graph partitioning algorithms to prove a tradeoff between the conductance guarantee and the volume of the output set.

Theorem 1. *Given an undirected graph $G = (V, E)$ and a parameter k , there is a polynomial time algorithm to do the following:*

1. Find a set S with $\phi(S) = O(\sqrt{\phi_k(G)/\epsilon})$ and $\text{vol}(S) \leq k^{1+\epsilon}$ for any $\epsilon > 1/k$.
2. Find a set S with $\phi(S) = O(\sqrt{\phi_k(G) \log k/\epsilon})$ and $\text{vol}(S) \leq (1 + \epsilon)k$ for any $\epsilon > 2 \log k/k$.

For the small sparsest cut problem, when k is sublinear ($k = O(m^c)$ for $c < 1$), the performance guarantee of the bicriteria approximation algorithm in Theorem 1(2) is similar to that of Raghavendra, Steurer and Tetali [16]. Also, when k is sublinear, the conductance guarantee of Theorem 1(1) is independent of n , which matches the performance of spectral partitioning while having a bound on the volume of the output set. These show that random walk algorithms can also be used to give nontrivial bicriteria approximations for the small sparsest cut problem. We note that the result of Anderson and Peres [3] implies a similar statement to Theorem 1(2), with the same conductance guarantee and $\text{vol}(S) = O(k)$. The algorithms in Theorem 1 can also be implemented locally by using the truncated random walk algorithm.

Theorem 2. *For an undirected graph $G = (V, E)$ and a set $U \subseteq V$, given $\phi \geq \phi(U)$ and $k \geq \text{vol}(U)$, there exists an initial vertex such that the truncated random walk algorithm can find a set S with $\phi(S) \leq O(\sqrt{\phi/\epsilon})$ and $\text{vol}(S) \leq O(k^{1+\epsilon})$ for any $\epsilon > 2/k$. The runtime of the algorithm is $\tilde{O}(k^{1+2\epsilon}\phi^{-2})$.*

When k is sublinear, the conductance guarantee of Theorem 2 matches that of spectral partitioning, improving on the conductance guarantees in previous local graph partitioning algorithms. However, we note that our notion of a local graph partitioning algorithm is much weaker than previous work [18, 2, 3], as they proved that a random initial vertex u will work with a constant probability, while we only prove that there exists an initial vertex that will work and unable to prove the high probability statement.

In Section 4 we discuss a connection to the small set expansion conjecture.

Independent Work Oveis Gharan and Trevisan [20] proved Theorem 1 independently. They also proved a stronger version of Theorem 2, with a faster running time ($\tilde{O}(k^{1+2\epsilon}\phi^{-1/2})$) and also the algorithm works for a random initial vertex in S with constant probability. We note that their result implies that our truncated random walk algorithm will also succeed with constant probability if we start from a random initial vertex in S .

1.2 Techniques

The techniques are from the work of Spielman and Teng [18] and Chung [8]. Our goal in Theorem 1(1) is equivalent to distinguish the following two cases: (a) there is a set S with $\text{vol}(S) \leq k$ and $\phi(S) \leq \varphi$, or (b) the conductance of every set of volume at most ck is at least $\Omega(\sqrt{\varphi})$ for some $c > 1$. As in [18], we use the method of Lovász and Simonovits [14] that considers the total probability of the k edges with largest probability after t steps of random walk, call this number $C_t(k)$. In case (a), we use the idea of Chung [8] that uses the local eigenvector of S of the Laplacian matrix to show that there exists an initial vertex such that $C_t(k) \geq (1 - \frac{\varphi}{2})^t$. In case (b), we use a result of Lovász and Simonovits [14] to show that $C_t(k) \leq \frac{1}{c} + \sqrt{k}(1 - M\varphi)^t$ for a large enough constant M , no matter what is the initial vertex of the random walk. Hence, say when $c \geq k^{0.01}$, by setting $t = \Theta(\log k/\varphi)$, we expect that $C_t(k)$ is significantly greater than $1/c$ in case (a) but at most $1/c$ plus a negligible term in case (b), and so we can distinguish the two cases. To prove Theorem 2(1), we use the truncated random walk algorithm as in [18] to give a bound on the runtime. Theorem 1(2) is a corollary of Theorem 1(1).

2 Finding Small Sparse Cuts

The organization of this section is as follows. First we review some basics about random walk in undirected graphs. Then we present our algorithm in Theorem 1 and the proof outline, and then we present the analysis and complete the proof of Theorem 1.

2.1 Random Walk

In the following we assume $G = (V, E)$ is a simple unweighted undirected connected graph with $n = |V|$ vertices and $m = |E|$ edges. Our algorithms are based on random walk. Let p_0 be an initial probability distribution on vertices. Let A be the adjacency matrix of G , D be the diagonal degree matrix of G , and $W = \frac{1}{2}(I + D^{-1}A)$ be the lazy random walk matrix. The probability distribution after t steps of lazy random walk is defined as $p_t = p_0 W^t$. (For convenience, we use p_t to denote a row vector, while all other vectors by default are column vectors.) For a subset $S \subseteq V$, we use $p_t(S)$ to denote $\sum_{u \in S} p_t(u)$.

To analyze the probability distribution after t steps of lazy random walk, we use the method developed by Lovász and Simonovits [14] as in other local graph partitioning algorithms [18, 2]. We view the graph as directed by replacing each undirected edge with two directed edges with opposite directions. Given a probability distribution p on vertices, each directed edge $e = uv$ is assigned probability $q(e) = p(u)/d_u$. Let e_1, e_2, \dots, e_{2m} be an ordering of the directed edges such that $q(e_1) \geq q(e_2) \geq \dots \geq q(e_{2m})$. The curve introduced by Lovász and Simonovits $C : [0, 2m] \rightarrow [0, 1]$ is defined as follows: for integral x , $C(x) = \sum_{i=1}^x q(e_i)$; for fractional $x = \lfloor x \rfloor + r$, $C(x) = (1-r)C(\lfloor x \rfloor) + rC(\lceil x \rceil)$. Let C_t be

the curve when the underlying distribution is p_t . Let v_1, v_2, \dots, v_n be an ordering of the vertices such that $p_t(v_1)/d(v_1) \geq p_t(v_2)/d(v_2) \geq \dots \geq p_t(v_n)/d(v_n)$. Then $C_t(\sum_{i=1}^j d(v_i)) = \sum_{i=1}^j p_t(v_i)$ for all $j \in [n]$. We call the points $x_j = \sum_{i=1}^j d(v_i)$ extreme points, and note that the curve is linear between two extreme points. We also call the sets $S_{t,j} = \{v_1, \dots, v_j\}$ for $1 \leq j \leq n$ the level sets at time t .

The curve C_t is concave, and it approaches the straight line $x/(2m)$ when p_t approaches the stationary distribution. Lovász and Simonovits [14] analyzed the convergence rate of this curve to the straight line based on the conductances of the level sets.

Lemma 1 (Lovász-Simonovits [14]). *Let $x = x_j \leq m$ be an extreme point at time t and $S = S_{t,j}$ be the corresponding level set. If $\phi(S) \geq \varphi$, then $C_t(x) \leq \frac{1}{2}(C_{t-1}(x - \varphi x) + C_{t-1}(x + \varphi x))$.*

2.2 Algorithm

Our algorithm is simple and is the same as in Arora, Barak and Steurer [4]. For each vertex v , we use it as the initial vertex of the random walk, and compute the probability distributions p_t for $1 \leq t \leq O(n^2 \log n)$. Then we output the set of smallest conductance among all level sets $S_{t,j}$ (of all initial vertices) of volume at most ck , where in Theorem 1(1) we set $c = k^\epsilon$ and in Theorem 1(2) we set $c = 1 + \epsilon$. Clearly this is a polynomial time algorithm.

To analyze the performance of the algorithm, we give upper and lower bound on the curve based on the conductances. On one hand, we use Lemma 1 to prove that if all level sets of volume at most ck are of conductance at least ϕ_1 , then the curve satisfies $C_t(x) \leq f_t(x) := \frac{x}{ck} + \sqrt{x}(1 - \frac{\phi_1^2}{8})^t$ for all $x \leq k$. Informally, this says that if ϕ_1 is large, then $C_t(k)$ is at most $1/c$ plus a negligible term when t is large enough. This statement holds regardless of the initial vertex of the random walk. On the other hand, if there exists a set S of volume at most k with conductance ϕ_2 , then we use the idea of Chung [8] that uses the local eigenvector of S of the Laplacian matrix to show that there exists an initial vertex for which $C_t(k) \geq (1 - \frac{\phi_2}{2})^t$. Informally, this says that if ϕ_2 is small, then $C_t(k)$ is significantly larger than $1/c$ if c is large. Finally, by combining the upper and lower bound for $C_t(k)$ and choosing an appropriate t , we show that $\phi_1 \leq O(\sqrt{\phi_2})$ when $c = k^\epsilon$ and $\phi_1 \leq O(\sqrt{\phi_2 \log k})$ when $c = 1 + \epsilon$. Hence the algorithm can find a level set with the required conductance.

2.3 Upper Bound

We prove the upper bound using Lemma 1. We note that the following statement is true for any initial probability distribution, in particular when $p_0 = \chi_v$ for any v .

Theorem 3. *Suppose for all $t' \leq t$ and $i \in [n]$, we have $\phi(S_{t',i}) \geq \phi_1$ whenever $\text{vol}(S_{t',i}) \leq l \leq m$. Then the curve satisfies $C_t(x) \leq f_t(x) := \frac{x}{l} + \sqrt{x}(1 - \frac{\phi_1^2}{8})^t$ for all $x \leq 2m$.*

Proof. Let the extreme points x_i satisfy $0 = x_0 \leq x_1 \leq x_2 \leq \dots \leq x_i \leq l < x_{i+1}$. Note that C_t is linear between extreme points and between x_i and l , and f_t is concave. So we only need to show the inequality for extreme points and for $x \geq l$. When $x \geq l$, the inequality always hold as $f_t(x) \geq 1 \geq C_t(x)$ for any t . Now we would prove by induction. When $t = 0$ the inequality is trivial as $f_0(x) \geq 1 \geq C_0(x)$ for all $x \geq 1$. When $t > 0$ and x is an extreme point,

$$\begin{aligned} C_t(x) &\leq \frac{1}{2}(C_{t-1}(x - \phi_1 x) + C_{t-1}(x + \phi_1 x)) \quad (\text{by Lemma 1}) \\ &\leq \frac{1}{2}(f_{t-1}(x - \phi_1 x) + f_{t-1}(x + \phi_1 x)) \quad (\text{by induction}) \\ &= \frac{x}{l} + \frac{1}{2}\sqrt{x}\left(1 - \frac{\phi_1^2}{8}\right)^{t-1}(\sqrt{1 - \phi_1} + \sqrt{1 + \phi_1}) \\ &\leq \frac{x}{l} + \sqrt{x}\left(1 - \frac{\phi_1^2}{8}\right)^t, \end{aligned}$$

where the last inequality follows from Taylor expansions of $\sqrt{1 - \phi_1}$ and $\sqrt{1 + \phi_1}$.

2.4 Lower Bound

The idea is to use the local eigenvector of S of the normalized Laplacian matrix to show that there is an initial distribution such that $p_t(S) \geq (1 - \frac{\phi_2}{2})^t$.

Theorem 4. *Assume $S \subseteq V$ where $\text{vol}(S) \leq m$ and $\phi(S) \leq \phi_2$. Then there exists a vertex v such that if $p_0 = \chi_v$, then $p_t(S) \geq (1 - \frac{\phi_2}{2})^t$.*

Proof. Let $\mathcal{L} = D^{-\frac{1}{2}}LD^{-\frac{1}{2}}$ be the normalized Laplacian matrix, where $L = D - A$ is the Laplacian matrix of the graph. For any matrix M with rows and columns indexed by V , let M_S be the $|S| \times |S|$ submatrix of M with rows and columns indexed by the vertices in S . Consider the smallest eigenvalue λ_S of \mathcal{L}_S and its corresponding eigenvector v_S . Let χ_S be the characteristic vector of S . We have

$$(D_S^{\frac{1}{2}}\mathbf{1})^T \mathcal{L}_S (D_S^{\frac{1}{2}}\mathbf{1}) = \mathbf{1}^T L_S \mathbf{1} = \sum_{e=uv \in E} (\chi_S(u) - \chi_S(v))^2 = |\delta(S)|.$$

So, by the Courant-Fischer theorem,

$$\lambda_S \leq \frac{(D_S^{\frac{1}{2}}\mathbf{1})^T \mathcal{L}_S (D_S^{\frac{1}{2}}\mathbf{1})}{\|D_S^{\frac{1}{2}}\mathbf{1}\|_2^2} = \frac{|\delta(S)|}{\text{vol}(S)} \leq \phi_2.$$

We assume without loss of generality that S is a connected subgraph. Then, by the Perron-Frobenius theorem, the eigenvector v_S can be assumed to be positive, and we can rescale v_S such that $D_S^{\frac{1}{2}}v_S$ is a probability distribution. Let $p_{t,S}$ denote the restriction of p_t on S . We set the initial distribution p_0 such that

$p_{0,S} = (D_S^{\frac{1}{2}} v_S)^T$, and $p_{0,V-S} = 0$. We would show that $p_{t,S} \geq (1 - \frac{\lambda_S}{2})^t p_{0,S}$ by induction. Clearly the statement is true when $t = 0$. For $t > 0$, we have

$$\begin{aligned}
p_{t,S} &\geq p_{t-1,S} W_S \\
&= p_{t-1,S} \cdot \frac{(I_S + D_S^{-1} A_S)}{2} \\
&\geq (1 - \frac{\lambda_S}{2})^{t-1} v_S^T D_S^{\frac{1}{2}} \frac{(I_S + D_S^{-1} A_S)}{2} \quad (\text{by induction}) \\
&= (1 - \frac{\lambda_S}{2})^{t-1} v_S^T (I - \frac{\mathcal{L}_S}{2}) D_S^{\frac{1}{2}} \\
&= (1 - \frac{\lambda_S}{2})^t v_S^T D_S^{\frac{1}{2}} \\
&= (1 - \frac{\lambda_S}{2})^t p_{0,S}.
\end{aligned}$$

Therefore,

$$p_t(S) = p_{t,S}(S) \geq (1 - \frac{\lambda_S}{2})^t p_{0,S}(S) \geq (1 - \frac{\phi_2}{2})^t.$$

Since random walk is linear and v_S is a convex combination of χ_v where $v \in S$, there exists a vertex $v \in S$ such that if $p_0 = \chi_v$, then $p_t(S) \geq (1 - \frac{\phi_2}{2})^t$.

2.5 Proof of Theorem 1

We combine the upper bound and the lower bound to prove Theorem 1. We note that Theorem 1 is trivial if $\phi_k(G) \geq \epsilon$, and so we assume $\phi_k(G) < \epsilon$. We also assume $\epsilon \leq 0.01$, as otherwise we reset $\epsilon = 0.01$ and lose only a constant factor.

The algorithm is simple. Set $T = \epsilon k^2 \log k / 4$. For each vertex u , set $p_0 = \chi_u$ and compute $S_{t,i}$ for all $t \leq T$ and $i \in [n]$. Denote these sets by $S_{t,i,u}$ to specify the starting vertex u . Output a set $S = S_{t,i,u}$ that achieves the minimum in $\min_{\text{vol}(S_{t,i,u}) \leq k^{1+\epsilon}} \phi(S_{t,i,u})$. Clearly the algorithm runs in polynomial time.

We claim that $\phi(S) \leq 4\sqrt{\phi_k(G)/\epsilon}$. Suppose to the contrary that the algorithm does not return such a set. Consider $t = \frac{\epsilon \log k}{2\phi_k(G)}$; note that $t \leq T$ as $\phi_k(G) \geq 1/k^2$ for a simple unweighted graph. Applying Theorem 3 with $l = k^{1+\epsilon}$, for any starting vertex u , we have

$$\begin{aligned}
C_t(k) &\leq \frac{k}{k^{1+\epsilon}} + \sqrt{k} (1 - 2\frac{\phi_k(G)}{\epsilon})^t \\
&\leq k^{-\epsilon} + \sqrt{k} \exp(-2\frac{\phi_k(G)}{\epsilon} \frac{\epsilon \log k}{2\phi_k(G)}) \\
&= k^{-\epsilon} + \sqrt{k} \exp(-\log k) \\
&= k^{-\epsilon} + k^{-\frac{1}{2}}.
\end{aligned}$$

On the other hand, suppose S^* is a set with $\text{vol}(S^*) \leq k$ and $\phi(S^*) = \phi_k(G)$. Then Theorem 4 says that there exists a starting vertex $u^* \in S^*$ such that

$$p_t(S^*) \geq (1 - \frac{\phi_k(G)}{2})^t$$

$$\begin{aligned}
&\geq \exp(-\phi_k(G)t) \quad (\text{for } \phi_k(G) < 0.01) \\
&= \exp\left(-\frac{1}{2}\epsilon \log k\right) \\
&= k^{-\frac{\epsilon}{2}} \\
&> k^{-\epsilon} + k^{-\frac{1}{2}} \quad (\text{for } k \geq \frac{1}{\epsilon} \text{ and } \epsilon \leq 0.01)
\end{aligned}$$

This is contradicting since $C_t(k) \geq p_t(S^*)$ for that starting vertex, completing the proof of Theorem 1(1).

Now we obtain Theorem 1(2) as a corollary of Theorem 1(1). Set $\epsilon' = \frac{\epsilon}{2 \log k}$. Then $k^{1+\epsilon'} \leq (1+\epsilon)k$. By using Theorem 1(1) with ϵ' , we have Theorem 1(2).

3 Local Graph Partitioning

To implement the algorithm locally, we use truncated random walk as in [18]. Let $q_0 = \chi_v$. For each $t \geq 0$, we define \tilde{p}_t by setting $\tilde{p}_t(v) = 0$ if $q_t(v) < \epsilon d(v)$ and setting $\tilde{p}_t(v) = q_t(v)$ if $q_t(v) \geq \epsilon d(v)$, and we define $q_{t+1} = \tilde{p}_t W$. Then, we just use \tilde{p}_t to replace p_t in the algorithm in Section 2. To prove that the truncated random walk algorithm works, we first show that \tilde{p}_t is a good approximation of p_t and can be computed locally. Then we show that the curve defined by \tilde{p}_t satisfies the upper bound in Theorem 3, and it almost satisfies the lower bound in Theorem 4. Finally we combine the upper bound and the lower bound to prove Theorem 2.

3.1 Computing Truncated Distributions

Lemma 2. *There is an algorithm that compute \tilde{p}_t such that $\tilde{p}_t \leq p_t \leq \tilde{p}_t(v) + \epsilon t d$ for every $0 \leq t \leq T$, with time complexity $O(T/\epsilon)$, where d is the degree vector.*

Proof. First we prove the approximation guarantee. By induction, we have the upper bound

$$\tilde{p}_t \leq q_t = \tilde{p}_{t-1} W \leq p_{t-1} W = p_t.$$

Also, by induction, we have the lower bound

$$p_t = p_{t-1} W \leq (\tilde{p}_{t-1} + \epsilon(t-1)d)W = q_t + \epsilon(t-1)d \leq \tilde{p}_t + \epsilon t d.$$

Next we bound the computation time. Let S_t be the support of \tilde{p}_t . In order to compute q_{t+1} from \tilde{p}_t , we need to update each vertex $v \in S_t$ and its neighbors. Using a perfect hash function, the neighbors of a vertex v can be updated in $O(d(v))$ steps, and thus q_{t+1} and \tilde{p}_{t+1} can be computed in $O(\text{vol}(S_t))$ steps. Since each vertex $v \in S_t$ satisfies $\tilde{p}_t \geq \epsilon d(v)$, we have $\text{vol}(S_t) = \sum_{v \in S_t} d(v) \leq p_t(S_t)/\epsilon \leq 1/\epsilon$, and this completes the proof.

3.2 Approximate Upper Bound

We use the truncated probability distributions to define the curve \tilde{C}_t . Note that \tilde{p}_t may not be a probability distribution and $\tilde{C}_t(2m)$ may be less than one. And we define the level sets $\tilde{S}_{t,i} = \{v_1, v_2, \dots, v_i\}$ when we order the vertices such that $\tilde{p}_t(v_1)/d(v_1) \geq \tilde{p}_t(v_2)/d(v_2) \geq \dots \geq \tilde{p}_t(v_n)/d(v_n)$. We show that \tilde{C}_t would satisfy the same upper bound as in Theorem 3.

Lemma 3. *Suppose for all $t \leq T$ and $i \in [n]$, we have $\phi(\tilde{S}_{t,i}) \geq \phi_1$ whenever $\text{vol}(\tilde{S}_{t,i}) \leq l \leq m$. Then $\tilde{C}_t(x) \leq f_t(x) := \frac{x}{l} + \sqrt{x}(1 - \frac{\phi_1^2}{8})^t$ for all $x \leq 2m$.*

Proof. Let $\tilde{x}_i = \sum_{v \in \tilde{S}_{t,i}} d(v)$ be the extreme points defined by \tilde{p}_t . By the same proof as in Theorem 3, it suffices to prove that Lemma 1 still holds after replacing p_t by \tilde{p}_t . It means that we need to show if $x = \tilde{x}_i \leq m$ is an extreme point (at time t), $S = \tilde{S}_{t,j}$ is the corresponding set of vertices and $\text{vol}(S) \geq \phi$, then $\tilde{C}_t(x) \leq \frac{1}{2}(\tilde{C}_{t-1}(x - \phi x) + \tilde{C}_{t-1}(x + \phi x))$. This is true since the curve defined by $q_t = \tilde{p}_{t-1}W$ is less than $\frac{1}{2}(\tilde{C}_{t-1}(x - \phi x) + \tilde{C}_{t-1}(x + \phi x))$ by Lemma 1, and $\tilde{p}_t \leq q_t$.

3.3 Proof of Theorem 2

Suppose U is a subset of vertices with $\text{vol}(U) \leq k$ and $\phi(U) \leq \varphi$, where $\frac{1}{\epsilon} \leq k \leq m$. We would prove that given k and φ and an initial vertex u in U with $p_t(U) \geq \frac{1}{c}(1 - \frac{\varphi}{2})^t$ for a constant $c > 1$, the truncated random walk algorithm will output a set S with $\text{vol}(S) \leq O(k^{1+\epsilon})$ and $\phi(S) \leq 8\sqrt{\varphi/\epsilon}$. The running time of the algorithm is $O(\epsilon^2 k^{1+2\epsilon} \log^3 k / \varphi^2)$.

For concreteness we set $c = 4$ in the following calculations. Set $T = \frac{\epsilon \log k}{2\varphi}$ and $\epsilon' = \frac{k^{-1-\epsilon}}{20T}$. Applying Lemma 2 with T and ϵ' , we can compute all \tilde{p}_t and thus $\tilde{S}_{t,i}$ for all $t \leq T$ and $i \in [5k^{1+\epsilon}]$ in $O(T \log k / \epsilon') = O(\epsilon^2 k^{1+\epsilon} \log^3 k / \varphi^2)$ steps (with an additional $\log k$ factor for sorting). By Lemma 2, the starting vertex u will give $\tilde{p}_T(U) \geq \frac{1}{4}(1 - \frac{\varphi}{2})^T - \epsilon' T \text{vol}(U)$. We claim that one of the set $S = \tilde{S}_{t,i}$ must satisfy $\text{vol}(S) \leq 5k^{1+\epsilon}$ and $\phi(S) \leq 8\sqrt{\varphi/\epsilon}$. Otherwise, setting $\phi_1 \geq 8\sqrt{\varphi/\epsilon}$, we have

$$\begin{aligned} \tilde{p}_T(U) &\geq \frac{1}{4}(1 - \frac{\varphi}{2})^T - \epsilon' T \text{vol}(U) \\ &\geq \frac{1}{4} \exp(-\varphi T) - \frac{k^{-\epsilon}}{20} \quad (\text{for } \phi < 0.01) \\ &= \frac{k^{-\frac{\epsilon}{2}}}{4} - \frac{k^{-\epsilon}}{20} \\ &> \frac{k^{-\epsilon}}{5} + k^{-\frac{1}{2}} \quad (\text{using } k^{-\frac{\epsilon}{2}} > k^{-\epsilon} + 4k^{-\frac{1}{2}} \text{ for } k \geq \frac{1}{\epsilon} \text{ and } \epsilon \leq 0.01) \\ &\geq \frac{k}{5k^{1+\epsilon}} + \sqrt{k}(1 - \frac{\phi_1^2}{8})^T \\ &\geq \tilde{C}_T(k), \end{aligned}$$

which is a contradiction, completing the proof of Theorem 2.

4 Concluding Remarks

We presented a bicriteria approximation algorithm for the small sparsest cut problem with conductance guarantee independent of n , but the volume of the output set is $k^{1+\epsilon}$. We note that if one can also guarantee that the volume of the output set is at most Mk for an absolute constant M , then one can disprove the small set expansion conjecture, which states that for any constant ϵ there exists a constant δ such that distinguishing $\phi_{\delta m}(G) < \epsilon$ and $\phi_{\delta m}(G) > 1 - \epsilon$ is NP-hard. This can be viewed as an evidence that our analysis is almost tight, or an evidence that the small set expansion problem is not NP-hard. We note that this is also observed by Raghavendra, Stuerer and Tulsiani [17].

More formally, suppose there is a polynomial time algorithm with the following guarantee: given G with $\phi_k(G)$, always output a set S with $\phi(S) = f(\phi_k(G))$ and $\text{vol}(S) = Mk$ where $f(x)$ is a function that tends to zero when x tends to zero (e.g. $f(x) = x^{1/100}$) and M is an absolute constant. Then we claim that there is a (small) constant ϵ such that whenever $\phi_k(G) < \epsilon$ there is a polynomial time algorithm to return a set S with $\phi(S) < 1 - \epsilon$ and $\text{vol}(S) \leq k$.

We assume that G is a d -regular graph, as in [15] where the small set expansion conjecture was formulated. Suppose there is a subset U with $|U| = k$ and $\phi(U) < \epsilon$. First we use the algorithm to obtain a set S with $\phi(S) \leq f(\epsilon)$ and assume $|S| = Mk$ (instead of $|S| \leq Mk$). Next we show that a random subset $S' \subseteq S$ of size exactly k will have $\phi(S') < 1 - \epsilon$ with a constant probability for a small enough ϵ . Let $E(S)$ be the set of edges with both endpoints in S . Each edge in $E(S)$ has probability $2(\frac{1}{M})(1 - \frac{1}{M})$ to be in $\delta(S')$. So, the expected value of

$$|\delta(S')| \leq |\delta(S)| + 2\left(\frac{1}{M}\right)\left(1 - \frac{1}{M}\right)|E(S)|.$$

By construction $\text{vol}(S') = kd$, and so the expected value of

$$\phi(S') \leq \frac{|\delta(S)|}{kd} + \frac{2\left(\frac{1}{M}\right)\left(1 - \frac{1}{M}\right)|E(S)|}{kd}.$$

Note that $|E(S)| \leq Mkd/2$ and $|\delta(S)|/kd = M\phi(S) \leq Mf(\epsilon)$, so the expected value of

$$\phi(S') \leq Mf(\epsilon) + 1 - \frac{1}{M}.$$

For a small enough ϵ depending only on M , the expected value of $\phi(S') \leq 1 - 10\epsilon$. Therefore, with a constant probability, we have $\phi(S') < 1 - \epsilon$. This argument can be derandomized using standard techniques.

We show that random walk can be used to obtain nontrivial bicriteria approximation algorithms for the small sparsest cut problem. We do not know of an example showing that our analysis is tight. It would be interesting to find examples showing the limitations of random walk algorithms (e.g. showing that they fail to disprove the small set expansion conjecture).

References

1. N. Alon, V. Milman. *Isoperimetric inequalities for graphs, and superconcentrators*. Journal of Combinatorial Theory, Series B, 38(1), 73–88, 1985.
2. R. Anderson, F.R.K. Chung, K.J. Lang. *Local graph partitioning using PageRank vectors*. In Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS), 475–486, 2006.
3. R. Anderson, Y. Peres. *Finding sparse cuts locally using evolving sets*. In Proceedings of the 41st Annual ACM Symposium on Theory of Computing (STOC), 235–244, 2009.
4. S. Arora, B. Barak, D. Steurer. *Subexponential algorithms for unique games and related problems*. In Proceedings of the 51st Annual IEEE Symposium on Foundations of Computer Science (FOCS), 563–572, 2010.
5. S. Arora, S. Rao, U. Vazirani. *Expander flows, geometric embeddings and graph partitioning*. In Proceedings of the 36th Annual ACM Symposium on Theory of Computing (STOC), 222–231, 2004.
6. N. Bansal, U. Feige, R. Krauthgamer, K. Makarychev, V. Nagarajan, J. Naor, R. Schwartz. *Min-max graph partitioning and small set expansion*. In Proceedings of the 52nd Annual IEEE Symposium on Foundations of Computer Science (FOCS), 17–26, 2011.
7. J. Cheeger. *A lower bound for the smallest eigenvalue of the Laplacian*. Problems in Analysis, Princeton University Press, 195–199, 1970.
8. F.R.K. Chung. *A local graph partitioning algorithm using heat kernel pagerank*. In Proceedings of the 6th International Workshop on Algorithms and Models for the Web-Graph (WAW), 62–75, 2009.
9. S. Horry, N. Linial, A. Wigderson. *Expander graphs and their applications*. Bulletin of the American Mathematical Society 43(4), 439–561, 2006.
10. J.R. Lee, S. Oveis Gharan, L. Trevisan. *Multi-way spectral partitioning and higher-order Cheeger inequalities*. In Proceedings of the 44th Annual Symposium on Theory of Computing (STOC), 1117–1130, 2012.
11. F.T. Leighton, S. Rao. *Multicommodity max-flow min-cut theorem and their use in designing approximation algorithms*. Journal of the ACM 46(6), 787–832, 1999.
12. A. Louis, P. Raghavendra, P. Tetali, S. Vempala. *Many sparse cuts via higher eigenvalues*. In Proceedings of the 44th Annual ACM Symposium on Theory of Computing (STOC), 1131–1140, 2012.
13. L. Lovász, R. Kannan. *Faster mixing via average conductance*. In Proceedings of the 31st Annual ACM Symposium on Theory of Computing (STOC), 282–287, 1999.
14. L. Lovász, M. Simonovits. *The mixing time of Markov chains, an isoperimetric inequality, and computing the volume*. In Proceedings of the 31st Annual IEEE Symposium on Foundations of Computer Science (FOCS), 346–354, 1990.
15. P. Raghavendra, D. Steurer. *Graph expansion and the unique games conjecture*. In Proceedings of the 42nd Annual ACM Symposium on Theory of Computing (STOC), 755–764, 2010.
16. P. Raghavendra, D. Steurer, P. Tetali. *Approximations for the isoperimetric and spectral profile of graphs and related parameters*. In Proceedings of the 42nd Annual ACM Symposium on Theory of Computing (STOC), 631–640, 2010.
17. P. Raghavendra, D. Steurer, M. Tulsiani. *Reductions between expansion problems*. In Proceedings of the 27th Annual IEEE Conference on Computational Complexity (CCC), 2012.

18. D.A. Spielman, S.-H. Teng. *A local clustering algorithm for massive graphs and its applications to nearly-linear time graph partitioning*. CoRR, abs/0809.3232, 2008.
19. L. Orecchia, S. Sachdeva, N.K. Vishnoi. *Approximating the exponential, the Lanczos method, and an $\tilde{O}(m)$ -time spectral algorithm for balanced separator*. In Proceedings of the 44th Annual ACM Symposium on Theory of Computing (STOC), 1141–1160, 2012.
20. S. Oveis Gharan, L. Trevisan. *Approximating the expansion profile and almost optimal local graph clustering*. CoRR, abs/1204.2021, 2012.