Local Graph Partitioning using PageRank Vectors

Reid Andersen
University of California, San Diego

Fan Chung
University of California, San Diego

Kevin Lang
Yahoo! Research

Abstract

A local graph partitioning algorithm finds a cut near a specified starting vertex, with a running time that depends largely on the size of the small side of the cut, rather than the size of the input graph. In this paper, we present an algorithm for local graph partitioning using personalized PageRank vectors. We develop an improved algorithm for computing approximate PageRank vectors, and derive a mixing result for PageRank vectors similar to that for random walks. Using this mixing result, we derive an analogue of the Cheeger inequality for PageRank, which shows that a sweep over a single PageRank vector can find a cut with conductance ϕ, provided there exists a cut with conductance at most $f(\phi)$, where $f(\phi)$ is $\Omega(\phi^2/\log m)$, and where m is the number of edges in the graph. By extending this result to approximate PageRank vectors, we develop an algorithm for local graph partitioning that can be used to a find a cut with conductance at most ϕ, whose small side has volume at least 2^b, in time $O(2^b \log^3 m/\phi^2)$. Using this local graph partitioning algorithm as a subroutine, we obtain an algorithm that finds a cut with conductance ϕ and approximately optimal balance in time $O(m \log^4 m/\phi^3)$.

1 Introduction

One of the central problems in algorithmic design is the problem of finding a cut with a small conductance. There is a large literature of research papers on this topic, with applications in numerous areas.

Spectral partitioning, where an eigenvector is used to produce a cut, is one of the few approaches to this problem that can be analyzed theoretically. The Cheeger inequality [4] shows that the cut obtained by spectral partitioning has conductance within a quadratic factor of the optimum. Spectral partitioning can be applied recursively, with the resulting cuts combined in various ways, to solve more complicated problems; for example, recursive spectral algorithms have been used to find k-way partitions, spectral clusterings, and separators in planar graphs [2, 8, 13, 14]. There is no known way to lower bound the size of the small side of the cut produced by spectral partitioning, and this adversely affects the running time of recursive spectral partitioning.

Local spectral techniques provide a faster alternative to recursive spectral partitioning by avoiding the problem of unbalanced cuts. Spielman and Teng introduced a local partitioning algorithm called `Nibble`, which finds relatively small cuts near a specified starting vertex, in time proportional to the volume of the small side of the cut. The small cuts found by `Nibble` can be combined to form balanced cuts and multiway partitions in almost linear time, and the `Nibble` algorithm is an essential subroutine in algorithms for graph sparsification and solving linear systems [15]. The analysis of the `Nibble` algorithm is based on a mixing result by Lovász and Simonovits [9, 10], which shows that cuts with small conductance can be found by simulating a random walk and performing sweeps over the resulting sequence of walk vectors.

In this paper, we present a local graph partitioning algorithm that uses personalized PageRank vectors to produce cuts. Because a PageRank vector is defined recursively (as we will describe in section 2), we can consider a single PageRank vector in place of a sequence of random walk vectors, which simplifies the process of finding cuts and allows greater flexibility when computing approximations. We show directly that a sweep over a single approximate PageRank vector can produce cuts with small conductance. In contrast, Spielman and Teng show that when a good cut can be found from a series of walk distributions, a similar cut can be found from a series of approximate walk distributions. Our method of analysis allows us to find cuts using approximations with larger amounts of error, which improves the running time.

The analysis of our algorithm is based on the following results:

- We give an improved algorithm for computing approximate PageRank vectors. We use a technique introduced by Jeh-Widom [7], and further developed by Berkhin in his Bookmark Coloring Algorithm [1]. The algorithms of Jeh-Widom and Berkhin compute many personalized PageRank vectors simultaneously, more quickly than they could be computed individually. Our algorithm computes a single approximate PageRank vector more quickly than the algorithms of Jeh-Widom and Berkhin by a factor of $\log n$.

- We prove a mixing result for PageRank vectors that is similar to the Lovász-Simonovits mixing result for random walks. Using this mixing result, we show that if a sweep over a PageRank vector does not produce a cut with small conductance, then that PageRank vector is close to the stationary distribution. We then show that for any set C with small conductance, and for many starting vertices contained in C, the resulting PageRank vector is not close to the stationary distribution, because it has significantly more probability within C. Combining these results yields a local version of the Cheeger inequality for PageRank vectors: if C is a set with conductance $\Phi(C) \leq f(\phi)$, then a sweep over a PageRank vector $\mathrm{pr}(\alpha, \chi_v)$ finds a set with conductance at most ϕ, provided that α is set correctly depending on ϕ, and that v is one of a significant number of good starting vertices within C. This holds for a function $f(\phi)$ that satisfies $f(\phi) = \Omega(\phi^2/\log m)$.

Using the results described above, we produce a local partitioning algorithm `PageRank-Nibble` which improves both the running time and approximation ratio of `Nibble`. `PageRank-Nibble` takes as input a starting vertex v, a target conductance ϕ, and an integer $b \in [1, \log m]$. When v is a good starting vertex for a set C with conductance $\Phi(C) \leq g(\phi)$, there is at least one value of b where `PageRank-Nibble` produces a set S with the following properties: the conductance of S is at most ϕ, the volume of S is at least 2^{b-1} and at most $(2/3)\mathrm{vol}(G)$, and the intersection of S and C satisfies $\mathrm{vol}(S \cap C) \geq 2^{b-2}$. This holds for a function $g(\phi)$ that satisfies $g(\phi) = \Omega(\phi^2/\log^2 m)$. The running time of `PageRank-Nibble` is $O(2^b \log^3 m/\phi^2)$, which is nearly linear in the volume of S. In comparison, the `Nibble` algorithm requires that C have conductance $O(\phi^3/\log^2 m)$, and runs in time $O(2^b \log^4 m/\phi^5)$.

`PageRank-Nibble` can be used interchangeably with `Nibble`, leading immediately to faster algorithms with improved approximation ratios in several applications. In particular, we obtain an algorithm `PageRank-Partition` that finds cuts with small conductance and approximately optimal balance: if there exists a set C satisfying $\Phi(C) \leq g(\phi)$ and $\mathrm{vol}(C) \leq \frac{1}{2}\mathrm{vol}(G)$, then the algorithm finds a set S such that $\Phi(S) \leq \phi$ and $\frac{1}{2}\mathrm{vol}(C) \leq \mathrm{vol}(S) \leq \frac{5}{6}\mathrm{vol}(G)$, in time $O(m \log^4 m/\phi^3)$. This holds for a function $g(\phi)$ that satisfies $g(\phi) = \Omega(\phi^2/\log^2 m)$.

2 Preliminaries

In this paper we consider an undirected, unweighted graph G, where V is the vertex set, E is the edge set, n is the number of vertices, and m is the number of undirected edges. We write $d(v)$ for the degree of vertex v, let D be the degree matrix (the diagonal matrix with $D_{i,i} = d(v_i)$), and let A be the adjacency matrix. We will consider *distributions* on V, which are vectors indexed by the vertices in V, with the additional requirement that each entry be nonnegative. A distribution p is considered to be a row vector, so we can write the product of p and A as pA.

2.1. Personalized Pagerank Vectors

PageRank was introduced by Brin and Page [12, 3]. For convenience, we introduce a lazy variation of PageRank, which we define to be the unique solution $\mathrm{pr}(\alpha, s)$ of the equation

$$\mathrm{pr}(\alpha, s) = \alpha s + (1-\alpha)\mathrm{pr}(\alpha, s)W, \tag{1}$$

where α is a constant in $(0, 1]$ called the *teleportation constant*, s is a distribution called the *preference vector*, and W is the lazy random walk transition matrix $W = \frac{1}{2}(I + D^{-1}A)$. In the Appendix, we show that this is equivalent to the traditional definition of PageRank (which uses a regular random walk step instead of a lazy step) up to a change in α.

The PageRank vector that is usually associated with search ranking has a preference vector equal to the uniform distribution $\frac{\vec{1}}{n}$. PageRank vectors whose preference vectors are concentrated on a smaller set of vertices are often called *personalized PageRank vectors*. These were introduced by Haveliwala [6], and have been used to provide personalized search ranking and context-sensitive search [1, 5, 7]. The preference vectors used in our algorithms have all probability on a single starting vertex.

Here are some useful properties of PageRank vectors (also see [6] and [7]). The proofs are given in the Appendix.

Proposition 1. *For any starting distribution s, and any constant α in $(0, 1]$, there is a unique vector $\mathrm{pr}(\alpha, s)$ satisfying $\mathrm{pr}(\alpha, s) = \alpha s + (1-\alpha)\mathrm{pr}(\alpha, s)W$.*

Proposition 2. *For any fixed value of α in $(0, 1]$, there is a linear transformation R_α such that $\mathrm{pr}(\alpha, s) = sR_\alpha$. Furthermore, R_α is given by the matrix*

$$R_\alpha = \alpha \sum_{t=0}^{\infty} (1-\alpha)^t W^t, \tag{2}$$

which implies that a PageRank vector is a weighted average of lazy walk vectors,

$$\mathrm{pr}(\alpha, s) = \alpha \sum_{t=0}^{\infty} (1-\alpha)^t \left(sW^t\right). \tag{3}$$

It follows that $\mathrm{pr}(\alpha, s)$ is linear in the preference vector s.

2.2 Conductance

The *volume* of a subset $S \subseteq V$ of vertices is

$$\mathrm{vol}(S) = \sum_{x \in S} d(x).$$

We remark that $\mathrm{vol}(V) = 2m$, and we will sometimes write $\mathrm{vol}(G)$ in place of $\mathrm{vol}(V)$. The *edge boundary* of a set is defined to be $\partial(S) = \{\{x, y\} \in E \mid x \in S, y \notin S\}$, and the *conductance* of a set is

$$\Phi(S) = \frac{|\partial(S)|}{\min\left(\mathrm{vol}(S), 2m - \mathrm{vol}(S)\right)}.$$

2.3. Distributions

Two distributions we will use frequently are the stationary distribution,

$$\psi_S(x) = \begin{cases} \frac{d(x)}{\mathrm{vol}(S)} & \text{if } x \in S \\ 0 & \text{otherwise} \end{cases}.$$

and the indicator function,

$$\chi_v(x) = \begin{cases} 1 & \text{if } x = v \\ 0 & \text{otherwise} \end{cases}.$$

The amount of probability from a distribution p on a set S of vertices is written

$$p\left(S\right) = \sum_{x \in S} p(x).$$

We will sometimes refer to the quantity $p(S)$ as an amount of probability even if $p(V)$ is not equal to 1. As an example of this notation, the PageRank vector with teleportation constant α and preference vector χ_v is written $\mathrm{pr}(\alpha, \chi_v)$, and the amount of probability from this distribution on a set S is written $\left[\mathrm{pr}(\alpha, \chi_v)\right](S)$. The support of a distribution is $\mathrm{Supp}(p) = \{v \mid p(v) \neq 0\}$.

2.4. Sweeps

A *sweep* is an efficient technique for producing cuts from an embedding of a graph, and is often used in spectral partitioning [11, 14]. We will use the following degree-normalized version of a sweep. Given a distribution p, with support size $N_p = |\mathrm{Supp}(p)|$, let $v_1, \ldots, v_{N_p}$ be an ordering of the vertices such that $\frac{p(v_i)}{d(v_i)} \geq \frac{p(v_{i+1})}{d(v_{i+1})}$. This produces a collection of sets, $S_j^p = \{v_1, \ldots, v_j\}$ for each $j \in \{0, \ldots, N_p\}$, which we call *sweep sets*. We let

$$\Phi(p) = \min_{j \in [1, N_p]} \Phi(S_j^p)$$

be the smallest conductance of any of the sweep sets. A cut with conductance $\Phi(p)$ can be found by sorting p and computing the conductance of each sweep set, which can be done in time $O(\mathrm{vol}(\mathrm{Supp}(p)) \log n)$.

2.5. Measuring the spread of a distribution

We measure how well a distribution p is spread in the graph using a function $p\left[k\right]$ defined for all integers $k \in [0, 2m]$. This function is determined by setting

$$p\left[k\right] = p\left(S_j^p\right),$$

for those values of k where $k = \mathrm{vol}(S_j^p)$, and the remaining values are set by defining $p\left[k\right]$ to be piecewise linear between these points. In other words, for any integer $k \in [0, 2m]$, if j is the unique vertex such that $\mathrm{vol}(S_j^p) \leq k \leq \mathrm{vol}(S_{j+1}^p)$, then

$$p\left[k\right] = p\left(S_j^p\right) + \frac{k - \mathrm{vol}(S_j^p)}{d(v_j)} p\left(v_{j+1}\right).$$

This implies that $p\left[k\right]$ is an increasing function of k, and a concave function of k. It is not hard to see that $p\left[k\right]$ is an upper bound on the amount of probability from p on any set with volume k; for any set S, we have $p\left(S\right) \leq p\left[\mathrm{vol}(S)\right]$.

3 Computing approximate PageRank vectors

To approximate a PageRank vector $\mathrm{pr}(\alpha, s)$, we compute a pair of distributions p and r with the following property.

$$p + \mathrm{pr}(\alpha, r) = \mathrm{pr}(\alpha, s). \tag{4}$$

If p and r are two distributions with this property, we say that p is an *approximate PageRank vector*, which approximates $\mathrm{pr}(\alpha, s)$ with the *residual vector* r. We will use the notation $p = \mathrm{apr}(\alpha, s, r)$ to refer to an approximate PageRank vector obeying the equation above. Since the residual vector is nonnegative, it is always true that $\mathrm{apr}(\alpha, s, r) \leq \mathrm{pr}(\alpha, s)$, for any residual vector r.

In this section, we give an algorithm that computes an approximate PageRank vector with a small residual vector and small support, with running time independent of the size of the graph.

Theorem 1. `ApproximatePageRank`(v, α, ϵ) *runs in time* $O(\frac{1}{\epsilon\alpha})$*, and computes an approximate PageRank vector* $p = \mathrm{apr}(\alpha, \chi_v, r)$ *such that the residual vector* r *satisfies* $\max_{u \in V} \frac{r(u)}{d(u)} < \epsilon$*, and such that* $\mathrm{vol}(\mathrm{Supp}(p)) \leq \frac{1}{\epsilon\alpha}$.

We remark that this algorithm is based on the algorithms of Jeh-Widom [7] and Berkhin [1], both of which can be used to compute similar approximate PageRank vectors in time $O(\frac{\log n}{\epsilon\alpha})$. The extra factor of $\log n$ in the running time of these algorithms is overhead from maintaining a heap or priority queue, which we eliminate. The proof of Theorem 1 is based on a series of facts which we describe below.

Our algorithm is motivated by the following observation of Jeh-Widom.

$$\mathrm{pr}(\alpha, s) = \alpha s + (1 - \alpha)\mathrm{pr}(\alpha, sW). \tag{5}$$

Notice that the equation above is similar to, but different from, the equation used in Section 2 to define PageRank. This observation is simple, but it is instrumental in our algorithm, and it is not trivial. To prove it, first reformulate the linear transformation R_α that takes a starting distribution to its corresponding PageRank vector, as follows.

$$\begin{aligned} R_\alpha &= \alpha \sum_{t=0}^{\infty} (1-\alpha)^t W^t \\ &= \alpha I + (1-\alpha) W R_\alpha. \end{aligned}$$

Applying this rearranged transformation to a starting distribution s yields equation (5).

$$\begin{aligned} \mathrm{pr}(\alpha, s) &= s R_\alpha \\ &= \alpha s + (1-\alpha) s W R_\alpha \\ &= \alpha s + (1-\alpha)\mathrm{pr}(\alpha, sW). \end{aligned}$$

This provides a flexible way to compute an approximate PageRank vector. We maintain a pair of distributions: an approximate PageRank vector p and its associated residual vector r. Initially, we set $p = \vec{0}$ and $r = \chi_v$. We then apply a series of push operations, based on equation (5), which alter p and r. Each push operation takes a single vertex u, moves an α fraction of the probability from $r(u)$ onto $p(u)$, and then spreads the remaining $(1 - \alpha)$ fraction within r, as if a single step of the lazy random walk were applied only to the vertex u. Each push operation maintains the invariant

$$p + \mathrm{pr}(\alpha, r) = \mathrm{pr}(\alpha, \chi_v), \tag{6}$$

which ensures that p is an approximate PageRank vector for $\mathrm{pr}(\alpha, \chi_v)$ after any sequence of push operations. We now formally define $\texttt{push}_\texttt{u}$, which performs this push operation on the distributions p and r at a chosen vertex u.

$\texttt{push}_u(p, r)$:

1. Let $p' = p$ and $r' = r$, except for the following changes:
   (a) $p'(u) = p(u) + \alpha r(u)$.
   (b) $r'(u) = (1 - \alpha)r(u)/2$.
   (c) For each v such that $(u, v) \in E$: $\quad r'(v) = r(v) + (1 - \alpha)r(u)/(2d(u))$.
2. Return (p', r').

Lemma 1. *Let p' and r' be the result of the operation $\texttt{push}_u$ on p and r. Then*

$$p' + \mathrm{pr}(\alpha, r') = p + \mathrm{pr}(\alpha, r).$$

The proof of Lemma 1 can be found in the Appendix. During each push, some probability is moved from r to p, where it remains, and after sufficiently many pushes r can be made small. We can bound the number of pushes required by the following algorithm.

`ApproximatePageRank` (v, α, ϵ):

1. Let $p = \vec{0}$, and $r = \chi_v$.
2. While $\max_{u \in V} \frac{r(u)}{d(u)} \geq \epsilon$:
   (a) Choose any vertex u where $\frac{r(u)}{d(u)} \geq \epsilon$.
   (b) Apply $\texttt{push}_u$ at vertex u, updating p and r.
3. Return p, which satisfies $p = \mathrm{apr}(\alpha, \chi_v, r)$ with $\max_{u \in V} \frac{r(u)}{d(u)} < \epsilon$.

Lemma 2. *Let T be the total number of push operations performed by* `ApproximatePageRank`*, and let d_i be the degree of the vertex u used in the ith push. Then*

$$\sum_{i=1}^{T} d_i \leq \frac{1}{\epsilon\alpha}.$$

Proof. The amount of probability on the vertex pushed at time i is at least ϵd_i, therefore $|r|_1$ decreases by at least $\alpha\epsilon d_i$ during the ith push. Since $|r|_1 = 1$ initially, we have $\alpha\epsilon \sum_{i=1}^{T} d_i \leq 1$, and the result follows. □

To implement `ApproximatePageRank`, we determine which vertex to push at each step by maintaining a queue containing those vertices u with $r(u)/d(u) \geq \epsilon$. At each step, push operations are performed on the first vertex in the queue until $r(u)/d(u) < \epsilon$ for that vertex, which is then removed from the queue. If a push operation raises the value of $r(x)/d(x)$ above ϵ for some vertex x, that vertex is added to the back of the queue. This continues until the queue is empty, at which point every vertex has $r(u)/d(u) < \epsilon$. We will show that this algorithm has the properties promised in Theorem 1. The proof is contained in the Appendix.

4 A mixing result for PageRank vectors

In this section, we prove a mixing result for PageRank vectors that is an analogue of the Lovász-Simonovits mixing result for random walks. For an approximate PageRank vector $\mathrm{apr}(\alpha, s, r)$, we give an upper bound on $\mathrm{apr}(\alpha, s, r)\,[k]$ that depends on the smallest conductance found by a sweep over $\mathrm{apr}(\alpha, s, r)$. In contrast, the mixing result of Lovász and Simonovits bounds the quantity $p^{(t)}\,[k]$ for the lazy random walk distribution $p^{(t)}$ in terms of the smallest conductance found by sweeps over the previous walk distributions $p^{(0)}, \ldots, p^{(t-1)}$. The recursive property of PageRank allows us to consider a single vector instead of a sequence of random walk vectors, simplifying the process of finding cuts.

We use this mixing result to show that if an approximate PageRank vector $\mathrm{apr}(\alpha, s, r)$ has significantly more probability than the stationary distribution on any set, the sweep over $\mathrm{apr}(\alpha, s, r)$ produces a cut with small conductance.

Theorem 2. *If there exists a set S of vertices and a constant $\delta \geq \frac{2}{\sqrt{m}}$ satisfying*

$$\mathrm{apr}(\alpha, s, r)(S) - \frac{\mathrm{vol}(S)}{\mathrm{vol}(G)} > \delta,$$

then

$$\Phi(\mathrm{apr}(\alpha, s, r)) < \sqrt{\frac{18\alpha \ln m}{\delta}}.$$

The proof of this theorem, and the more general mixing result from which it is derived, is described at the end of this section. The proof requires a sequence of lemmas, which we present below.

Every approximate PageRank vector, no matter how large the residual vector, obeys the following inequality. It is a one-sided version of the equation used to define PageRank.

Lemma 3. *If* $\mathrm{apr}(\alpha, s, r)$ *is an approximate PageRank vector, then*

$$\mathrm{apr}(\alpha, s, r) \leq \alpha s + (1 - \alpha)\mathrm{apr}(\alpha, s, r)W.$$

The proof of Lemma 3 can be found in the Appendix. Notice that this inequality relates $\mathrm{apr}(\alpha, s, r)$ to $\mathrm{apr}(\alpha, s, r)W$. We will soon prove a result, Lemma 4, which describes how probability mixes in the single walk step between $\mathrm{apr}(\alpha, s, r)$ and $\mathrm{apr}(\alpha, s, r)W$. We will then combine Lemma 4 with the inequality from Lemma 3 to relate $\mathrm{apr}(\alpha, s, r)$ to itself, removing any reference to $\mathrm{apr}(\alpha, s, r)W$.

We now present definitions required for Lemma 4. Instead of viewing an undirected graph as a collection of undirected edges, we view each undirected edge $\{u, v\}$ as a pair of directed edges (u, v) and (v, u). For each directed edge (u, v) we let

$$p(u, v) = \frac{p(u)}{d(u)}.$$

For any set of directed edges A, we define

$$p(A) = \sum_{(u,v)\in A} p(u, v).$$

When a lazy walk step is applied to the distribution p, the amount of probability that moves from u to v is $\frac{1}{2}p(u,v)$. For any set S of vertices, we have the set of directed edges into S, and the set of directed edges out of S, defined by $\text{in}(S) = \{(u,v) \in E \mid u \in S\}$, and $\text{out}(S) = \{(u,v) \in E \mid v \in S\}$, respectively.

Lemma 4. *For any distribution p, and any set S of vertices,*

$$pW(S) \leq \frac{1}{2}\left(p\left(\text{in}(S) \cup \text{out}(S)\right) + p\left(\text{in}(S) \cap \text{out}(S)\right)\right).$$

The proof of Lemma 4 can be found in the Appendix. We now combine this result with the inequality from Lemma 3 to relate $\text{apr}(\alpha, s, r)$ to itself. In contrast, the proof of Lovász and Simonovits [9, 10] relates the walk distributions $p^{(t)}$ and $p^{(t+1)}$, where $p^{(t+1)} = p^{(t)}W$, and $p^{(0)} = s$.

Lemma 5. *If $p = \text{apr}(\alpha, s, r)$ is an approximate PageRank vector, then for any set S of vertices,*

$$p(S) \leq \alpha s(S) + (1-\alpha)\frac{1}{2}\left(p\left(\text{in}(S) \cup \text{out}(S)\right) + p\left(\text{in}(S) \cap \text{out}(S)\right)\right).$$

Furthermore, for each $j \in [1, n-1]$,

$$p\left[\text{vol}(S_j^p)\right] \leq \alpha s\left[\text{vol}(S_j^p)\right] + (1-\alpha)\frac{1}{2}\left(p\left[\text{vol}(S_j^p) - |\partial(S_j^p)|\right] + p\left[\text{vol}(S_j^p) + |\partial(S_j^p)|\right]\right).$$

The proof of Lemma 5 is included in the Appendix.

The following lemma uses the result from Lemma 5 to place an upper bound on $\text{apr}(\alpha, s, r)[k]$. More precisely, it shows that if a certain upper bound on $\text{apr}(\alpha, s, r)[k] - \frac{k}{2m}$ does not hold, then one of the sweep sets from $\text{apr}(\alpha, s, r)$ has both small conductance and a significant amount of probability from $\text{apr}(\alpha, s, r)$. This lower bound on probability will be used in Section 6 to control the volume of the resulting sweep set.

Theorem 3. *Let $p = \text{apr}(\alpha, s, r)$ be an approximate PageRank vector with $|s|_1 \leq 1$. Let ϕ and γ be any constants in $[0, 1]$. Either the following bound holds for any integer t and any $k \in [0, 2m]$:*

$$p[k] - \frac{k}{2m} \leq \gamma + \alpha t + \sqrt{\min(k, 2m-k)}\left(1 - \frac{\phi^2}{8}\right)^t,$$

or else there exists a sweep cut S_j^p with the following properties:

1. $\Phi(S_j^p) < \phi$,
2. $p\left(S_j^p\right) - \frac{\text{vol}(S_j^p)}{2m} > \gamma + \alpha t + \sqrt{\min(\text{vol}(S_j^p), 2m - \text{vol}(S_j^p))}\left(1 - \frac{\phi^2}{8}\right)^t$, *for some integer t,*
3. $j \in [1, |\text{Supp}(p)|]$.

The proof can be found in the Appendix.

We can rephrase the sequence of bounds from Theorem 3 to prove the theorem promised at the beginning of this section. Namely, we show that if there exists a set of vertices, of any size, that contains a constant amount more probability from $\text{apr}(\alpha, s, r)$ than from the stationary distribution, then the sweep over $\text{apr}(\alpha, s, r)$ finds a cut with conductance roughly $\sqrt{\alpha \ln m}$. We remark that this applies to any approximate PageRank vector, regardless of the size of the residual vector: the residual vector only needs to be small to ensure that $\text{apr}(\alpha, s, r)$ is large enough that the theorem applies. The proof is given in the appendix.

5 Local partitioning using approximate PageRank vectors

In this section, we show how sweeps over approximate PageRank vectors can be used to find cuts with nearly optimal conductance. Unlike traditional spectral partitioning, where a sweep over an eigenvector produces a cut with conductance near the global minimum, the cut produced by a PageRank vector depends on the starting vertex v, and also on α. We first identify a sizeable collection of starting vertices for which we can give a lower bound on $\mathrm{apr}(\alpha, \chi_v, r)(C)$.

Theorem 4. *For any set C and any constant α, there is a subset $C_\alpha \subseteq C$, with $\mathrm{vol}(C_\alpha) \geq \mathrm{vol}(C)/2$, such that for any vertex $v \in C_\alpha$, the approximate PageRank vector $\mathrm{apr}(\alpha, \chi_v, r)$ satisfies*

$$\mathrm{apr}(\alpha, \chi_v, r)(C) \geq 1 - \frac{\Phi(C)}{\alpha} - \mathrm{vol}(C) \max_{u \in V} \frac{r(u)}{d(u)}.$$

We will outline the proof of Theorem 4 at the end of this section. Theorem 4 can be combined with the mixing results from Section 4 to prove the following theorem, which describes a method for producing cuts from an approximate PageRank vector.

Theorem 5. *Let ϕ be a constant in $[0, 1]$, let $\alpha = \frac{\phi^2}{135 \ln m}$, and let C be a set satisfying*

1. $\Phi(C) \leq \frac{\phi^2}{1350 \ln m}$,
2. $\mathrm{vol}(C) \leq \frac{2}{3}\mathrm{vol}(G)$.

If $v \in C_\alpha$, and if $\mathrm{apr}(\alpha, \chi_v, r)$ is an approximate PageRank vector with residual vector r satisfying $\max_{u \in V} \frac{r(u)}{d(u)} \leq \frac{1}{10\mathrm{vol}(C)}$, then $\Phi(\mathrm{apr}(\alpha, \chi_v, r)) < \phi$.

We prove Theorem 5 by combining Theorem 4 and Theorem 2. A detailed proof is provided in the Appendix. As an immediate consequence of Theorem 5, we obtain a local Cheeger inequality for personalized PageRank vectors, which applies when the starting vertex is within a set that achieves the minimum conductance in the graph.

Theorem 6. *Let $\Phi(G)$ be the minimum conductance of any set with volume at most $\mathrm{vol}(G)/2$, and let C^{opt} be a set achieving this minimum. If $\mathrm{pr}(\alpha, \chi_v)$ is a PageRank vector where $\alpha = 10\Phi(G)$, and $v \in C^{\mathrm{opt}}_\alpha$, then*

$$\Phi(\mathrm{pr}(\alpha, \chi_v)) < \sqrt{1350\Phi(G) \ln m}.$$

Theorem 6 follows immediately from Theorem 5 by setting $\phi = \sqrt{1350\Phi(G) \ln m}$.

To prove Theorem 4, we will show that a set C with small conductance contains a significant amount of probability from $\mathrm{pr}(\alpha, \chi_v)$, for many of the vertices v in C. We first show that this holds for an average of the vertices in C, by showing that C contains a significant amount of probability from $\mathrm{pr}(\alpha, \psi_C)$.

Lemma 6. *The PageRank vector $\mathrm{pr}(\alpha, \psi_C)$ satisfies*

$$[\mathrm{pr}(\alpha, \psi_C)](\bar{C}) \leq \frac{\Phi(C)}{2\alpha}.$$

The proof of Lemma 6 will be given in the Appendix. To prove Theorem 4 from Lemma 6, we observe that for many vertices in C, $\mathrm{pr}(\alpha, \chi_v)$ is not much larger than $\mathrm{pr}(\alpha, \psi_C)$, and then bound the difference between $\mathrm{apr}(\alpha, \chi_v, r)$ and $\mathrm{pr}(\alpha, \chi_v)$ in terms of the residual vector r. A detailed proof can be found in the Appendix.

6 An algorithm for nearly linear time graph partitioning

In this section, we extend our local partitioning techniques to find a set with small conductance, while providing more control over the volume of the set produced. The result is an algorithm called `PageRank-Nibble` that takes a scale b as part of its input, runs in time proportional to 2^b, and only produces a cut when it finds a set with conductance ϕ and volume roughly 2^b. We prove that `PageRank-Nibble` finds a set with these properties for at least one value of $b \in [1, \lceil \log m \rceil]$, provided that v is a good starting vertex for a set of conductance at most $g(\phi)$, where $g(\phi) = \Omega(\phi^2 / \log^2 m)$.

`PageRank-Nibble`(v, ϕ, b):
Input: a vertex v, a constant $\phi \in (0, 1]$, and an integer $b \in [1, B]$, where $B = \lceil \log_2 m \rceil$.

1. Let $\alpha = \frac{\phi^2}{225 \ln(100\sqrt{m})}$.
2. Compute an approximate PageRank vector $p = \mathrm{apr}(\alpha, \chi_v, r)$ with residual vector r satisfying $\max_{u \in V} \frac{r(u)}{d(u)} \leq 2^{-b} \frac{1}{48B}$.
3. Check each set S_j^p with $j \in [1, |\mathrm{Supp}(p)|]$, to see if it obeys the following conditions:

   Conductance: $\Phi(S_j^p) < \phi$,

   Volume: $2^{b-1} < \mathrm{vol}(S_j^p) < \frac{2}{3}\mathrm{vol}(G)$,

   Probability Change: $p\left[2^b\right] - p\left[2^{b-1}\right] > \frac{1}{48B}$,
4. If some set S_j^p satisfies all of these conditions, return S_j^p. Otherwise, return nothing.

Theorem 7. `PageRank-Nibble`(v, ϕ, b) *can be implemented with running time* $O(2^b \frac{\log^3 m}{\phi^2})$.

Theorem 8. *Let C be a set satisfying $\Phi(C) \leq \phi^2/(22500 \log^2 100m)$ and $\mathrm{vol}(C) \leq \frac{1}{2}\mathrm{vol}(G)$, and let v be a vertex in C_α for $\alpha = \phi^2/(225 \ln(100\sqrt{m}))$. Then, there is some integer $b \in [1, \lceil \log_2 m \rceil]$ for which* `PageRank-Nibble`(v, ϕ, b) *returns a set S. Any set S returned by* `PageRank-Nibble`(v, ϕ, b) *has the following properties:*

1. $\Phi(S) < \phi$,
2. $2^{b-1} < \mathrm{vol}(S) < \frac{2}{3}\mathrm{vol}(G)$,
3. $\mathrm{vol}(S \cap C) > 2^{b-2}$.

The proofs of Theorems 7 and 8 are included in the Appendix.

`PageRank-Nibble` improves both the running time and approximation ratio of the `Nibble` algorithm of Spielman and Teng, which runs in time $O(2^b \log^4 m/\phi^5)$, and requires $\Phi(C) = O(\phi^3/\log^2 m)$. `PageRank-Nibble` can be used interchangeably with `Nibble` in several important applications. For example, both `PageRank-Nibble` and `Nibble` can be applied recursively to produce cuts with nearly optimal balance. An algorithm `PageRank-Partition` with the following properties can be created in essentially the same way as the algorithm `Partition` in [15], so we omit the details.

Theorem 9. *The algorithm* `PageRank-Partition` *takes as input a parameter ϕ, and has expected running time $O(m \log(1/p) \log^4 m/\phi^3)$. If there exists a set C with $\mathrm{vol}(C) \leq \frac{1}{2}\mathrm{vol}(G)$ and $\Phi(C) \leq \phi^2/(1845000 \log^2 m)$, then with probability at least $1 - p$,* `PageRank-Partition` *produces a set S satisfying $\Phi(S) \leq \phi$ and $\frac{1}{2}\mathrm{vol}(C) \leq \mathrm{vol}(S) \leq \frac{5}{6}\mathrm{vol}(G)$.*

References

[1] Pavel Berkhin. Bookmark-coloring approach to personalized pagerank computing. *Internet Mathematics*, To appear.

[2] Christian Borgs, Jennifer T. Chayes, Mohammad Mahdian, and Amin Saberi. Exploring the community structure of newsgroups. In *KDD*, pages 783–787, 2004.

[3] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117, 1998.

[4] F. Chung. *Spectral graph theory*, volume Number 92 in CBMS Regional Conference Series in Mathematics. American Mathematical Society, 1997.

[5] D. Fogaras and B. Racz. Towards scaling fully personalized pagerank. In *Proceedings of the 3rd Workshop on Algorithms and Models for the Web-Graph (WAW)*, pages pages 105–117, October 2004.

[6] Taher H. Haveliwala. Topic-sensitive pagerank: A context-sensitive ranking algorithm for web search. *IEEE Trans. Knowl. Data Eng.*, 15(4):784–796, 2003.

[7] Glen Jeh and Jennifer Widom. Scaling personalized web search. In *Proceedings of the 12th World Wide Web Conference (WWW)*, pages 271–279, 2003.

[8] Ravi Kannan, Santosh Vempala, and Adrian Vetta. On clusterings: Good, bad and spectral. *J. ACM*, 51(3):497–515, 2004.

[9] László Lovász and Miklós Simonovits. The mixing rate of markov chains, an isoperimetric inequality, and computing the volume. In *FOCS*, pages 346–354, 1990.

[10] László Lovász and Miklós Simonovits. Random walks in a convex body and an improved volume algorithm. *Random Struct. Algorithms*, 4(4):359–412, 1993.

[11] M. Mihail. Conductance and convergence of markov chains—a combinatorial treatment of expanders. In *Proc. of 30th FOCS*, pages pp. 526–531, 1989.

[12] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[13] Horst D. Simon and Shang-Hua Teng. How good is recursive bisection? *SIAM Journal on Scientific Computing*, 18(5):1436–1445, 1997.

[14] Daniel A. Spielman and Shang-Hua Teng. Spectral partitioning works: Planar graphs and finite element meshes. In *IEEE Symposium on Foundations of Computer Science*, pages 96–105, 1996.

[15] Daniel A. Spielman and Shang-Hua Teng. Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems. In *ACM STOC-04*, pages 81–90, New York, NY, USA, 2004. ACM Press.

7 Appendix

To demonstrate the equivalence of lazy and standard PageRank vectors, let $\mathrm{rpr}(\alpha, s)$ be the standard PageRank vector, defined to be the unique solution p of the equation $p = \alpha s + (1-\alpha)pM$, where M is the random walk transition matrix $M = D^{-1}A$. We prove the following proposition.

Proposition 3. $\mathrm{pr}(\alpha, s) = \mathrm{rpr}(\frac{2\alpha}{1+\alpha}, s)$.

Proof. We have the following sequence of equations.

$$\begin{aligned}
\mathrm{pr}(\alpha, s) &= \alpha s + (1-\alpha)\mathrm{pr}(\alpha, s)W \\
\mathrm{pr}(\alpha, s) &= \alpha s + (\frac{1-\alpha}{2})\mathrm{pr}(\alpha, s) + (\frac{1-\alpha}{2})\mathrm{pr}(\alpha, s)(D^{-1}A) \\
(\frac{1+\alpha}{2})\mathrm{pr}(\alpha, s) &= \alpha s + (\frac{1-\alpha}{2})\mathrm{pr}(\alpha, s)(D^{-1}A) \\
\mathrm{pr}(\alpha, s) &= (\frac{2\alpha}{1+\alpha})s + (\frac{1-\alpha}{1+\alpha})\mathrm{pr}(\alpha, s)m
\end{aligned}$$

Since $\mathrm{pr}(\alpha, s)$ satisfies the equation for $\mathrm{rpr}(\frac{2\alpha}{1+\alpha}, s)$, and since this equation has a unique solution, the result follows. □

Proof of Proposition 1. The equation $p = \alpha s + (1-\alpha)pW$ is equivalent to $\alpha s = p[I - (1-\alpha)W]$. The matrix $(I - (1-\alpha)W)$ is nonsingular, since it is strictly diagonally dominant, so this equation has a unique solution p. □

Proof of Proposition 2. The sum in equation (2) that defines R_α is convergent for $\alpha \in (0, 1]$, and the following computation shows that sR_α obeys the steady state equation for $\mathrm{pr}(\alpha, s)$.

$$\begin{aligned}
\alpha s + (1-\alpha)sR_\alpha W &= \alpha s + (1-\alpha)s\left(\alpha\sum_{t=0}^{\infty}(1-\alpha)^t W^t\right)W \\
&= \alpha s + s\left(\alpha\sum_{t=1}^{\infty}(1-\alpha)^t W^t\right) \\
&= s\left(\alpha\sum_{t=0}^{\infty}(1-\alpha)^t W^t\right) \\
&= sR_\alpha.
\end{aligned}$$

Since the solution to the steady state equation is unique by Proposition 1, it follows that $\mathrm{pr}(\alpha, s) = sR_\alpha$. □

Proof of Lemma 1. After the push operation, we have

$$\begin{aligned}
p' &= p + \alpha r(u)\chi_u. \\
r' &= r - r(u)\chi_u + (1-\alpha)r(u)\chi_u W.
\end{aligned}$$

Using equation (5),

$$\begin{aligned}
p + \mathrm{pr}(\alpha, r) &= p + \mathrm{pr}(\alpha, r - r(u)\chi_u) + \mathrm{pr}(\alpha, r(u)\chi_u) \\
&= p + \mathrm{pr}(\alpha, r - r(u)\chi_u) + [\alpha r(u)\chi_u + (1-\alpha)\mathrm{pr}(\alpha, r(u)\chi_u W)] \\
&= [p + \alpha r(u)\chi_u] + \mathrm{pr}(\alpha, [r - r(u)\chi_u + (1-\alpha)r(u)\chi_u W]) \\
&= p' + \mathrm{pr}(\alpha, r').
\end{aligned}$$

□

Proof of Theorem 1. Lemma 1 implies that $p + \mathrm{pr}(\alpha, r) = \mathrm{pr}(\alpha, \chi_v)$ at every step of the algorithm, and so the vector returned by the algorithm is an approximate PageRank vector $\mathrm{apr}(\alpha, \chi_v, r)$. It is clear from the stopping condition that $\max_{u \in V} \frac{r(u)}{d(u)} < \epsilon$.

To bound the support volume, notice that for each vertex in $\mathrm{Supp}(p)$, `ApproximatePageRank` must have performed at least one push operation on that vertex. If d_i is the degree of the vertex pushed during step i, then Lemma 2 implies

$$\mathrm{vol}(\mathrm{Supp}(p)) = \sum_{v \in \mathrm{Supp}(p)} d(v) \leq \sum_{i=1}^{T} d_i \leq \frac{1}{\epsilon\alpha}.$$

It is possible to perform a push operation on the vertex u, and perform any necessary queue updates, in time proportional to $d(u)$. The running time then follows from Lemma 2. □

Proof of Lemma 3. From the definition of the approximate PageRank vector $\mathrm{apr}(\alpha, s, r)$, we have the following sequence of equations.

$$\begin{aligned}
\mathrm{apr}(\alpha, s, r) &= \mathrm{pr}(\alpha, s) - \mathrm{pr}(\alpha, r) \\
&= \alpha s + (1-\alpha)\mathrm{pr}(\alpha, s)W - \mathrm{pr}(\alpha, r) \\
&= \alpha s + (1-\alpha)(\mathrm{pr}(\alpha, s) - \mathrm{pr}(\alpha, r))W + (1-\alpha)\mathrm{pr}(\alpha, r)W - \mathrm{pr}(\alpha, r) \\
&= \alpha s + (1-\alpha)\mathrm{apr}(\alpha, s, r)W + ((1-\alpha)\mathrm{pr}(\alpha, r)W - \mathrm{pr}(\alpha, r)) \\
&= \alpha s + (1-\alpha)\mathrm{apr}(\alpha, s, r)W - \alpha r \\
&\leq \alpha s + (1-\alpha)\mathrm{apr}(\alpha, s, r)W.
\end{aligned}$$

The last line uses the fact that r is nonnegative. □

Proof of Lemma 4. The amount of probability from pW on a vertex u can be written as follows.

$$\begin{aligned}
pW(u) &= \frac{1}{2}p(u) + \frac{1}{2}\sum_{(v,u)\in E} \frac{p(v)}{d(v)} \\
&= \frac{1}{2}\sum_{(u,v)\in E} p(u,v) + \frac{1}{2}\sum_{(v,u)\in E} p(v,u) \\
&= \frac{1}{2}p(\mathrm{in}(u)) + \frac{1}{2}p(\mathrm{out}(u)).
\end{aligned}$$

The amount of probability on a set S can then be written this way.

$$\begin{aligned}
pW(S) &= p(\mathrm{in}(S)) + p(\mathrm{out}(S)) \\
&= p(\mathrm{in}(S) \cup \mathrm{out}(S)) + p(\mathrm{in}(S) \cap \mathrm{out}(S)).
\end{aligned}$$

□

Proof of Lemma 5. Let $p = \mathrm{apr}(\alpha, s, r)$ be an approximate PageRank vector. By Lemma 3 we have the inequality $p \leq \alpha s + (1-\alpha)pW$, which implies

$$\begin{aligned}
p(S) &\leq [\alpha s + (1-\alpha)pW](S) \\
&= \alpha s(S) + (1-\alpha)pW(S) \\
&\leq \alpha s(S) + (1-\alpha)\frac{1}{2}(p(\mathrm{in}(S) \cup \mathrm{out}(S)) + p(\mathrm{in}(S) \cap \mathrm{out}(S))).
\end{aligned}$$

This proves the first part of the lemma. To prove the second part, recall that $p\left[\text{vol}(S_j^p)\right] = p(S_j^p)$ for any integer $j \in [0, n]$. Also, for any set of directed edges A, we have the bound $p(A) \leq p\left[|A|\right]$. Therefore,

$$\begin{aligned} p\left[\text{vol}(S_j^p)\right] &= p(S_j^p) \\ &\leq \alpha s(S_j^p) + (1-\alpha)\frac{1}{2}\left(p\left(\text{in}(S_j^p) \cup \text{out}(S_j^p)\right) + p\left(\text{in}(S_j^p) \cap \text{out}(S_j^p)\right)\right) \\ &\leq \alpha s\left[\text{vol}(S_j^p)\right] + (1-\alpha)\frac{1}{2}\left(p\left[\left|\text{in}(S_j^p) \cup \text{out}(S_j^p)\right|\right] + p\left[\left|\text{in}(S_j^p) \cap \text{out}(S_j^p)\right|\right]\right). \end{aligned}$$

All that remains is to bound the sizes of the sets in the inequality above. Notice that

$$\left|\text{in}(S_j^p) \cup \text{out}(S_j^p)\right| + \left|\text{in}(S_j^p) \cap \text{out}(S_j^p)\right| = 2\text{vol}(S_j^p),$$

and

$$\left|\text{in}(S_j^p) \cup \text{out}(S_j^p)\right| - \left|\text{in}(S_j^p) \cap \text{out}(S_j^p)\right| = 2|\partial(S_j^p)|.$$

This implies that

$$\left|\text{in}(S_j^p) \cup \text{out}(S_j^p)\right| = \text{vol}(S_j^p) + |\partial(S_j^p)|,$$

and

$$\left|\text{in}(S_j^p) \cap \text{out}(S_j^p)\right| = \text{vol}(S_j^p) - |\partial(S_j^p)|.$$

The result follows. □

Proof of Theorem 3. Let $k_j = \text{vol}(S_j^p)$, let $\overline{k_j} = \min(k_j, 2m - k_j)$, and let

$$f_t(k) = \gamma + \alpha t + \sqrt{\min(k, 2m-k)}\left(1 - \frac{\phi^2}{8}\right)^t.$$

Assuming that there does not exist a sweep cut with all of the properties stated in the theorem, we will prove by induction that the following holds for all $t \geq 0$:

$$p\left[k\right] - \frac{k}{2m} \leq f_t(k), \quad \text{for any } k \in [0, 2m]. \tag{7}$$

For the base case, equation (7) holds for $t = 0$, with any choice of γ and ϕ. To see this, notice that for each integer $k \in [1, 2m-1]$,

$$p\left[k\right] - \frac{k}{2m} \leq 1 \leq \sqrt{\min(k, 2m-k)} \leq f_0(k).$$

For $k = 0$ and $k = 2m$ we have $p\left[k\right] - \frac{k}{2m} \leq 0 \leq f_0(k)$. The claim follows because f_0 is concave, $p\left[k\right]$ is less than f_0 for each integer value of k, and $p\left[k\right]$ is linear between these integer values.

Assume for the sake of induction that equation (7) holds for t. To prove that equation (7) holds for $t+1$, which will complete the proof of the theorem, it suffices to show that the following equation holds for each $j \in [1, |\text{Supp}(p)|]$:

$$p\left[k_j\right] - \frac{k_j}{2m} \leq f_{t+1}(k_j). \tag{8}$$

This equation holds trivially for $j = 0$, and $j = n$. The theorem will follow because f_{t+1} is concave, we have shown that equation (8) holds at k_j for each j in the set $[1, |\mathrm{Supp}(p)|] \cup \{0, n\}$, and $p\,[k]$ is linear between these points.

Consider an index $j \in [1, |\mathrm{Supp}(p)|]$. If property 2 does not hold for j, then this directly implies that equation (8) holds at j. If property 1 does not hold for j, then we have $\Phi(S_j^p) \geq \phi$, and Lemma 5 implies the following.

$$\begin{aligned} p\left[\mathrm{vol}(S_j^p)\right] &\leq \alpha s\left[\mathrm{vol}(S_j^p)\right] + (1-\alpha)\frac{1}{2}\left(p\left[\mathrm{vol}(S_j^p) - |\partial(S_j^p)|\right] + p\left[\mathrm{vol}(S_j^p) + |\partial(S_j^p)|\right]\right) \\ &\leq \alpha + \frac{1}{2}\left(p\left[\mathrm{vol}(S_j^p) - |\partial(S_j^p)|\right] + p\left[\mathrm{vol}(S_j^p) + |\partial(S_j^p)|\right]\right) \\ &= \alpha + \frac{1}{2}\left(p\left[k_j - \Phi(S_j^p)\overline{k_j}\right] + p\left[k_j + \Phi(S_j^p)\overline{k_j}\right]\right) \\ &\leq \alpha + \frac{1}{2}\left(p\left[k_j - \phi\overline{k_j}\right] + p\left[k_j + \phi\overline{k_j}\right]\right). \end{aligned}$$

The last step above follows from the concavity of $p\,[k]$. Using the induction hypothesis,

$$\begin{aligned} p\,[k_j] &\leq \alpha + \frac{1}{2}\left(f_t(k_j - \phi\overline{k_j}) + \frac{k_j - \phi\overline{k_j}}{2m} + f_t(k_j + \phi\overline{k_j}) + \frac{k_j + \phi\overline{k_j}}{2m}\right). \\ &= \alpha + \frac{k_j}{2m} + \frac{1}{2}\left(f_t(k_j - \phi\overline{k_j}) + f_t(k_j + \phi\overline{k_j})\right). \end{aligned}$$

Therefore,

$$\begin{aligned} p\,[k_j] - \frac{k_j}{2m} &\leq \alpha + \frac{1}{2}\left(f_t(k_j - \phi\overline{k_j}) + f_t(k_j + \phi\overline{k_j})\right) \\ &= \gamma + \alpha + \alpha t \\ &\quad + \frac{1}{2}\left(\sqrt{\min(k_j - \phi\overline{k_j}, 2m - k_j + \phi\overline{k_j})} + \sqrt{\min(k_j + \phi\overline{k_j}, 2m - k_j - \phi\overline{k_j})}\right)\left(1 - \frac{\phi^2}{8}\right)^t \\ &\leq \gamma + \alpha + \alpha t + \frac{1}{2}\left(\sqrt{\overline{k_j} - \phi\overline{k_j}} + \sqrt{\overline{k_j} + \phi\overline{k_j}}\right)\left(1 - \frac{\phi^2}{8}\right)^t. \end{aligned}$$

This last step can be verified by considering the two cases $k_j \leq m$ and $k_j \geq m$ separately.

By examining the Taylor series of $\sqrt{1+\phi}$ at $\phi = 0$, we obtain the following for any $k \geq 0$ and $\phi \in [0, 1]$.

$$\begin{aligned} \frac{1}{2}\left(\sqrt{k - \phi k} + \sqrt{k + \phi k}\right) &\leq \frac{\sqrt{k}}{2}\left((1 - \frac{\phi}{2} - \frac{\phi^2}{8}) + (1 + \frac{\phi}{2} - \frac{\phi^2}{8})\right) \\ &\leq \sqrt{k}\left(1 - \frac{\phi^2}{8}\right). \end{aligned}$$

By applying this with $k = \overline{k_j}$, we obtain

$$\begin{aligned} p\,[k_j] - \frac{k_j}{2m} &\leq \gamma + \alpha + \alpha t + \sqrt{\overline{k_j}}\left(1 - \frac{\phi^2}{8}\right)\left(1 - \frac{\phi^2}{8}\right)^t \\ &= f_{t+1}(k_j). \end{aligned}$$

□

Proof of Theorem 2. Let $\phi = \Phi(\mathrm{apr}(\alpha, s, r))$. Theorem 3 implies

$$\mathrm{apr}(\alpha, s, r)(S) - \frac{\mathrm{vol}(S)}{\mathrm{vol}(G)} \leq \alpha t + \sqrt{\min(\mathrm{vol}(S), 2m - \mathrm{vol}(S))} \left(1 - \frac{\phi^2}{8}\right)^t,$$

for any integer $t \geq 0$ and any $k \in [0, 2m]$. If we let $t = \lceil \frac{8}{\phi^2} \ln \frac{2\sqrt{m}}{\delta} \rceil$, then we have

$$\delta < \mathrm{apr}(\alpha, s, r)(S) - \frac{\mathrm{vol}(S)}{\mathrm{vol}(G)} \leq \alpha \lceil \frac{8}{\phi^2} \ln \frac{2\sqrt{m}}{\delta} \rceil + \frac{\delta}{2},$$

which implies

$$\frac{\delta}{2} < \alpha \lceil \frac{8}{\phi^2} \ln \frac{2\sqrt{m}}{\delta} \rceil \leq \alpha \frac{9}{\phi^2} \ln m.$$

The result follows by solving for ϕ. □

Proof of Lemma 6. We first prove the following monotonicity property for the PageRank operator: for any starting distribution s, and any $k \in [0, 2m]$,

$$\mathrm{pr}(\alpha, s)\,[k] \leq s\,[k]\,. \tag{9}$$

This is a consequence of Lemma 5; if we let $p = \mathrm{pr}(\alpha, s)$, then for each $j \in [1, n-1]$ we have

$$\begin{aligned} p\left[\mathrm{vol}(S_j^p)\right] &\leq \alpha s\left[\mathrm{vol}(S_j^p)\right] + (1-\alpha)\frac{1}{2}\left(p\left[\mathrm{vol}(S_j^p) - |\partial(S_j^p)|\right] + p\left[\mathrm{vol}(S_j^p) + |\partial(S_j^p)|\right]\right) \\ &\leq \alpha s\left[\mathrm{vol}(S_j^p)\right] + (1-\alpha)p\left[\mathrm{vol}(S_j^p)\right], \end{aligned}$$

where the last line follows from the concavity of $p\,[k]$. This implies that $\mathrm{pr}(\alpha, s)\,[k_j] \leq s\,[k_j]$, where $k_j = \mathrm{vol}(S_j^{\mathrm{pr}(\alpha,s)})$, for each $j \in [1, n-1]$. The result follows, since $s\,[k]$ is concave, and $\mathrm{pr}(\alpha, s)\,[k]$ is linear between the points where $k = k_j$.

The amount of probability that moves from C to $\bar{C}$ in the step from $\mathrm{pr}(\alpha, \psi_C)$ to $\mathrm{pr}(\alpha, \psi_C)W$ is bounded by $\frac{1}{2}\mathrm{pr}(\alpha, \psi_C)\,[|\partial(C)|]$, since $|\partial(C)|$ is the number of directed edges from C to $\bar{C}$. By the monotonicity property,

$$\begin{aligned} \mathrm{pr}(\alpha, \psi_C)\,[|\partial(C)|] &\leq \psi_C\,[|\partial(C)|] \\ &= \frac{|\partial(C)|}{\mathrm{vol}(C)} \\ &= \Phi(C). \end{aligned}$$

Using the recursive property of PageRank,

$$\begin{aligned} [\mathrm{pr}(\alpha, \psi_C)]\left(\bar{C}\right) &= [\alpha\psi_C + (1-\alpha)\mathrm{pr}(\alpha, \psi_C)W]\left(\bar{C}\right) \\ &\leq (1-\alpha)[\mathrm{pr}(\alpha, \psi_C)]\left(\bar{C}\right) + \frac{1}{2}\mathrm{pr}(\alpha, \psi_C)\,[|\partial(C)|] \\ &\leq (1-\alpha)[\mathrm{pr}(\alpha, \psi_C)]\left(\bar{C}\right) + \frac{1}{2}\Phi(C). \end{aligned}$$

This implies

$$[\mathrm{pr}(\alpha, \psi_C)](\bar{C}) \leq \frac{\Phi(C)}{2\alpha}.$$

□

Proof of Theorem 4. For a set $C \subseteq V$, let C_α be the set of vertices v in C satisfying

$$\mathrm{pr}(\alpha, \chi_v)\left(\bar{C}\right) \leq \frac{\Phi(C)}{\alpha}.$$

Let v be a vertex chosen randomly from the distribution ψ_C, and define the random variable $X = \mathrm{pr}(\alpha, \chi_v)\left(\bar{C}\right)$. The linearity property of PageRank vectors from Proposition 2, combined with the bound from Lemma 6, implies the following bound on the expectation of X.

$$\mathrm{E}\left[X\right] = \mathrm{pr}(\alpha, \psi_C)\left(\bar{C}\right) \leq \frac{\Phi(C)}{2\alpha}.$$

Then,

$$\Pr\left[v \notin C_\alpha\right] \leq \Pr\left[X > 2\mathrm{E}\left[X\right]\right] \leq \frac{1}{2}.$$

Since $\Pr\left[v \in C_\alpha\right] \geq \frac{1}{2}$, the volume of C_α is at least $\frac{1}{2}\mathrm{vol}(G)$.

If v is a vertex in C_α, we can obtain a lower bound for $\mathrm{apr}(\alpha, \chi_v, r)(C)$ by bounding the difference between $\mathrm{apr}(\alpha, \chi_v, r)$ and $\mathrm{pr}(\alpha, \chi_v)$ in terms of the residual vector r. Using the monotonicity property $\mathrm{pr}(\alpha, r)\left[k\right] \leq r\left[k\right]$ from equation (9), we have

$$\begin{aligned}
\mathrm{apr}(\alpha, \chi_v, r)(C) &= \mathrm{pr}(\alpha, \chi_v)(C) - \mathrm{pr}(\alpha, r)(C) \\
&\geq \mathrm{pr}(\alpha, \chi_v)(C) - \mathrm{pr}(\alpha, r)\left[\mathrm{vol}(C)\right] \\
&\geq \mathrm{pr}(\alpha, \chi_v)(C) - r\left[\mathrm{vol}(C)\right] \\
&\geq 1 - \frac{\Phi(C)}{\alpha} - \mathrm{vol}(C) \max_{u \in V} \frac{r(u)}{d(u)}.
\end{aligned}$$

□

Proof of Theorem 5. Theorem 4 gives a lower bound on $\mathrm{apr}(\alpha, \chi_v, r)(C)$.

$$\begin{aligned}
\mathrm{apr}(\alpha, \chi_v, r)\left(C\right) &\geq 1 - \frac{\Phi(C)}{\alpha} - \mathrm{vol}(C) \max_{u \in V} \frac{r(u)}{d(u)} \\
&\geq 1 - \frac{\Phi(C)}{\alpha} - \frac{1}{10}.
\end{aligned}$$

Since $\frac{\Phi(C)}{\alpha} \leq \frac{1}{10}$, we have $\mathrm{apr}(\alpha, \chi_v, r)(C) \geq \frac{4}{5}$, which implies

$$\mathrm{apr}(\alpha, \chi_v, r)(C) - \frac{\mathrm{vol}(C)}{\mathrm{vol}(G)} \geq \frac{4}{5} - \frac{2}{3} = \frac{2}{15}.$$

Theorem 2 then implies

$$\Phi(\mathrm{apr}(\alpha, s, r)) < \sqrt{135\alpha \ln m} = \phi.$$

□

We remark that it is possible to replace the term $\ln m$ in Theorem 5 with $\ln M$, where M is an upper bound on the volume of the set C. This can be done by setting α as a function of $\log M$ rather than $\log m$, and changing the value of t used in the proof of Theorem 2. Although the proof follows by similar methods, this would complicate the statement of the theorem. Similarly, the term $\ln m$ in Theorem 6 could be replaced with $\ln(\mathrm{vol}(C_{\mathrm{opt}}))$.

Proof of Theorem 7. An approximate PageRank vector $p = \mathrm{apr}(\alpha, \chi_v, r)$, with residual vector r satisfying $\max_{u \in V} \frac{r(u)}{d(u)} \leq \frac{2^{-b}}{48B}$, can be computed in time $O(2^b \frac{\log m}{\alpha})$ using `ApproximatePageRank`. By Theorem 1, we have $\mathrm{vol}(\mathrm{Supp}(p)) = O(2^b \frac{\log m}{\alpha})$. It is possible to check each of the conditions in step 4, for every set S_j^p with $j \in [1, |\mathrm{Supp}(p)|]$, in time

$$O(\mathrm{vol}(\mathrm{Supp}(\mathrm{apr}(\alpha, \chi_v, r))) \log n) = O(2^b \frac{\log^2 m}{\alpha}).$$

Therefore, the running time of `PageRank-Nibble` is

$$O(2^b \frac{\log^2 m}{\alpha}) = O(2^b \frac{\log^3 m}{\phi^2}).$$

□

Proof of Theorem 8. Consider the PageRank vector $\mathrm{pr}(\alpha, \chi_v)$. Since v is in C_α, and since $\frac{\Phi(C)}{\alpha} \leq \frac{1}{96B}$, we have

$$\begin{aligned} \mathrm{pr}(\alpha, \chi_v)\,[\mathrm{vol}(C)] - \frac{\mathrm{vol}(C)}{2m} &\geq (1 - \frac{\phi(C)}{\alpha}) - \frac{1}{2} \\ &\geq \frac{1}{2} - \frac{1}{96}. \end{aligned}$$

We have set α so that $\alpha t \leq 1/25$ when $t = \lceil \frac{8}{\phi^2} \ln(100\sqrt{m}) \rceil$, and with this choice of t we have

$$\alpha t + \sqrt{\min(\mathrm{vol}(C), 2m - \mathrm{vol}(C))} \left(1 - \frac{\phi^2}{8}\right)^t < \frac{1}{25} + \frac{1}{100}.$$

Since $\frac{1}{2} - \frac{1}{96} > \frac{5}{12} + \frac{1}{25} + \frac{1}{100}$, the following equation holds with $\gamma = \frac{5}{12}$.

$$\mathrm{pr}(\alpha, \chi_v)\,[\mathrm{vol}(C)] - \frac{\mathrm{vol}(C)}{2m} > \gamma + \alpha t + \sqrt{\min(\mathrm{vol}(C), 2m - \mathrm{vol}(C))} \left(1 - \frac{\phi^2}{8}\right)^t. \tag{10}$$

Let $B = \lceil \log_2 m \rceil$. For each integer b in $[1, B]$, let $\gamma_b = \gamma(\frac{9}{10} + \frac{1}{10}\frac{b}{B})$. Consider the smallest value of b in $[1, B]$ for which the following equation holds for some $k \leq 2^b$.

$$\mathrm{pr}(\alpha, \chi_v)\,[k] - \frac{k}{2m} > \gamma_b + \alpha t + \sqrt{\min(k, 2m - k)} \left(1 - \frac{\phi^2}{8}\right)^t, \quad \text{for some integer } t \geq 0. \tag{11}$$

Equation (10) shows that this equation holds with $b = B$ and $k = m$. Let b_0 be the smallest value of b for which this equation holds, and let k_0 be some value such that $k_0 \leq m$ and such that this equation holds with $b = b_0$ and $k = k_0$. Notice that $s^{b_0 - 1} < k_0 \leq s^{b_0}$, because if equation (11) holds for $b = b_0$ and $k = k_0$, it also holds for $b = b_0 - 1$ and k_0.

When `PageRank-Nibble` is run with $b = b_0$, the approximate PageRank vector $\mathrm{apr}(\alpha, \chi_v, r)$ computed by `PageRank-Nibble` has only a small amount of error on a set of volume k_0: the error is small

enough that the difference $\mathrm{pr}(\alpha, \chi_v)[k_0] - \mathrm{apr}(\alpha, \chi_v, r)[k_0]$ is less than $\gamma_b - \gamma_{b-1} = \frac{1}{10B}\gamma = \frac{1}{24B}$.

$$\begin{aligned}
\mathrm{apr}(\alpha, \chi_v, r)[k_0] &\geq \mathrm{pr}(\alpha, \chi_v)[k_0] - \max_{u \in V} \frac{r(u)}{d(u)} k_0 \\
&\geq \mathrm{pr}(\alpha, \chi_v)[k_0] - \frac{2^{-b_0}}{48B} k_0 \\
&\geq \mathrm{pr}(\alpha, \chi_v)[k_0] - \frac{1}{48B} \\
&\geq \mathrm{pr}(\alpha, \chi_v)[k_0] - (\gamma_{b_0} - \gamma_{b_0-1}) + \frac{1}{48B}.
\end{aligned}$$

We then use the lower bound on $\mathrm{pr}(\alpha, v)[k_0]$ implied by the definition of b_0: for some integer $t \geq 0$,

$$\begin{aligned}
\mathrm{apr}(\alpha, \chi_v, r)[k_0] - \frac{k_0}{2m} &> \left(\gamma_{b_0} + \alpha t + \sqrt{\min(k_0, 2m - k_0)} \left(1 - \frac{\phi^2}{8}\right)^t\right) - (\gamma_{b_0} - \gamma_{b_0-1}) + \frac{1}{48B} \\
&> (\gamma_{b_0-1} + \frac{1}{48B}) + \alpha t + \sqrt{\min(k_0, 2m - k_0)} \left(1 - \frac{\phi^2}{8}\right)^t.
\end{aligned}$$

Theorem 3 then shows that there exists a sweep cut S_j, with $S_j = S_j^{\mathrm{apr}(\alpha, \chi_v, r)}$ for some value of j in the range $[1, |\mathrm{Supp}(\mathrm{apr}(\alpha, \chi_v, r))|]$, such that $\Phi(S_j) \leq \phi$, and such that following lower bound holds for some integer t:

$$\mathrm{apr}(\alpha, \chi_v, r)(S_j) - \frac{\mathrm{vol}(S_j)}{2m} > (\gamma_{b_0-1} + \frac{1}{48B}) + \alpha t + \sqrt{\min(\mathrm{vol}(S_j), 2m - \mathrm{vol}(S_j))} \left(1 - \frac{\phi^2}{8}\right)^t. \quad (12)$$

We will show that this cut S_j satisfies all the requirements of `PageRank-Nibble`.

It must be true that $\mathrm{vol}(S_j) > 2^{b_0-1}$, since if were true that $\mathrm{vol}(S_j) \leq 2^{b_0-1}$, the definition of b_0 would imply that for any integer t,

$$\begin{aligned}
\mathrm{apr}(\alpha, \chi_v, r)(S_j) - \frac{\mathrm{vol}(S_j)}{2m} &= \mathrm{apr}(\alpha, \chi_v, r)[\mathrm{vol}(S_j)] - \frac{\mathrm{vol}(S_j)}{2m} \\
&\leq \mathrm{pr}(\alpha, s)[\mathrm{vol}(S_j)] - \frac{\mathrm{vol}(S_j)}{2m} \\
&\leq \gamma_{b_0-1} + \alpha t + \sqrt{\min(\mathrm{vol}(S_j), 2m - \mathrm{vol}(S_j))} \left(1 - \frac{\phi^2}{8}\right)^t,
\end{aligned}$$

and this would contradict the lower bound from equation (12).

It must also be true that $\mathrm{vol}(S_j) < \frac{2}{3}\mathrm{vol}(G)$. Otherwise, the lower bound from equation (12) would imply that for some integer t,

$$\begin{aligned}
\mathrm{apr}(\alpha, \chi_v, r)(S_j) &> \frac{\mathrm{vol}(S_j)}{2m} + \gamma_{b_0-1} + \alpha t + \sqrt{\min(\mathrm{vol}(S_j), 2m - \mathrm{vol}(S_j))} \left(1 - \frac{\phi^2}{8}\right)^t \\
&> \frac{2}{3} + \gamma_{b_0-1} \\
&\geq \frac{2}{3} + \frac{9}{10}\gamma.
\end{aligned}$$

Since $\gamma = \frac{5}{12}$, this implies $\mathrm{apr}(\alpha, \chi_v, r)(S_j) > 1$, which is impossible.

To prove that there is a significant difference between $\text{apr}(\alpha, \chi_v, r)\left[2^{b_0}\right]$ and $\text{apr}(\alpha, \chi_v, r)\left[2^{b_0-1}\right]$, observe that equation (12) does not hold with $b = b_0 - 1$ and $k = 2^{b_0-1}$. Therefore, for every integer $t \geq 0$,

$$\text{apr}(\alpha, \chi_v, r)\left[2^{b_0-1}\right] - \frac{k_0}{2m} \leq \gamma_{b_0-1} + \alpha t + \sqrt{\min(2^{b_0-1}, 2m - 2^{b_0-1})}\left(1 - \frac{\phi^2}{8}\right)^t. \tag{13}$$

We also know that for some integer t,

$$\text{apr}(\alpha, \chi_v, r)\left[k_0\right] - \frac{k_0}{2m} > (\gamma_{b_0-1} + \frac{1}{48B}) + \alpha t + \sqrt{\min(k_0, 2m - k_0)}\left(1 - \frac{\phi^2}{8}\right)^t. \tag{14}$$

Since $2^{b_0-1} \leq k_0 \leq m$, we have $\sqrt{\min(s^{b_0-1}, 2m - s^{b_0-1})} \leq \sqrt{\min(k_0, 2m - k_0)}$. Taking an integer t that makes equation (14) true, and plugging this value of t into equations (14) and (13), yields the following inequality.

$$\begin{aligned} \text{apr}(\alpha, \chi_v, r)\left[2^{b_0}\right] - \text{apr}(\alpha, \chi_v, r)\left[2^{b_0-1}\right] &\geq \text{apr}(\alpha, \chi_v, r)\left[k_0\right] - \text{apr}(\alpha, \chi_v, r)\left[2^{b_0-1}\right] \\ &> \frac{1}{48B}. \end{aligned}$$

We have shown that S_j meets all the requirements of `PageRank-Nibble`, which proves that the algorithm outputs some cut when run with $b = b_0$. We now prove a lower bound on $\text{vol}(S \cap C)$, which holds for any cut S output by `PageRank-Nibble`, with any value of b. Let $p'\left[k\right] = p\left[k\right] - p\left[k-1\right]$. Since $p'\left[k\right]$ is a decreasing function of k,

$$\begin{aligned} p'\left[2^{b-1}\right] &\geq \frac{p\left[2^b\right] - p\left[2^{b-1}\right]}{2^b - 2^{b-1}} \\ &> \frac{1}{2^{(b-1)}48B}. \end{aligned}$$

It is not hard to see that combining this lower bound on $p'\left[2^{b-1}\right]$ with the upper bound $p\left(\bar{C}\right) \leq \frac{\Phi(C)}{\alpha}$ gives the following bound on the volume of the intersection.

$$\begin{aligned} \text{vol}(S_j \cap C) &\geq 2^{b-1} - \frac{p\left(\bar{C}\right)}{p'\left[2^{b-1}\right]} \\ &> 2^{b-1} - 2^{b-1}(48B\frac{\Phi(C)}{\alpha}). \end{aligned}$$

Since we have assumed that $\frac{\Phi(C)}{\alpha} \leq \frac{1}{96B}$, we have

$$\text{vol}(S \cap C) > 2^{b-1} - 2^{b-2} = 2^{b-2}.$$

□