Approximating the Exponential, the Lanczos Method and an $\tilde{O}(m)$-Time Spectral Algorithm for Balanced Separator

Lorenzo Orecchia, MIT *Sushant Sachdeva*, Princeton University *Nisheeth Vishnoi*, MSR India

Talk on Tuesday at 2:25 pm, Session 13A

1 BALANCED GRAPH PARTITIONING AND SPECTRAL METHODS

For an undirected unweighted instance graph $G=(V,E)$ with $|V|=n$ and $|E|=m$, the conductance of a cut $S \subseteq V$ is defined as

$$\phi(S) = \frac{|E(S,\bar{S})|}{\min\{\mathrm{vol}(S), \mathrm{vol}(\bar{S})\}}$$

where $\mathrm{vol}(S)$ is the total degree in set S. A cut S is b-balanced if $\mathrm{vol}(S) \geq b \cdot \mathrm{vol}(V)$.

b-BALANCED CUT PROBLEM: Given graph G, parameter $\gamma \in (0,1)$ and balance $b \in (0,1/2)$, does G contain a b-balanced cut of conductance at most γ?
This problem is NP-hard. However, approximation algorithms exist:

Algorithm	Method	Distinguishes $\geq \gamma$ and	Running Time
Iterative Eigenvector	Spectral	$O(\sqrt{\gamma})$	$\tilde{O}(n^2)$
[Leighton, Rao `88]	Flow	$O(\gamma \log n)$	$\tilde{O}(m^{\frac{3}{2}})$ [AK `07, OSVV `08]
[Arora, Rao, Vazirani `04]	SDP (Flow + Spectral)	$O(\gamma\sqrt{\log n})$	$\tilde{O}(m^{\frac{3}{2}})$ [Sherman `09]
[Madry `10]	SDP (Flow + Spectral)	$\gamma \mathrm{polylog}(n)$	$\tilde{O}(m^{1+\epsilon})$

We use "spectral methods" to refer to algorithms that explore the graph by performing matrix-vector multiplications involving the graph Laplacian L. Such algorithms detect low-conductance cuts by exploiting the connection between the mixing of random walks in the graph and the cut structure of G.

Spectral methods for finding balanced cuts perform well in many applications and have fast running times and optimized implementations that make them a popular choice among practitioners. The following are the theoretical guarantees of some spectral algorithms for this problem.

Algorithm	Method	Distinguishes $\geq \gamma$ and	Time
Iterative Eigenvector	Eigenvector	$O(\sqrt{\gamma})$	$\tilde{O}(n^2)$
[Andersen, Peres `09]	Local Random Walks	$O\left(\sqrt{\gamma \log n}\right)$	$\tilde{O}(m/\sqrt{\gamma})$
[Orecchia, Vishnoi `10]	SDP	$O(\sqrt{\gamma})$	$\tilde{O}(m/\gamma)$

OUR THEOREM: We give an algorithm that either outputs an $\Omega(b)$-balanced cut $S \subset V$ such that $\phi(S) \leq O(\sqrt{\gamma})$, or outputs a certificate that no b-balanced cut of conductance γ exists. The algorithm runs in time $O(m\ \mathrm{poly}(\log n))$.

TECHNICAL COMPONENTS:

1) SDP primal-dual iterative algorithm with a simple random walk interpretation
2) Novel analysis of Lanczos methods for computing heat-kernel vectors

2 DETECTING AND REMOVING UNBALANCED CUTS

Unbalanced cuts of low conductance are obstacles to detecting balanced cuts.

Spectral methods are targeted towards finding low-conductance cuts, regardless of how balanced they are. For this reason, spectral methods may have to find and remove all unbalanced cuts of low conductance before finding a balanced cut.

The algorithmic challenge is to detect and remove unbalanced cuts of low conductance quickly.

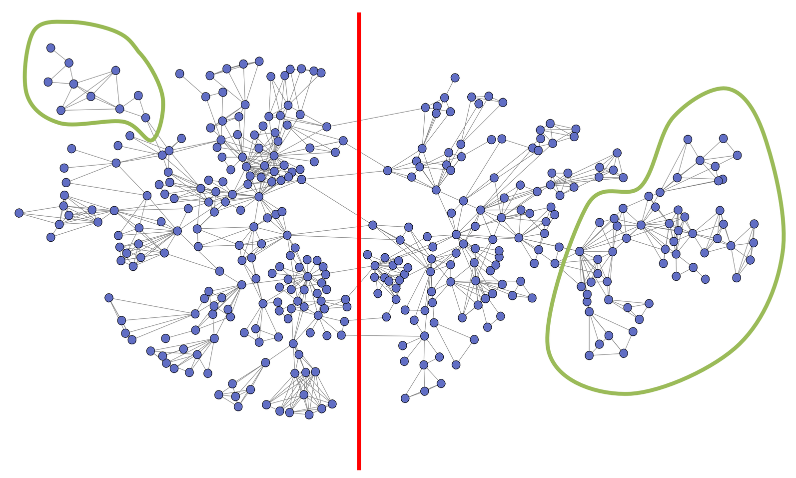


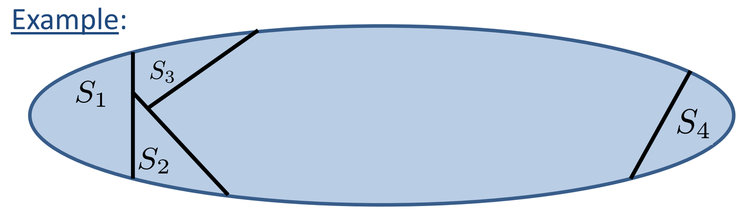
Figure: Balanced cuts and unbalanced cuts, both of low conductance, in a citation graph.

Iterative Eigenvector Approach

Set $G_1 = G$. For $t=1, \ldots, n$ do:

- Compute the slowest mixing eigenvector of G_t and corresponding Laplacian eigenvalue λ_2.
- If $\lambda_2 \geq \gamma$, output NO. Otherwise, sweep eigenvector to find cut S_t such that $\phi(S_t) \cdot\ O(\sqrt{\gamma})$.
- If $\cup S_t$ is $(b/2)$-balanced, output $\cup S_t$.
- Otherwise, let G_{t+1} be the graph induced by G_t on V-S_t with self-loops replacing the edges going to S_t. Recurse on G_{t+1}.

Example:

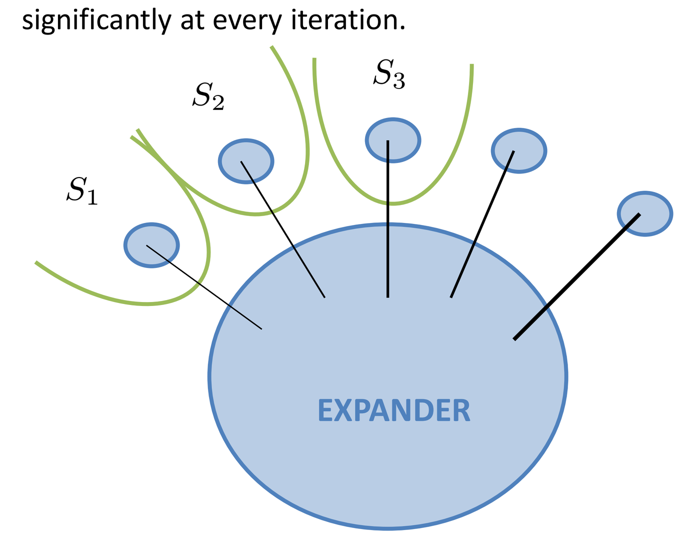


$\lambda_2(G_5) \geq \gamma$

Quadratic Running Time

Worst-case instance: - (n) iterations.
Each iteration takes time - (n), yielding a total running time of - (n).

NB: It is not possible to argue that $\lambda_2(G_t)$ grows significantly at every iteration.



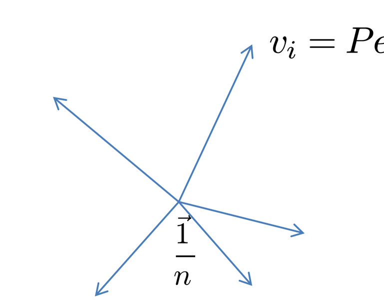
3 OUR RANDOM-WALK APPROACH FOR FINDING UNBALANCED CUTS

The eigenvector is unstable, as it may capture only a small cut and be oblivious to rest of the graph. Hence, we switch to a more stable distribution over low eigenvectors, represented by the transition matrix of a random walk that has converged to have most of its norm over eigenvectors with eigenvalue at most O(γ).
For simplicity, take G to be regular. The graph Laplacian is $L = I\text{-}W$.

IDEA BEHIND OUR ALGORITHM:
Replace eigenvector by multidimensional embedding of heat-kernel random-walk.

$$P_t = e^{-\tau L(G_t)} \qquad \tau = \log n/\gamma$$

Denote the embedding $\{v_i\}$ given by $v_i = Pe_i$ with $P = P_t$.



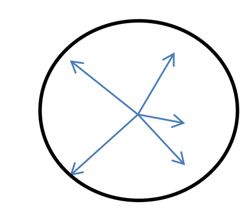
MIXING:
Define the deviation from stationary for a set $S \subseteq V$ for process P:

$$\Psi(P,S) = \sum_{i \in S} ||v_i - \vec{1}/n||^2$$

$\Psi(P,V)$ measures the total deviation from stationary, which is also the variance of the geometric embedding $\{v_i\}$.

ROUNDING. Three possible cases:

WALK HAS MIXED



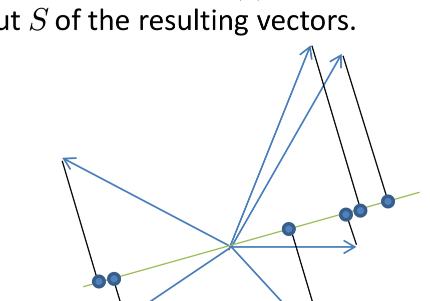
All vectors are short.
The total deviation is:

$$\Psi(P_t, V) \cdot \ \frac{1}{\mathrm{poly}(n)}$$

This yields a certificate that $\lambda_2(G_t) \geq \gamma$,
Which we turn into a NO certificate.

WALK HAS NOT MIXED AND BALANCED CUT IS FOUND

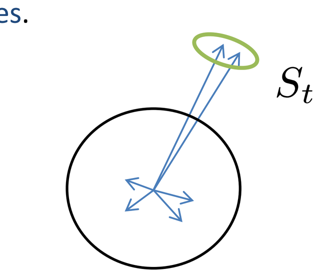
We take $O(\log n)$ random projections and find the best $\Omega(b)$-balanced sweep cut S of the resulting vectors.



If $\phi(S) \leq O(\sqrt{\gamma})$, we output S and answer YES.

WALK HAS NOT MIXED AND NO BALANCED CUT IS FOUND

Some vectors must be very long, i.e. walk converges very poorly from some vertices.



We can find S_t with $\phi(S_t) \leq O(\sqrt{\gamma})$.
Moreover,

$$\Psi(P_t, S_t) \geq \tfrac{1}{2}\Psi(P_t, V).$$

4 HOW WE UPDATE THE GRAPH AND OUR MIXING ANALYSIS

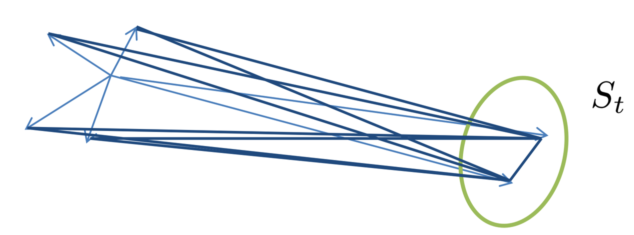
If a low-conductance balanced cut is found, we perform a soft removal of this cut by modifying the current random walk as follows:

$$P_{t+1} = e^{-\tau\left(L + \gamma \sum_{j=1}^{t-1} \sum_{i \in S_t} L(\mathrm{Star}_i)\right)}$$

where Star_i is the star graph rooted at vertex i.

The transition rate from S_t to all vertices increases, making the process converge faster to its stationary.

Increased Transition Rates



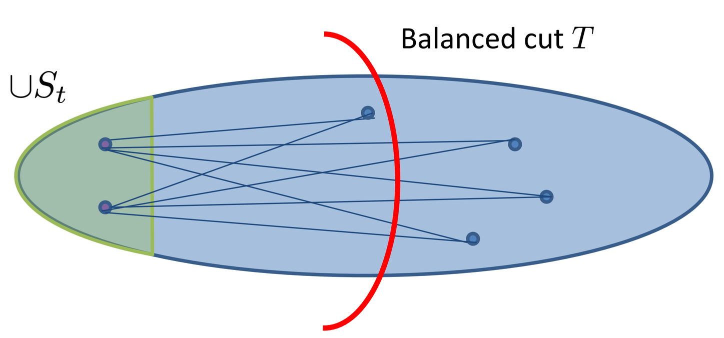
MIXING ANALYSIS: Using properties of the heat-kernel random walk, we show that the mixing improves significantly:

$$\Psi(P_{t+1}, V) \cdot \ \Psi(P_t, V) - \tfrac{1}{2}\Psi(P_t, S_t) \cdot \ \tfrac{3}{4}\Psi(P_t, V)$$

After $T = O(\log n)$ iterations, if no low-conductance $\Omega(b)$-balanced cut is found, the following holds:

$$\Psi(P_T, V) \cdot \ \frac{1}{\mathrm{poly}(n)}$$

We show that we can turn this fact into a NO certificate for the b-Balanced Cut problem:



$$\phi(T) \geq \gamma - \gamma\frac{|\cup S_t|}{|T|} \geq \gamma - \gamma\frac{b/2}{b} \geq \frac{\gamma}{2}$$

5 HOW TO COMPUTE OUR RANDOM-WALK VECTORS

GOAL: For symmetric diagonally-dominant A and vector u, with sparsity m, compute vector v such that

$$||e^{-A}u - v|| \cdot \ \delta \quad \text{in time} \quad \tilde{O}(m \cdot \mathrm{polylog}(||A||) \cdot \mathrm{polylog}(1/\delta))$$

In our case, $A = \frac{\log n}{\gamma} L$, so that $||A|| = \tilde{O}(\frac{1}{\gamma})$ and the running time is $\tilde{O}(m)$.

ITERATIVE APPROACHES:

- Taylor Series Approximation: requires $\Theta(||A|| \log(1/\delta))$ terms and yields a running time of $\tilde{O}(m/\gamma)$.
- Direct Lanczos Method: requires $\Theta(\sqrt{||A||}\mathrm{polylog}(1/\delta))$ iterations . The running time is $\tilde{O}(m/\sqrt{\gamma})$.
- **Our Algorithm** relies on a linear-system solver for A. We obtain the following running times:
  - Using Spielman-Teng solver: $\tilde{O}((m+n) \cdot \log(2+||A||) \cdot \mathrm{polylog}(1/\delta))$, yielding $\tilde{O}(m)$
  - Using Conjugate Gradient: $\tilde{O}((m+n) \cdot \sqrt{1+||A||} \cdot \log(2+||A||) \cdot \mathrm{polylog}(1/\delta))$, yielding $\tilde{O}(m/\sqrt{\gamma})$

OUR APPROACH exploits the speed of the linear-system solver by using it to perform the inverse iteration. We speed up convergence by applying Lanczos method.

Review of Lanczos Method

IDEA: Perform k matrix-vector multiplications to obtain the subspace

$$R_k = \mathrm{Span}\{u, Au, A^2u, A^3u, A^4u, \ldots, A^ku\}$$

Compute an orthonormal basis Q_k for R_k and let T_k be A restricted to R_k:

$$T_k = Q_k A Q_k^T$$

As k grows, T_k becomes a better approximation to A. If a function f is close to a polynomial p of degree k, i.e.

$$||p(x) - f(x)|| \cdot \ \delta \ \ \forall x \in \mathrm{Spectrum}(A)$$

then

$$||Q_k f(T_k) Q_k^T u - f(A)u|| \cdot \ 2\delta||u||$$

6 APPLYING LANCZOS METHOD TO THE INVERSE ITERATION

A direct application of Lanczos method does not meet our goal, but applying Lanczos to the inverse iteration yield our result.

DIRECT LANCZOS: There exists a polynomial p such that

$$\sup_{x \in (0,||A||)} |p(x) - e^{-x}| \cdot \ \frac{1}{\mathrm{poly}(n)}$$

and p has degree $\tilde{O}(\sqrt{||A||})$.

IMPLICATION: $\tilde{O}(\sqrt{||A||}) = \tilde{O}(1/\sqrt{\gamma})$ iterations are sufficient.

LOWER BOUND: This bound is tight: $\tilde{O}(\sqrt{||A||})$ iterations may be necessary.

IDEA: Apply Lanczos Method to $B = (I + \frac{A}{k})^{-1}$, i.e. compute subspace

$$R_k = \mathrm{Span}\{u, Bu, B^2u, B^3u, B^4u, \ldots, B^ku\}$$

NB: We can use the linear-system solvers to compute, for any vector x,

$$Bx = (I + \tfrac{A}{k})^{-1}x$$

APPROXIMATION: In fact, the quality of the k-approximation depends on the existence of a good approximation to the exponential function by rational functions, a larger class of functions than polynomials.

THEOREM (Saff, Schonhage, Varga `75) : There exists a polynomial p such that

$$\sup_{x \in (0,||A||)} \left| \frac{p(x)}{(1+x/k)^k} - e^{-x} \right| \cdot \ \delta$$

and p has degree $\tilde{O}(\log(2+||A||) \cdot \mathrm{polylog}(1/\delta))$.

STRONG ERROR GUARANTEE