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

$$\phi(S) = \frac{\text{cut}(S)}{\sqrt{\text{vol}(S) \cdot \text{vol}(\bar{S})}}$$

where $\text{cut}(S)$ is the total degree in $S$, $\text{vol}(S)$ is the size of $S$, and $\text{vol}(\bar{S})$ is the size of the complement of the cut.

2. DETECTING AND REMOVING UNBALANCED CUTS

Unbalanced cuts of low conductance are obtained by iteratively finding 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 cycles of low conductance before finding a balanced cut.

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

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 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 $\lambda_2$.

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

$P_t = e^{-tL(G)}$

Denote the embedding of $v_i$ given by $v_i = P_t v_i$ with $P_t = P_t(v_i)$.

MIXING:
Define the deviation from stationary for a set $S_i$ for process $P$: $\Phi(P_t, S_i) = \sum_{(u,v) \in E} |v_t(u) - v_t(v)|$.

$\Phi(P_t)$ measures the total deviation from stationary, which is also the variance of the geometric embedding $v_i$.

BOUNDING:
Three possible cases:

- WALK HAS MIXED:
All vectors are short.
The total deviation is:

$$\Phi(P_t, S_i) \leq O(\sqrt{\text{vol}(S_i)})$$

This yields a certificate that $S_i$ is not cut-

- WALK HAS NOT MIXED AND NO BALANCED CUT IS FOUND:
Some vectors must be very long, i.e.,

$$\Phi(P_t, S_i) \geq \Omega(\sqrt{\text{vol}(S_i)})$$

We can find $S_i$ with $O(\text{vol}(S_i))$ queries.

- WALK HAS NOT MIXED AND BALANCED CUT IS FOUND:

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:

$$P_{t+1} = \alpha P_t + (1-\alpha)\frac{1}{d_i}e_i e_i^T$$

where $\alpha$ is the softness and $d_i$ is the degree of the node.

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

5. HOW TO COMPUTE OUR RANDOM-WALK VECTORS

**Goal:** For symmetric diagonally-dominant $A$ and vector $v_i$, with sparsity $n$, compute vector $v_i$ such that:

$$\|A^{\frac{n}{2}}v_i - \bar{v}_i\| \leq \delta$$

where $|| \cdot || = \text{L}1$ norm and the running time is $O(m\log n)$.

**Iterative Approach:**

- **Taylor Series Approximation:**
  - Requires $O(\sqrt{\log n})$ terms and yields a running time of $O(m \cdot \log n)$.
  - **Direct Lanczos Method:**
    - Requires $O(\sqrt{\log n})$ terms and a running time of $O(m \cdot \log n)$.
    - Our Algorithm relies on a linear system solver for $A$.

**Our Approach** exploits the speed of the linear system solver by using it to perform the reverse iteration. We speed up convergence by applying Lanczos method.

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

$$\sup_{x \in [0,1]}|p(x) - e^{-x}| = \frac{1}{\text{poly}(n)}$$

and $p$ has degree $O(\sqrt{\log n})$.

**Implication:** $\tilde{O}(\sqrt{\log n})$ iterations are sufficient.

**Lower Bound:** This bound is tight: $O(\sqrt{\log n})$ iterations may be necessary.

**SIEA:** Apply Lanczos Method to $B = (I + \frac{1}{x})^{-1}$, i.e. compute subspace $R_n = \text{Span}(x, Bx, B^2x, B^3x, \ldots, B^nx)$.

**NB:** We can use the linear-system solvers to compute, for any vector $x$, $Bx = e$ ($I + \frac{1}{x} = 0$).

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

**Theorem:** ([Galat, Schroeppers, Varga 75]) There exists a polynomial $p$ such that

$$\sup_{x \in [0,1]}|p(x) - e^{-x}| = \frac{1}{\text{poly}(n)}$$

and $p$ has degree $O(\log(2 + |\lambda_2|) \cdot \log(\|\lambda_2\|))$.

**Strong Error Guarantee**