GRAPH PARTITIONING USING SINGLE COMMODITY FLOWS [KRV'06]

notes by Petar MAYMOUNKOV

Theme — The algorithmic problem of finding a sparsest cut is related to the combinatorial problem of building expander graphs from "simple" building blocks!

1. Preliminaries

We consider an undirected graph G defined on $V = [n]$ and edge-weighted by $w(e) \geqslant 0$. For $S \subseteq V$ we use ∂S to be the set of edges with exactly one endpoint in S. Define $w(\partial S) = \sum_{e \in \partial S} w(e)$.

Definition 1.1. — *The expansion of G is defined as*

$$\Phi(G) = \min_{S \subset V} \frac{w(\partial S)}{\min\{|S|, |S^c|\}} \tag{1.1}$$

Definition 1.2. — *A single commodity flow ω between $s \neq t \in V$ can be viewed as $\omega \in \mathbb{R}^{V \times V}$, having the following properties:*

(i) *Skew symmetry:*

$$\omega_{xy} = -\omega_{yx}, \text{ for all } x, y \in V$$

(ii) *Capacity*

$$\omega_{xy} \leqslant w(x, y), \text{ for all } x, y \in V$$

(iii) *Conservation:*

$$\sum_y \omega_{xy} = 0, \text{ for all } s, t \neq x, y \in V$$

(iv) *Demand:*

$$\sum_y \omega_{sy} = d, \text{ which implies } \sum_y \omega_{ty} = -d$$

LEMMA 1.3 (Concentration of measure). — *If $v \in \mathbb{R}^d$ and ζ is a random unit vector in $\mathbb{R}^d$, then*

$$\mathbf{E}\,(v^*\zeta)^2 = \frac{\|v\|^2}{d}, \text{ and}$$
$$\mathbf{Pr}\Big[(v^*\zeta)^2 \geqslant x\|v\|^2/d\Big] \leqslant e^{-x/4}, \text{ for } x \leqslant d/16.$$

2. Embedding graphs by flow

LEMMA 2.1. — *If $H \overset{\xi}{\mapsto} G$, then $\Phi(H) \leqslant C_\xi \Phi(G)$.*

Proof. — By assumption H routes in $C_\xi \cdot G$ with no congestion. Consider any cut (S, S^c). If $h \in E(H)$ is cut, then it contributes $w_H(h)$ to $w_H(\partial S)$ and at least $w_H(h)$ to $w_{C_\xi G}(\partial S)$. Thus $w_H(\partial S) \leqslant w_{C_\xi G}(\partial S)$, and

$$\Phi(H) = \min_S \frac{w_H(\partial S)}{\min\{|S|, |S^c|\}} \leqslant \min_S \frac{w_{C_\xi G}(\partial S)}{\min\{|S|, |S^c|\}} = \Phi(C_\xi G) = C_\xi \Phi(G).$$

∎

3. Main result

THEOREM 3.1. — *Given an undirected graph G and an expansion parameter $\alpha > 0$, there is a randomized algorithm that w.h.p. outputs*

(i) *Either a cut (S, S^c) with $\Phi(S, S^c) \leqslant \alpha$,*

(ii) *or an expander H and an embeding $H \overset{\xi}{\mapsto} G$ with congestion at most $O(\log^2 n/\alpha)$*

ALGORITHM 3.1 (Main algorithm – conceptually). — *On input G and α:*

> *Build an expander graph H as the union of simple graphs $M_1, \ldots, M_t$. Try to embed each M_i in G with congestion $\leqslant 1/\alpha$. If success, then $\Phi(G) \geqslant \alpha/t$, otherwise an M_i fails to embed, which uncovers a sparse cut.*

4. Constructing expanding graphs incrementally

For a perfect matching μ, define the 1-step random walk $W_\mu := 1/2\cdot(I+\mu)$, where μ is viewed as an adjacency matrix. Let $(M_1, \ldots, M_t)$ be a sequence of perfect matchings, and define the natural *matching walk* as the t-step walk with transition matrix $W = W_{M_t} \cdots W_{M_1}$.

Definition 4.1. — *A matching walk is mixing iff* $\mathbf{1}_x^* W \mathbf{1}_j \geqslant 1/2n$ *for all* $x, y \in [n]$.

Lemma 4.2 (Matching walk mixes — matching union expands). — *If* $(M_1, \ldots, M_t)$ *mixes, then* $\Phi(\bigcup_{i=1}^t M_i) \geqslant 1/2$.

Proof. — Consider the directed "time-line" graph H defined on $\{1, \ldots, n\} \times \{0, \ldots, t\}$, where for all $i \in \{1, \ldots, n\}$ and $k \in \{0, \ldots, t-1\}$

(i) $(i, j) \overset{H}{\sim} (i, j+1)$, and

(ii) $(i, j) \overset{H}{\sim} (i', j+1)$ where i and i' are matched by M_{j+1}.

Every edge in H is given capacity $1/2$. The matching walk from $x \in [n]$ induces a flow of value 1 on H from $(x, 0)$ to $(1, t), \ldots, (n, t)$. If the walk mixes, the flow delivers at least $1/2n$ units to each $(1, t), \ldots, (n, t)$.

It is easy to see (using induction) that the simultaneous unit flows from $(1, 0), \ldots, (n, 0)$ do not violate the edge capacities of H. Observe that the vertex-projection of H onto $\{1, \ldots, n\}$ equals $\bigcup_{i=1}^t M_i$, and thus the simultaneous flow is realizable there as well.

For any (S, S^c) cut in $[n]$ where $S \leqslant n/2$, this flow delivers at least $|S||n - S|/2n$ units to S^c. From here, the Min-cut Max-flow Theorem asserts $\Phi(\bigcup_{i=1}^t M_i) \geqslant 1/2$. ∎

We are going to use a *potential* function $\psi(W)$ to measure how far W is from mixing. For now $\psi(W)$ is abstract.

5. Idea

I am betting that G has expansion $\geqslant \alpha$:

(a) *If I am right, i.e. $\alpha \leqslant \Phi(G)$, I can build an expander as the union of $O(\log^2 n)$ matchings so that each embeds in G with congestion $\leqslant 1/\alpha$. Thus, proving that G's expansion is at least $O(\alpha/\log^2 n)$*

(b) *If I am wrong and $\Phi(G) \leqslant \alpha \leqslant O(\log^2 n \cdot \Phi(G))$, I might be just lucky enough that the matchings I need happen to be embedable in G with congestion $\leqslant 1/\alpha$ (i.e. I never happen to need a matching across the sparsest cut), in which case I still prove that $O(\alpha/\log^2 n) \leqslant \Phi(G)$*

(c) *However, if I am significantly off, i.e. $O(\log^2 n \cdot \Phi(G)) \leqslant \alpha$, then I am bound to run into a cut sparser than α in my attempt to realize some needed matching in G*

6. Main algorithm

Here $\psi(t) := \psi(W_{M_t} \cdots W_{M_1})$.

ALGORITHM 6.1 (Main). — *On input G and α:*

1. *(Find-Bisection) Find a bisection (S, S^c) such that adding any perfect matching M_{t+1} between S and S^c to $\{M_1, \dots, M_t\}$ reduces the potential, in expectation, by $1 - \Theta(1/\log n)$, i.e.*

$$\mathbf{E}\psi(t+1) \leqslant (1 - \Theta(1/\log n))\psi(t)$$

2. *Using a maximum-flow procedure*
   (a) *Either, produce a perfect matching M_{t+1} that embeds in G with congestion $\leqslant 1/\alpha$,*
   (b) *Or, find a cut in G of expansion at most α*

6.0.1. What lies ahead

We'll define a potential so if $\psi \leqslant O(1/n^2)$ then W is mixing, and thus the algorithm terminates in at most $O(\log^2 n)$ iterations w.h.p. thereby producing an embedding $(M_1, \dots, M_{O(\log^2 n)}) \mapsto G$ with congestion $O(\log^2 n)/\alpha$.

7. Find next best matching

Note, *Find-Bisection* has nothing to do with G. It's an algorithm about matchings and their mixing properties.

ALGORITHM 7.1 (*Find-Bisection*). — *On input $\{M_1, \dots, M_t\}$, output a bisection (S, S^c) so that the addition of any matching M_{t+1} between S and S^c brings the matching walk "significantly" closer to mixing:*

— *Choose $\zeta \in \{\pm 1\}^n$ randomly, so $\zeta \perp \mathbf{1}$*
— *Compute $u = W_t \dots W_1 \zeta$, where $W_i = 1/2 \cdot (I + M_i)$*
— *Form S from the first $n/2$ smallest entries of u*

8. Single step random walks and mixing

The main object of study here is a *positive* and *doubly stochastic* matrix $P \in \mathbb{R}^{n \times n}$ which encodes a random walk step on $[n]$.

DEFINITION 8.1. — *The 1-step walk P is mixing iff $P_{xy} \geqslant 1/2n$ for all x, y.*

DEFINITION 8.2. — *Define the potential of P as*

$$\psi(P) := \|P - \mathbf{J}/n\|^2 \quad (8.1)$$

Remark 8.3. — The potential of P is intended to measure (upper-bound) the ℓ_2 distance between "a 1-step random walk on P starting from a uniform distribution" and "the uniform distribution", i.e.

$$\psi'(P) := \|P\mathbf{1}/n - \mathbf{1}/n\|^2 \tag{8.2}$$

This intention is justified by:

LEMMA 8.4. — $\psi(P) \geqslant n\psi'(P)$

Proof. —

$$\begin{aligned}
\psi'(P) &= \|P\mathbf{1}/n - \mathbf{1}/n\|^2 \\
&= \|P\mathbf{1}/n - \mathbf{J}\mathbf{1}/n^2\|^2 && \text{(where } \mathbf{J} = \mathbf{1}\mathbf{1}^*\text{)} \\
&= 1/n^2 \cdot \|(P - \mathbf{J}/n)\mathbf{1}\|^2 \\
&\leqslant 1/n\|P - \mathbf{J}/n\|^2 && \text{(Cauchy-Schwarz or Frobenius norm)} \\
&= \psi(P)/n
\end{aligned}$$

∎

When $\psi(P)$ is sufficiently small, the 1-step walk is mixing:

LEMMA 8.5. — *If $\psi(P) \leqslant 1/4n^2$ then $P_{xy} \geqslant 1/2n$ for all $x, y \in V$.*

9. Averaging transformation and matchings

All matchings in this text are perfect. For any self-inverse permutation (or a matching, in particular) on $[n]$ characterized by its matrix $\mu \in \mathbb{R}^{n\times n}$, consider the averaging transformation $W_\mu = 1/2 \cdot (I + \mu)$ applied to P.

Remark 9.1. — Note that $W_\mu P$ represents the random walk where the first step is taken according to P and the second according to W_μ.

LEMMA 9.2. — *The composition $W_\mu P$ has the following properties:*

(i) $\mathbf{1}_x^*(W_\mu P) = \mathbf{1}_y^*(W_\mu P)$ *for all* $x \overset{\mu}{\sim} y$,
(ii) $(\mathbf{1}_x + \mathbf{1}_y)^*(W_\mu P)\mathbf{1} = (\mathbf{1}_x + \mathbf{1}_y)^* P\mathbf{1}$, *and*
(iii) $W_\mu P$ *is positive and doubly stochastic.*

LEMMA 9.3. —

$$\psi(P) - \psi(W_\mu P) \geqslant \frac{1}{2} \sum_{x \overset{\mu}{\sim} y} \|(\mathbf{1}_x - \mathbf{1}_y)^* P\|^2$$

Proof. — Focus on contribution of $x \overset{\mu}{\sim} y$. First,

$$\mathbf{1}_x^* W_\mu P = \mathbf{1}_y^* W_\mu P = \frac{1}{2}(\mathbf{1}_x + \mathbf{1}_y)^* P$$

Write $\psi(P)$ as

$$\psi(P) = \sum_x \|\mathbf{1}_x^* P - \mathbf{1}/n\|^2$$

Use the Parallelogram identity

$$\|f\|^2 + \|g\|^2 = \frac{1}{2}\|f+g\|^2 + \frac{1}{2}\|f-g\|^2$$

to compute the contribution of x and y to $\psi(W_\mu P) - \psi(P)$ as

$$2\|1/2 \cdot (\mathbf{1}_x + \mathbf{1}_y)^* P - \mathbf{1}/n\|^2 - \|\mathbf{1}_x^* P - \mathbf{1}/n\|^2 - \|\mathbf{1}_y^* P - \mathbf{1}/n\|^2 = \|(\mathbf{1}_x - \mathbf{1}_y)^* P\|^2$$

∎

10. Approximating potential reduction

For a random $\zeta \in \{\pm 1\}^n$:

LEMMA 10.1. — *With high probability, for all x, y,*

$$\|(\mathbf{1}_x - \mathbf{1}_y)^* P\|^2 \geqslant \frac{n-1}{O(\log n)} \cdot |(\mathbf{1}_x - \mathbf{1}_y)^* P\zeta| \tag{10.1}$$

Proof. — Use $(\mathbf{1}_x - \mathbf{1}_y)^* P \perp \mathbf{1}$ and $\zeta \perp \mathbf{1}$ and concentration of measure lemma. ∎

COROLLARY 10.2. — *With high probability,*

$$\psi(P) - \psi(W_\mu P) \geqslant \frac{n-1}{O(\log n)} \cdot \sum_{x \overset{\mu}{\sim} y} |(\mathbf{1}_x - \mathbf{1}_y)^* P\zeta| \tag{10.2}$$

Remark 10.3. — The success probability in both statements above can be chosen to be $1 - n^{-C}$ for any constant C (due to concentration of measure).

11. Isolating a cut of good matchings

We now additionally require $\zeta \perp \mathbf{1}$. Let $u = P\zeta$ and renumber its coordinates so that $u_1 \leqslant u_2 \leqslant \cdots \leqslant u_n$.

LEMMA 11.1. — *For any matching μ between $\{1, \ldots, n/2\}$ and $\{n/2+1, \ldots, n\}$,*

$$\mathbf{E} \sum_{x \overset{\mu}{\sim} y} (u_x - u_y)^2 \geqslant \frac{\psi(P)}{n-1} \tag{11.1}$$

COROLLARY 11.2. —

$$\mathbf{E}\Big(\psi(P) - \psi(W_\mu P)\Big) \geqslant \frac{\psi(P)}{O(\log n)} \tag{11.2}$$

Proof of Lemma 11.1. — Let η be the median of $u_1, \ldots, u_n$.

$$\begin{aligned}\sum_{x \overset{\mu}{\sim} y} (u_x - u_y)^2 &\geqslant \sum_{x \overset{\mu}{\sim} y} \Big((u_x - \eta)^2 + (u_y - \eta)^2\Big) \\ &= \sum_x (u_x - \eta)^2 \\ &= \sum_x u_x^2 - 2\eta \sum_x u_x + n\eta \\ &\geqslant \sum_x u_x^2 \qquad (\dagger)\end{aligned}$$

Step (†) follows from $\zeta \perp \mathbf{1}$ and the column-stochasticity of P as $\sum_x u_x = \mathbf{1}^* P\zeta = \mathbf{1}^*\zeta = 0$.

Note that $u_x = \mathbf{1}_x^* P\zeta = \big(\mathbf{1}_x P - \mathbf{1}^*/n\big)\zeta$, since $\zeta \perp \mathbf{1}$. Observe that $w_x := (\mathbf{1}_x^* P - \mathbf{1}/n) \perp \mathbf{1}$, using row-stochasticity. And since $\zeta \perp \mathbf{1}$ and $w_x \perp \mathbf{1}$, applying the concentration of measure lemma gives

$$\mathbf{E}\, u_x^2 = \frac{\|\mathbf{1}_x^* P - \mathbf{1}/n\|^2}{n-1}$$

Thus,

$$\begin{aligned}\mathbf{E} \sum_{x \overset{\mu}{\sim} y} (u_x - u_y)^2 &\geqslant \mathbf{E} \sum_x u_x^2 \\ &= \sum_x \frac{\|\mathbf{1}_x^* P - \mathbf{1}/n\|^2}{n-1} \\ &= \frac{\psi(P)}{n-1}\end{aligned}$$

∎

12. Finding a matching or a cut

For simplicity, assume G is unweighted.

ALGORITHM 12.1 (*Cut-or-Flow*). — *The input is $S \subset V$ with $|S| = n/2$ and maximum allowable congestion $1/\alpha > 0$:*

1. *Assign each edge in G capacity $1/\alpha$. Add a* source *node with an outgoing unit-capacity arc to each vertex in S, and add a* sink *node with an incoming unit-capacity arc from each vertex in S^c.*

2. *Find a maximum flow between the source and the sink*
3. *If the flow value is at least $n/2$, we produce a matching between S and S^c, by decomposing the flow into flow paths*
4. *Otherwise, we find a minimum cut separating the source and the sink, and output the partition induced on V*

Remark 12.1. — If a matching is found, then by construction it can be embedded in G with congestion at most $1/\alpha$.

Lemma 12.2. — *If the maximum flow-value between the source and the sink is less than $n/2$, then the minimum cut in G has expansion less than α.*

Proof. —

— If the maximum flow is $< n/2$ then the minimum cut (separating the source and the sink) is also $< n/2$.
— Let the number (and capacity) of edges in the cut incident to the source (and respectively sink) be n_s (and n_t).
— The remaining cut capacity is $n/2 - n_s - n_t$, thus using at most $\alpha(n/2 - n_s - n_t)$ edges of G
— The cut separates $\geqslant n/2 - n_s$ vertices in S from $\geqslant n/2 - n_t$ vertices in S^c, thus

$$\Phi_G(S, S^c) \leqslant \alpha \frac{n/2 - n_s - n_t}{\min\{n/2 - n_s, n/2 - n_t\}} \leqslant \alpha$$

∎

13. Running time and speedup via sparsification

Break down:

— $O(\log^2 n)$ iterations w.h.p., each including:
— *Find-Bisection* in $\tilde{O}(n)$,
— Single commodity flow in $O(m^{3/2})$ using [3],
— Decomposition into paths in $\tilde{O}(m)$

In total $\tilde{O}(n + m^{3/2})$. Can get $\tilde{O}(m + n^{3/2})$ using

Theorem 13.1 (Benczùr–Karger [2]). — *Given a graph G with n vertices and m edges and an error parameter $\epsilon > 0$, there is a graph $\hat{G}$ such that*

(i) *$\hat{G}$ has $O(n \log n/\epsilon^2)$ edges and*
(ii) *The value of every cut in $\hat{G}$ is within $(1 \pm \epsilon)$ factor of the corresponding cut in G.*

$\hat{G}$ can be constructed in $O(m \log^2 n)$ time if G is unweighted and in $O(m \log^3 n)$ time if G is weighted.

14. Authors' conclusions and open problems

— The union of the flows corresponds to an embedding of a complete graph
— Can this approach yield a (tight) $O(\log n)$ approximation for embedding a complete graph?
— What about improving to a $O(\sqrt{\log n})$ approximation algorithm by embedding an arbitrary expander?
— Can this analysis be related to random collapsing process underlying METIS?

15. Petar's remarks

(1) Lower bound gap — When an embedding is found, the KRV algorithm/analysis is insensitive to whether some subset of the matchings can be embedded simultaneously (with congestion $\leqslant 1/\alpha$), which would exhibit tighter lower-bound certificates
(2) Upper bound gap — consider the sparsest cut in a graph comprised of two copies of K_n connected by an edge. Explain how the KRV algorithm can succeed in finding an embedding when $\alpha = \log^2 n/n$
(3) It is conceivable that for a choice of α in the critical region $\Phi(G) \leqslant \alpha \leqslant O(\log^2 n \cdot \Phi(G))$, multiple runs of the algorithm will eventually find a sparser cut. But this needs to be proven and will most likely happen with tiny probability (unless the graph has many sparsest cuts)
(4) Is it possible to build an expander using $o(\log n)$ matchings? How (Butterfly, de Bruijn)? If yes, then replicating the same argument will give an approximation guarantee (using multi-commodity flow to embed desired matchings, and multi-commodity duality to get a cut otherwise). More on this in [4]
(5) The analysis in this paper is very much about expanders built from matchings, and it is not tight because $\psi \leqslant 1/4n^2$ implies $P_{xy} \geqslant 1/2n$, but the converse is not true. In other words, the needed condition is $\|\mathbf{J}-P\|_\infty \leqslant 1-1/2n$, but the enforced condition is $\|P-\mathbf{J}/n\|_2 \leqslant 1/2n$

BIBLIOGRAPHY

[1] Khandekar, Rao, Vazirani, *Graph partitioning using single commodity flows*, STOC'06
[2] Benczùr, Karger, *Approximating s-t minimum cuts in $\tilde{O}(n^2)$ time*, STOC'96

[3] Goldberg, Rao, *Beyond the flow decomposition barrier*, JACM'98
[4] Arora, Hazan, Kale, $O(\sqrt{\log n})$ *approximation to sparsest cut in* $\tilde{O}(n^2)$ *time*, FOCS'04