On Partitioning Graphs via Single Commodity Flows

Lorenzo Orecchia [*]
University of California
Berkeley, USA
orecchia@cs.berkeley.edu

Leonard J. Schulman[T]
California Institute of Technology
Pasadena, USA
schulman@caltech.edu

Umesh V. Vazirani
University of California
Berkeley, USA
vazirani@cs.berkeley.edu

Nisheeth K. Vishnoi [‡]
IBM India Research Lab
New Delhi, India
nisheeth.vishnoi@gmail.com

ABSTRACT

In this paper we present two combinatorial algorithms for Sparsest Cut which achieve an $O(\log n)$ approximation factor and run in time $\widetilde{O}(m + n^{1.5})$. This is achieved in the framework proposed in [Khandekar et al., 2006] to partition graphs *quickly* using only poly-logarithmically many single-commodity max-flow computations while, at the same time, keeping the approximation factor $O(\log^2 n)$. Complementing our algorithmic results, we prove that, in this framework of reducing the Sparsest Cut problem to single-commodity flows, one cannot get an approximation factor better than $\Omega(\sqrt{\log n})$.

Categories and Subject Descriptors

F.22 [**Analysis of Algorithms and Problem Complexity**]: Non-numerical Algorithms and Problems

General Terms

Algorithms, Theory

Keywords

Edge-separator, graph partitioning, single-commmodity max-flow, sparsest cut, spectral method, matrix exponential

1. INTRODUCTION

The problem. The Sparsest Cut problem is the following: Given a graph $G = (V, E)$, find a partition $(S, \overline{S})$ of the vertices which has the minimum edge expansion. The edge expansion of a cut $(S, \overline{S})$ is defined to be $\frac{|E(S,\overline{S})|}{\min\{|S|,|\overline{S}|\}}$. Interest in this problem derives both from its numerous practical applications such as image segmentation, VLSI layout and clustering (see the survey [Shmoys, 1996]), and from its theoretical connections to spectral methods, linear/semidefinite programming, measure concentration and metric embeddings. In this paper, we provide fast algorithms for this problem using the framework proposed in [Khandekar et al., 2006]. This framework reduces the NP-hard Sparsest Cut problem to the computation of a poly-logarithmic number of single-commodity max-flows while, at the same time, keeping the approximation factor poly-logarithmic. We also prove lower bounds on the best approximation factor that can be obtained in this framework.

The cut-matching game. At the heart of the results of [Khandekar et al., 2006] lies the following two person game which starts with an empty graph on n vertices: in each round, the cut player chooses a bisection $(S, \overline{S})$ of the vertices and in response the matching player chooses a perfect matching that pairs each vertex in S with a unique vertex in $\overline{S}$. The game ends when the (multi)-graph consisting of the multi-set union of the perfect matchings has edge edge expansion at least 1, i.e., every subset T of at most $n/2$ vertices has at least $|T|$ edges leaving it. The goal of the cut player is to minimize the number of rounds of play, while the matching player tries to draw the game out for as many rounds as possible. [Khandekar et al., 2006] show that any strategy for the cut player that guarantees termination in t rounds yields a $O(t)$-approximation algorithm for Sparsest Cut. Moreover the resulting algorithm runs in $O(t(T_c+T_f))$ where T_c is the time to implement the cut player and T_f is the running time of a single commodity max-flow computation. They also give a quasi-linear time implementation of a cut player strategy that achieves $t = O(\log^2 n)$, thereby obtaining an $O(\log^2 n)$ approximation algorithm for Sparsest Cut whose running time is dominated by $O(\log^2 n)$ single commodity max-flow computations. Subsequently, [Arora and Kale, 2007] gave an algorithm that achieved an approximation ratio of $O(\log n)$ while still running in time dominated by poly-logarithmic single commodity max-flow computations. Their algorithm did not rely on the cut-matching game, but worked in a more general framework for designing primal-dual algorithms for SDPs.

[T]This work is supported by NSA Award H98230-06-1-0074, NSF Award CCF-0515342.
[‡]Part of this work was done when author was visiting UC Berkeley and is supported by ??
[*]All authors were supported by NSF Grant 0635401.

STOC'08, May 17–20, 2008, Victoria, British Columbia, Canada.

This work. In this paper we investigate what is the best approximation factor for Sparsest Cut achievable in this framework of [Khandekar et al., 2006]. We start by observing that rather than the number of rounds, a better measure of performance in the cut-matching game is the ratio of the number of rounds to the expansion of the union of matchings produced. This quantity, which we refer to as expansion ratio, upper bounds the approximation ratio achieved by the algorithm. Thus, rather than reducing the number of rounds, we can increase the expansion of the union of matchings. Indeed, we cannot hope to reduce the number of rounds, because it can be shown that the bound of $\Omega(1)$ on the expansion in [Khandekar et al., 2006] is tight, and therefore, the approximation ratio achieved by their algorithm is equal to the number of rounds and, therefore, $\Omega(\log^2 n)$. We give two simple modifications of the cut-finding procedure of [Khandekar et al., 2006] that result in an expansion of $\Omega(\log n)$, while maintaining number of rounds equal to $O(\log^2 n)$. These result in simple and easy to analyze combinatorial algorithms for Sparsest Cut that match the Leighton-Rao approximation ratio of $O(\log n)$, while using single-commodity max-flows rather than multi-commodity flows. In theoretical terms, single-commodity max-flows can be computed in time roughly $O(m+n^{1.5})$ by a result of [Goldberg and Rao, 1998]. The central motivation in [Khandekar et al., 2006] was to design fast algorithms with provable approximation guarantees for Sparsest Cut that could compete with practical heuristics such as METIS in running time. They also noted certain similarities of the resulting algorithm with METIS. The algorithms in this paper add to this mix, both in terms of the potential for practical improvements over METIS, as well as a rigorous justification of METIS-like heuristics. DELETED NEXT:

The cut-finding procedures we propose are quite similar to those of [Khandekar et al., 2006]. They perform a random walk on the graph of matchings, starting from a suitably random initial distribution, to find the bisection to present next to the cut player. The difference lies in the kind of random walk. In [Khandekar et al., 2006] the walk mixed along each matching in turn (which we refer to as a round robin walk), whereas our walks more closely resemble the natural random walk on the graph of matchings and allow for more effective bounds on the spectral gap. The analysis of the resulting cut-finding procedures is similar to that of [Khandekar et al., 2006] in that it makes use of a potential function related to the mixing of the walk on the current union of matchings: However, bounding the change in potential function is now much trickier. The advantage is that we can exploit the direct connection of the mixing of the walk to the spectral gap of the graph.

Turning to lower bounds, we prove an $\Omega(\sqrt{\log n})$ bound on the rounds to expansion ratio for the cut-matching game. It follows that this framework cannot be used to design an algorithm for Sparsest Cut with approximation ratio better than $\Omega(\sqrt{\log n})$. By contrast, the best lower bound known for the approach in [Arora et al., 2004b] is $\Omega(\log \log n)$ as proved in [Devanur et al., 2006] via a involved and technical argument. This suggests that the cut-matching game provides an attractive, concrete framework in which to study the complexity of finding sparse cuts. An intruiging question here is whether this lower bound is tight and whether there is an efficient cut player strategy that achieves a $O(\sqrt{\log n})$ bound. It is encouraging to note that the algorithm of [Arora et al., 2004b], which achieves a $O(\sqrt{\log n})$ approximation, also outputs, as a certificate of expansion, a union of large matchings.

The proof of the lower bound relies on the following combinatorial statement that is related to the isoperimetric inequality for the hypercube: Given a bisection of the vertices of the d-dimensional hypercube, there is always a pairing of vertices between the two sides such that the average Hamming distance between paired vertices is at most $O(\sqrt{d})$. As far as we know, this natural property has not been studied, and is worth exploring further.

Finally, we mention that a combinatorial study of the cut-matching game was done in [Khandekar et al., 2007], where it was shown that there is a cut-finding procedure, albeit, not known to be efficient, which would terminate in $O(\log n)$ rounds. In the same report, the connection between the approach of [Arora et al., 2004b] and [Khandekar et al., 2006] was made explicit and it was shown that any integrality gap, $\mathsf{gap}(n)$ example for the approach of [Arora et al., 2004b] can be used to prove a $\mathsf{gap}(n)$ lower bound on the number of rounds required by any cut-finding procedure.

Organization. The two cut-finding procedures appear in Section 3 and the lower bound appears in Section 4. Proofs of the more technical lemmata are deferred to the Appendix. In the remaining of this paper, we denote by $\phi(G)$ the edge expansion of a graph G, i.e., the minimum edge expansion over all cuts of G. We start by reviewing the framework of [Khandekar et al., 2006].

2. REVIEW OF KRV FRAMEWORK

Certifying expansion. One way to certify that a graph G has no sparse cut is to appeal to the expander flow formalism of [Arora et al., 2004b]. This consists of constructing a graph H of known expansion on n vertices and embedding it as a flow in G such that the flow routed through each edge in G is at most unit. Then the expansion of H is a lower bound on the expansion of G. Indeed, the seminal result of [Leighton and Rao, 1999] may be viewed as a multi-commodity flow based algorithm to embed the scaled complete graph $\frac{\alpha}{n \log n} K_n$ in any n-vertex graph G or produce a cut in G of expansion at most α. The expander flow formulation of the $O(\sqrt{\log n})$ factor approximation in [Arora et al., 2004b] established that for every graph G, either one can find, using an SDP, a cut of expansion at most α, or there is an $\Omega(\alpha/\sqrt{\log n})$ expander $H(G)$ that one can embed as a flow in it. An efficient implementation of this algorithm was given by [Arora et al., 2004a], showing how to combinatorially solve this SDP in time comparable to that of computing multi-commodity flows.

Converting a cut player strategy to a Sparsest Cut **algorithm.** [Khandekar et al., 2006] show that a cut player strategy producing an expander in $t(n)$ rounds can be converted into a $O(t(n))$ approximation algorithm using single-commodity max-flows. Here is how: Given a bisection $(S_t, \overline{S_t})$, the matching player performs a single-commodity max-flow computation in an attempt to route a perfect matching between S_t and $\overline{S_t}$ with minimum congestion c_t and outputs the matching and the min-cut obtained. It is not hard to show that the expansion of the min-cut is at most $\frac{1}{c_t}$. After $t(n)$ rounds, the algorithm outputs the worst such cut, which has expansion $\frac{1}{c_{\max}}$. At this point, the union of the matchings returned by the matching player has constant expansion and can be embedded in G with congestion equal to

$\sum c_t \leq c_{\max} t(n)$. This implies that G must have expansion at least $\Omega\left(\frac{1}{c_{\max} t(n)}\right)$. Hence, the cut that was output is within $O(t(n))$ of the optimal. Moreover, the running time of the algorithm is $O(t(n)(T_c + T_f))$, where T_c is the time required by the cut player to compute his response in each round, and T_f is the time for a single commodity max-flow computation.

The cut-finding procedure $\mathcal{C}_{\mathrm{KRV}}$. Let us recall the cut player strategy $\mathcal{C}_{\mathrm{KRV}}$ of [Khandekar et al., 2006]. Given a set of perfect matchings $(M_1, \ldots, M_t)$ on n vertices, such that their union is not an expander yet, the goal of $\mathcal{C}_{\mathrm{KRV}}$ is to find a non-expanding cut. The procedure starts with a random bisection on n vertices, which can be thought of as assigning to a randomly chosen set of half of the vertices a "charge" of 1 and to the other half a charge of -1. It then iterates through each matching in turn to "mix" the charge across the matchings. At time j, $\mathcal{C}_{\mathrm{KRV}}$ averages the charge across the edges of M_j. The total charge is always zero. More precisely, after $j-1$ steps, let $\mathbf{u}_{j-1}$ be the vector whose i-th coordinate indicates the charge on vertex i. Then $\mathbf{u}_j := \left(\frac{I}{2} + \frac{M_j}{2}\right)\mathbf{u}_{j-1}$. After all the t matchings have been processed, $\mathcal{C}_{\mathrm{KRV}}$ sorts the vertices by the charge remaining on them, indicated by the coordinates of vector $\mathbf{u}_t$, and then outputs a bisection whose two parts are the first half and the second half of vertices in this order respectively.

Intuitively, the reason this procedure finds a bisection containing a non-expanding cut is the following: Suppose $(S, \overline{S})$ is a cut across which there are very few edges, say none. Then the averaging procedure using matching edges never transfers any charge across this cut. Also, the initial random bisection will create a $\Theta(\sqrt{n})$ charge differential between S and $\overline{S}$, i.e, for all j, $|\sum_{i \in S} \mathbf{u}_j(i) - \sum_{i \in \overline{S}} \mathbf{u}_j(i)| = \Theta(\sqrt{n})$. Hence, in the bisection output after mixing and then sorting based on $\mathbf{u}_t$, the two sides of the bisection will have non-trivial correlations with S and $\overline{S}$ respectively. Thus, the matching added in the next iteration will add some edges across the sparse cut $(S, \overline{S})$. It is remarkable that strategy $\mathcal{C}_{\mathrm{KRV}}$, after $O(\log^2 n)$ of such simplistic steps, is able to ensure that there is no non-expanding cut with constant probability.

3. UPPER BOUNDS

3.1 Preliminaries

Graphs, Random Walks and Eigenvalues. For a symmetric matrix $M \in \mathbb{R}^{n \times n}$, let $L(M) \in \mathbb{R}^{n \times n}$ be its combinatorial Laplacian which is defined as $L(M)_{ii} := \sum_{j \neq i} M_{ij}$, and $L(M)_{ij} := -M_{ij}$ if $i \neq j$. The eigenvalues of a symmetric matrix $M \in \mathbb{R}^{n \times n}$, are denoted by $\lambda_1(M) \leq \cdots \leq \lambda_n(M)$. In the following, we identify a graph G with its adjacency matrix and let $L(G)$ denote the combinatorial Laplacian of the graph G. Note that $L(G)$ is symmetric and positive semi-definite. For all graphs G, we then have $\lambda_1(L(G)) = 0$, with corresponding eigenvector $\vec{1}$. The following is a fundamental relation between $\lambda_2(L(G))$ and the expansion of a d-regular graph [Chung, 1997].

Theorem 3.1 (**Cheeger's Bound**). *For an undirected d-regular graph G:*

$$\phi(G) \geq \frac{1}{2}\lambda_2(L(G))$$

All graphs considered in this paper are undirected, they may have parallel edges but have no loops, and are assumed to have an even number of vertices. Some of the graphs will have weighted edges. We extend the definition of expansion to these edge-weighted graphs by replacing the cardinality of the cut with the sum of the weights of the edges in the cut. It is also possible to show that Cheeger's Bound holds in the same form for such weighted graphs, if we let the degree of a vertex be the sum of the weights of the edges incident to it.

In the remainder of this section, we describe and analyze our two new cut player strategies. Before doing so, let us understand why the procedure $\mathcal{C}_{\mathrm{KRV}}$ described in the previous section cannot guarantee edge expansion more than $O(1)$ at termination. Suppose the union of matchings $(M_1, \ldots, M_t)$, have no edges crossing some bisection $(S, \overline{S})$. Now suppose that the round-robin walk on matchings $M_1, \ldots, M_t$ mixes the charges on each side of $(S, \overline{S})$ perfectly (or very well). The next cut selected by $\mathcal{C}_{\mathrm{KRV}}$ is necessarily $(S, \overline{S})$. Moreover, presented with any perfect matching crossing $\mathcal{C}_{\mathrm{KRV}}$, when $\mathcal{C}_{\mathrm{KRV}}$ mixes across that perfect matching, the charge at each vertex becomes zero, and $\mathcal{C}_{\mathrm{KRV}}$ will terminate. So at termination, the edge expansion across bisection $(S, \overline{S})$ is 1.

3.2 The Cut-Finding Procedure $\mathcal{C}_{\mathrm{nat}}$

After j rounds of the cut-matching game, we have seen matchings $M_1, \ldots, M_j$ and the cut player must pick a bisection of the graph $G_j := \sum_{i=1}^{j} M_i$.

Ideally we would like to analyze the following cut player strategy:

- Pick the initial vector of charges $\mathbf{u}$ as in the previously described cut finding strategy $\mathcal{C}_{\mathrm{KRV}}$.
- Repeat for t iterations: At the j-th iteration, given the graph G_j, apply t steps of the lazy natural random walk on G_j to $\mathbf{u}$, i.e., compute $\mathbf{y} := \left(\frac{jI + G_j}{2j}\right)^t \mathbf{u}$.
- Finishing up: As in strategy $\mathcal{C}_{\mathrm{KRV}}$, after t iterations, sort the coordinates of $\mathbf{y}$ and output the median cut.

Can we analyze this new strategy along the lines of the analysis of strategy $\mathcal{C}_{\mathrm{KRV}}$ in [Khandekar et al., 2006]? The broad outline of the analysis there was to introduce a potential function measuring how "well mixed" the walk is after j iterations, and show that the new matching M_{j+1} significantly reduces this potential function. The round robin nature of the random walk greatly facilitates this analysis, since the process after $j+1$ iterations is exactly the process after j iterations followed by averaging across matching M_{j+1}. Therefore computing the potential reduction in the $j+1$-th iteration simply boils down to analyzing how much the potential drops due to averaging across matching M_{j+1}.

Things are much more complicated for the lazy natural walk in the procedure sketched above. This is because the process after $j+1$ iterations no longer has such a simple decomposition: all steps of the random walk include the possibility of using edges from the matching M_{j+1}. The actual analysis of the random walk is based on the matrix inequality $\|(ABA)^k\|_F \leq \|A^k B^k A^k\|_F$, which facilitates the desired decomposition. To make all this work out we have to modify the actual random walk as follows: let $N_t := \frac{d}{d+1}I + \frac{1}{d+1}M_t$. Then a single step of the random walk after

t iterations is given by $N_t N_{t-1} \cdots N_2 N_1 N_2 \cdots N_{t-1} N_t$. The t-th iteration consists of applying this random walk $d+1$ times. Intuitively each step of the random walk corresponds to a step of the natural random walk. It might be possible to show that the analysis of the cut player strategy with this modified random walk directly implies an analysis with the natural random walk. We leave this as an open question.

The analysis. Let $d+1$ be such that it is equal to 2^k for some integer $k > 0$ to be fixed later. $d+1$ will be the duration of the game. Fix a matching player $\mathcal{M}$. Let M_t be the perfect matching on $[n]$ output by $\mathcal{M}$ at round t when presented by the cut $(S_t, \overline{S}_t)$ by $\mathcal{C}_{\text{nat}}$. The following probability transition matrices (in $\mathbb{R}^{n\times n}$), defined recursively, are used by the cut-finding procedure: $R_0 := I$. For $1 \le t \le d+1$,

$$R_t := N_t R_{t-1} N_t,$$

where $N_t := \frac{d}{d+1} I + \frac{1}{d+1} M_t$. Note that, for all t , R_t is a symmetric doubly stochastic matrix with largest eigenvalue 1 (corresponding to the all 1s eigenvector $\mathbf{1}$). Define:

$$W_t := R_t^{d+1}$$

The cut-finding procedure, $\mathcal{C}_{\text{nat}}$, for step $t+1$ is as follows. Let $(M_1, \ldots, M_t)$ be fixed for some $t \ge 0$.

1. Pick a random unit vector $\mathbf{u}_{t+1} \in \mathbb{S}^{n-1} \perp \mathbf{1}$ independent of all previous choices.
2. Compute $\mathbf{y}_{t+1} := W_t \mathbf{u}_{t+1}$.
3. Sort the entries of $\mathbf{y}_{t+1} = (y_1, \ldots, y_n)$ as $y_{i_1} \le \cdots \le y_{i_{n/2}} \le y_{i_{n/2+1}} \le \cdots \le y_{i_n}$.
4. Let $S_{t+1} := \{i_1, \ldots, i_{n/2}\}$ and $\overline{S}_{t+1} := [n] \backslash S_{t+1}$, and output $(S_{t+1}, \overline{S}_{t+1})$.

Analysis

THEOREM 3.2. *$\mathcal{C}_{\text{nat}}$ achieves an expansion of $\Omega(\log n)$ in $O(\log^2 n)$ rounds against any matching player $\mathcal{M}$ with constant probability.*

As in [Khandekar et al., 2006], we measure the progress of $\mathcal{C}_{\text{nat}}$ by a simple potential function which is the sum of squares of all the eigenvalues, except the largest one, of the walk matrix used to mix the charge. Formally, if for a symmetric matrix X, we let $\mathrm{Tr}_1[X]$ denote the sum of all the eigenvalues of X except the largest one, then the potential function for the analysis is $\mathrm{Tr}_1[W_t^2] = \sum_{i=1}^{n-1} \lambda_i^2(W_t) = \|W_t - \frac{1}{n}\vec{1}\vec{1}^T\|_F^2$. Now we analyze the reduction in the potential by comparing the potentials at time t and $t+1$ respectively. Let M_{t+1} be the perfect matching across $(S_{t+1}, \overline{S}_{t+1})$ obtained at the $(t+1)$-th step. The following lemma proves that once we fix $(M_1, \ldots, M_t)$, for every perfect matching M_{t+1} across the bisection $(S_{t+1}, \overline{S}_{t+1})$, the potential reduces by a multiplicative factor of $1 - \frac{c}{\log n}$, for some fixed constant $c > 0$, in expectation over the choice of $\mathbf{u}_{t+1}$.

LEMMA 3.3 (POTENTIAL REDUCTION). *Let $(M_1, \ldots, M_t)$ be fixed. Then there is a constant $c > 0$ such that, for any perfect matching M_{t+1} across the bisection $(S_{t+1}, \overline{S}_{t+1})$ (which is selected using $\mathbf{u}_{t+1}$),*

$$\mathbf{E}_{\mathbf{u}_{t+1}}\left[\mathrm{Tr}_1[W_{t+1}^2]\right] \le \left(1 - \frac{c}{\log n}\right) \mathrm{Tr}_1[W_t^2].$$

PROOF. Recall $W_t = R_t^{d+1}$ and $R_{t+1} := N_{t+1} R_t N_{t+1}$. Hence, by the matrix inequality in Theorem A.2 in the Appendix, it follows that

$$\mathrm{Tr}_1[W_{t+1}^2] = \mathrm{Tr}_1\left[(N_{t+1} R_t N_{t+1})^{2d+2}\right]$$
$$\le \mathrm{Tr}_1\left[N_{t+1}^{2d+2} R_t^{2d+2} N_{t+1}^{2d+2}\right] = \mathrm{Tr}_1\left[N_{t+1}^{4d+4} W_t^2\right].$$

Here the last equality follows from Fact A.1 in the Appendix and from the definition of W_t.

The following fact allows us to estimate the contribution of N_{t+1} in the expression above. Its proof is found in the Appendix.

FACT 3.4. *Let $\lambda := \frac{1}{2} - \frac{1}{2}\left(1 - \frac{2}{d+1}\right)^{4d+4}$. Then, $N_{t+1}^{4d+4} = I - \lambda(I - M_{t+1})$.*

Using this, one can write:

$$\mathrm{Tr}\left[N_{t+1}^{4d+4} W_t^2\right] = \mathrm{Tr}\left[(I - \lambda(I - M_{t+1}))\, W_t^2\right]$$
$$= \mathrm{Tr}\left[W_t^2\right] - \lambda \mathrm{Tr}\left[L(M_{t+1}) W_t^2\right].$$

The decrease in potential $\mathrm{Tr}_1\left[W_t^2\right] - \mathrm{Tr}_1\left[W_{t+1}^2\right]$ is

$$\lambda \mathrm{Tr}\left[L(M_{t+1}) W_t^2\right] = \lambda \sum_{\{i,j\} \in M_{t+1}} \|\mathbf{w}_i - \mathbf{w}_j\|^2,$$

where $\mathbf{w}_i$ is the i-th row vector of W_t. Now we need to show that the above potential reduction is large in expectation.

The following lemma is based on the Gaussian nature of projections and appears in the same form in [Khandekar et al., 2006]. The proof follows from the Projection Lemma (Lemma B.2 in the Appendix) applied to $\{v_i := \mathbf{w}_i - \frac{1}{n}\mathbf{1}\}$.

LEMMA 3.5. *There is a constant $c' > 0$ such that*

$$\mathbf{E}_{\mathbf{u}_{t+1}}\left[\sum_{\{i,j\} \in M_{t+1}} \|\mathbf{w}_i - \mathbf{w}_j\|^2\right] \ge \frac{c'}{\log n} \cdot \mathrm{Tr}_1\left[W_t^2\right]$$

Notice that $\lambda \ge \frac{1 - \mathrm{e}^{-8}}{2}$. Hence, it follows from the lemma that

$$E_{\mathbf{u}_{t+1}}\left[\mathrm{Tr}_1[W_{t+1}^2]\right] \le \left(1 - \frac{(1 - \mathrm{e}^{-8})c'}{2\log n}\right) \mathrm{Tr}_1[W_t^2].$$

This completes the proof of Lemma 3.3. □

This implies that starting with a potential of $n-1$, the potential drops at a rate of roughly $(1 - c/\log n)$ in expectation. Hence, combining the recursive application of the previous lemma with Markov's Inequality, we obtain the following corollary.

COROLLARY 3.6 (TOTAL POTENTIAL). *With constant probability over the choices of $(\mathbf{u}_1, \ldots, \mathbf{u}_{d+1})$,*

$$\mathrm{Tr}_1[W_{d+1}^2] \le \left(1 - \frac{c}{\log n}\right)^{d+1} n.$$

Hence, for $d = O(\log^2 n)$, the potential drops below $\frac{1}{n}$, and we have that $\lambda_{n-1}(W_{d+1}^2) \le \frac{1}{n}$.

Now, differently from [Khandekar et al., 2006], we embed R_{d+1} (and not the walk W_{d+1} itself) in G_{d+1} to show that $\phi(G_{d+1}) = \Omega(\log n)$. In particular, we use the bound on $\lambda_{n-1}(W_{d+1}^2)$ to show a lower bound on the spectral gap of the 1-regular weighted graph R_{d+1}. A simple application of Cheeger's Bound then yields the final result.

LEMMA 3.7 (ESTIMATING EXPANSION).
Consider the graph G_{d+1} on the vertex set $[n]$ formed by the union of matchings $(M_1, \ldots, M_{d+1})$. Then, for $d = O(\log^2 n)$, $\phi(G_{d+1}) \geq \Omega(\log n)$ with constant probability.

PROOF. By the argument above: $\lambda_{n-1}(W_{d+1}^2) = (1 - \lambda_2(L(R_{d+1})))^{2d+2} \leq \frac{1}{n}$. Hence, $\lambda_2(L(R_{d+1})) \geq \Omega\left(\frac{1}{\log n}\right)$. By Cheeger's Bound, this implies that $\phi(R_{d+1}) = \Omega\left(\frac{1}{\log n}\right)$. Finally, as R_{d+1} embeds in G_{d+1} with congestion $\frac{2}{d+1}$, we have $\phi(G_{d+1}) \geq \frac{d+1}{2}\ \phi(R_{d+1}) = \Omega(\log n)$. $\square$

Hence, with constant probability, after $d = O(\log^2 n)$ rounds one can ensure that the graph obtained by the union of matchings $(M_1, \ldots, M_{d+1})$ has expansion at least $\Omega(\log n)$. This completes the proof of Theorem 3.2.

Running time

Note that we do not need to compute W_t explicitly, as we only use W_t to compute $W_t\mathbf{u}_t$. Hence, at iteration t we only need to perform $O(2t\log^2 n)$ matrix-vector multiplications. As each matrix is a step of a lazy random walk along a matching, each of these operations takes time $O(n)$. As t varies from 0 to $O(\log^2 n)$, the total running time throughout the game of the cut-finding procedure $\mathcal{C}_{\text{nat}}$ is $O(n \log^6 n) = \widetilde{O}(n)$.

3.3 The Cut-Finding Procedure $\mathcal{C}_{\text{exp}}$

Preliminaries: Matrix Exponential

DEFINITION 3.8 (MATRIX EXPONENTIAL).
For a matrix $X \in \mathbb{R}^{n\times n}$, the matrix exponential e^X of X is defined as the following Taylor Series:

$$e^X := \sum_{j=0}^{\infty} \frac{X^j}{j!}$$

If X is symmetric, then X has an ortho-normal basis of eigenvectors and, hence, X and e^X are simultaneously diagonalizable. Thus, for all i: $\lambda_i(e^X) = e^{\lambda_i(X)}$.

A basic identity $e^X e^Y = e^{X+Y}$, that is true for scalar exponentiation, holds for matrix exponentiation only under the condition that X and Y commute. The following theorem shows that something weaker is true in general, and it turns out to be sufficient for our purpose.

THEOREM 3.9 (GOLDEN-THOMPSON INEQUALITY). *Let $X, Y \in \mathbb{R}^{n\times n}$ be symmetric matrices. Then, $\mathrm{Tr}[e^{X+Y}] \leq \mathrm{Tr}[e^X e^Y]$.*

It should be noted that this is only true for *two* matrices.

Finding the Cut

Fix a matching player $\mathcal{M}$. Let M_t be the perfect matching on $[n]$ output by $\mathcal{M}$ at round t when presented by the cut $(S_t, \overline{S}_t)$ by $\mathcal{C}_{\text{exp}}$. In the same way $\mathcal{C}_{\text{nat}}$ considered the transition matrix of a walk across the matchings to produce the next bisection to present, $\mathcal{C}_{\text{exp}}$ makes use of the matrix $W_t := e^{-\frac{1}{2}L(G_t)} = e^{-\frac{1}{2}\sum_{i=1}^{t} L(M_i)}$. We note that, for all t , W_t is a doubly stochastic positive semi-definite matrix.

The bisection finding procedure for $\mathcal{C}_{\text{exp}}$ is exactly the same as that described in Subsection 3.2 for $\mathcal{C}_{\text{nat}}$ if we take $W_t = e^{-\frac{1}{2}L(G_t)}$.

Overview. It is worthwhile pointing out that $\mathcal{C}_{\text{KRV}}$ can also be seen as using matrix exponentials. In particular, it is not difficult to show that $\mathrm{e}^{-\eta(I-M)}$ is the matrix representing a kind of slowed down lazy random walk along matching M. As $\left(\frac{I+M}{2}\right)^i = \frac{(I+M)}{2}$ for $i \geq 1$, the Taylor Series of the exponential yields: $\mathrm{e}^{-\eta(I-M)} = \frac{1+\mathrm{e}^{-\eta}}{2} I + \frac{1-\mathrm{e}^{-\eta}}{2} M$. Hence, the $\mathcal{C}_{\text{KRV}}$ mixing procedure is close to the walk

$$\mathrm{e}^{-\eta(I-M_t)}\mathrm{e}^{-\eta(I-M_{t-1})}\cdots\mathrm{e}^{-\eta(I-M_1)}$$

for small η. Our second cut-finding procedure mixes the charge using the matrix $\mathrm{e}^{-\eta(tI-(M_1+\cdots+M_t))}$ on the union of matchings to find a non-expanding cut. If it were true that $\mathrm{e}^{-\eta(tI-(M_1+\cdots+M_t))} \approx e^{-\eta(I-M_t)}\mathrm{e}^{-\eta(I-M_{t-1})}\cdots\mathrm{e}^{-\eta(I-M_1)}$, then it would establish that the two walks are almost the same. In fact, even under the weaker condition that

$$\mathrm{Tr}\left[\mathrm{e}^{-\eta(tI-(M_1+\cdots+M_t))}\right] \leq \mathrm{Tr}\left[\mathrm{e}^{-\eta(I-M_t)}\cdots\mathrm{e}^{-\eta(I-M_1)}\right]$$

we would be done. Unfortunately, we cannot show this to be true. Hence, a different way to analyze this walk is needed. We also note that the random walk in the cut-finding procedure of [Khandekar et al., 2006] is best thought of as a "discrete gradient descent" method while the random walk based on matrix exponential updates as a "continuous gradient descent" method with the same "loss" functions, i.e., those defined by the matchings.

Analysis

THEOREM 3.10. *$\mathcal{C}_{\text{exp}}$ achieves an expansion of $\Omega(\log n)$ in $O(\log^2 n)$ rounds against any matching player $\mathcal{M}$ with constant probability.*

The structure of the proof is the same as for $\mathcal{C}_{\text{nat}}$. Using the exponential walk means that we need to apply the Golden-Thompson Inequality to analyze the potential reduction. Moreover, our certificate of expansion is also different. The potential function is again $\mathrm{Tr}_1[W_t^2]$.

LEMMA 3.11 (POTENTIAL REDUCTION).
Let $(M_1, \ldots, M_t)$ be fixed. Then there is a constant $c > 0$ such that, for any perfect matching M_{t+1} across the bisection $(S_{t+1}, \overline{S}_{t+1})$ (which is selected using $\mathbf{u}_{t+1}$),

$$\mathbf{E}_{\mathbf{u}_{t+1}}\left[\mathrm{Tr}_1[W_{t+1}^2]\right] \leq \left(1 - \frac{c}{\log n}\right)\mathrm{Tr}_1[W_t^2].$$

PROOF. We have:

$$\mathrm{Tr}[W_{t+1}^2] = \mathrm{Tr}\left[e^{-\sum_{i=1}^{t+1} L(M_i)}\right] \leq \mathrm{Tr}\left[W_t^2 e^{-L(M_{t+1})}\right]$$
$$\leq \mathrm{Tr}\left[W_t^2\right] - \left(\frac{1-e^{-2}}{2}\right)\mathrm{Tr}\left[W_t^2 L(M_{t+1})\right].$$

The first inequality follows from the Golden-Thompson Inequality (3.9) and the second inequality follows from Fact A.4 in the Appendix. Hence, the decrease in potential at round $t+1$ is:

$$\mathrm{Tr}_1\left[W_t^2\right] - \mathrm{Tr}_1[W_{t+1}^2] \leq \left(\frac{1-e^{-2}}{2}\right)\mathrm{Tr}\left[W_t^2 L(M_{t+1})\right].$$

Let $\mathbf{w}_i$ denote the i-th row of W_t. Then:

$$\mathrm{Tr}_1\left[W_t^2\right] - \mathrm{Tr}_1[W_{t+1}^2] \leq \left(\frac{1-e^{-2}}{2}\right)\sum_{\{i,j\}\in M_{t+1}} \|\mathbf{w}_i - \mathbf{w}_j\|^2.$$

Now we need to show that the above potential reduction is large. As in the analysis of $\mathcal{C}_{\text{nat}}$, this is a simple application of the Lemma B.2 with $\{\mathbf{v}_i := \mathbf{w}_i - \frac{1}{n}\mathbf{1}\}$. We get:

$$\mathbf{E}_{\mathbf{u}_{t+1}}\left[\mathrm{Tr}_1[W_{t+1}^2]\right] \leq \mathrm{Tr}_1[W_t^2]\left(1 - \left(\frac{c'(1-e^{-2})}{2\log n}\right)\right)$$
$$= \mathrm{Tr}_1[W_t^2]\left(1 - \Omega\left(\frac{1}{\log n}\right)\right)$$

□

By recursively applying the lemma above and by Markov's Inequality, we obtain the following.

COROLLARY 3.12 (TOTAL POTENTIAL). *With constant probability over the choices of* $(\mathbf{u}_1, \ldots, \mathbf{u}_d)$,

$$\mathrm{Tr}_1[W_d^2] \leq \left(1 - \frac{c}{\log n}\right)^d n.$$

LEMMA 3.13 (ESTIMATING EXPANSION). *Consider the graph G_d on the vertex set $[n]$ formed by the union of matchings $(M_1, \ldots, M_d)$. Then, for $d = O(\log^2 n)$,*

$$\phi(G_d) = \Omega(\log n)$$

with constant probability.

PROOF. Let G_d be the union of matchings $(M_1, \ldots, M_d)$ obtained through the cut-finding procedure $\mathcal{C}_{\exp}$. From Corollary 3.12 it follows that for $d = O(\log^2 n)$, $\lambda_{n-1}(W_d) = \lambda_{n-1}(e^{-L(G_d)}) \leq \frac{1}{n}$. By Theorem 3.1, $\phi(G_t) \geq \frac{\lambda_2(L(G_t))}{2}$. Hence:

$$\phi(G_d) \geq \frac{\lambda_2(G_d)}{2} \geq \frac{-\log(\lambda_{n-1}(e^{-L(G_d)}))}{2} = \frac{\log n}{2}.$$

□

This completes the proof of Theorem 3.10.

Running time

We approximate the exponential $e^{-\frac{1}{2}L(G_t)}$ by truncating its Taylor series as it is also done in [Arora and Kale, 2007]. All we need to compute is $e^{-\frac{1}{2}L(G_t)}\mathbf{u}$ for some random unit vector $\mathbf{u}$. We define the approximation $\mathbf{v}_k$ as $\mathbf{v}_k := \sum_{j=0}^{k} \frac{1}{j!}\left(\frac{L(G_t)}{2}\right)^j \mathbf{u}$.

For $\mathbf{v}_k$ to be a good approximation for the purposes of the algorithm we require that

$$\|\mathbf{v}_k - e^{-\frac{1}{2}L(G_t)}\mathbf{u}\|^2 \leq O\left(\frac{1}{\log n}\right).$$

[Arora and Kale, 2007] show that this inequality is satisfied for

$$k \geq \max\left\{\frac{e^2}{2}t, \Omega(\log\log n)\right\}.$$

Finally, to compute $\mathbf{v}_k$ we just need to perform k matrix-vector multiplications. Each of these takes time $O(tn)$ as G_t is t regular. As $t \leq O(\log^2 n)$ and by the bound on k, we get a running time of $O(n\log^6 n) = \widetilde{O}(n)$ for the procedure $\mathcal{C}_{\exp}$.

4. LOWER BOUND

In this section we prove the following lower bound for the cut-matching game.

THEOREM 4.1. *There is a matching player $\mathcal{M}^*$ such that the number of rounds to expansion ratio for any cut player is at least $\Omega(\sqrt{\log n})$.*

This result establishes a limit on the upper bound arguments in the framework of the cut-matching game, i.e., that there do not exist cut players, even computationally unbounded ones, which can achieve a rounds to expansion ratio better than $\Omega(\sqrt{\log n})$ against $\mathcal{M}^*$. In this section we focus on the description and the analysis of $\mathcal{M}^*$ as claimed by the theorem.

4.1 Proof Idea

A matching player $\mathcal{M}$ could keep the game $\mathcal{G}(n)$ going for $\Omega(\sqrt{\log n})$ rounds if, at each of the first $\Omega(\sqrt{\log n})$ rounds, $\mathcal{M}$ is able to exhibit a cut with expansion less than $\frac{1}{4}$ in the graph formed by the union of the matchings thus far. A simple way for $\mathcal{M}$ to do this would be to pick a fixed cut $(D, \overline{D})$ at the beginning of the game and keep this cut as sparse as possible round after round. However, if the cut player guesses one bisection containing or equal to D, any perfect matching that $\mathcal{M}$ adds across this bisection will make the cut $(D, \overline{D})$ have constant expansion immediately.

To overcome this problem, the matching player $\mathcal{M}^*$ first identifies the vertex set $[n]$ with the vertex set of a hypercube with d coordinates, $\{-1,1\}^d$. (Assume $n = 2^d$.) Then, rather than trying to keep one bisection sparse, it tries to keep $d = \log n$ "orthogonal" bisections sparse on an average. The natural choice for such orthogonal bisections for the hypercube vertex set are those induced by the coordinate cuts. Formally, denote this set of bisections by $\mathcal{D} := \{(D_1, \overline{D}_1), \ldots, (D_d, \overline{D}_d)\}$. Here, $D_i := \{(x_1, \ldots, x_d) \in \{-1,1\}^d \mid x_i = 1\}$, and $\overline{D}_i := \{-1,1\}^d \backslash D_i$. The orthogonality makes it possible to add edges across one $(D_i, \overline{D}_i)$ without increasing the expansion of other bisections in $\mathcal{D}$ by too much. The Main Lemma formalizes this intuition by showing that at any iteration t, and for any choice of a bisection $(S_t, \overline{S}_t)$ (by any cut player), there exists a matching M_t across $(S_t, \overline{S}_t)$ which increases the average expansion over $\mathcal{D}$ by at most $O\left(\frac{1}{\sqrt{\log n}}\right)$. This implies that $\Omega(\sqrt{\log n})$ rounds are necessary before the union of matchings has a constant expansion.

Geometrically, the Main Lemma states that given any bisection of the hypercube vertex set, there exists a matching across it with edges of average ℓ_2^2 length $O\left(\frac{1}{\sqrt{d}}\right)$ in the standard embedding of the hypercube on the unit d-dimensional sphere. To establish this lemma, we first encode the task of finding a matching across the given bisection with minimum ℓ_2^2 length as a min-cost perfect-matching LP. Then, we show that the dual of this LP can be interpreted as a non-expanding embedding of the hypercube into ℓ_1. This allows us to use the hypercube vertex iso-perimetry to upper bound its optimal value.

4.2 Preliminaries

Cut vectors

For any cut $(S, \overline{S})$ of $[n]$, we define the cut vector $\vec{x}_S \in \mathbb{R}^n$ by:

$$(\vec{x}_S)_i = \begin{cases} +1 & \text{if } i \in S \\ -1 & \text{if } i \notin S \end{cases}$$

Hence, for any cut $(S, \overline{S})$:

$$\vec{x}_S^{\mathrm{T}} L(G) \vec{x}_S = 4|E(S, \overline{S})|.$$

Vertex iso-perimetry of the hypercube

For any graph $G = (V, E)$, let $\gamma(G)$ denote the vertex iso-perimetry number of G. $\gamma(G)$ is the minimum ratio among all cuts $S \subseteq V$, with $|S| \leq \frac{|V|}{2}$, of the number of neighbors of S outside of S to that of the size of S. That is

$$\gamma(G) := \min_{S \subseteq V,\ |S| \leq \frac{|V|}{2}} \frac{\{i \in V \backslash S : \exists j \in S : \{i,j\} \in E\}}{|S|}.$$

The following is a standard fact about the vertex iso-perimetry of the hypercube [Chung, 1997].

FACT 4.2. $\gamma(H_d) = \Theta\left(\frac{1}{\sqrt{d}}\right)$.

4.3 Proof of Theorem 4.1

Let $n := 2^d$ for a positive integer d. Let H_d denote the d-dimensional hypercube. This is the graph with $V(H_d) := \{-1, 1\}^d$ and $\{i, j\} \in E(H_d)$ if and only if i and j differ in exactly one coordinate. At the start of the game, $\mathcal{M}^*$ picks an arbitrary bijection $f : V \to H_d$. Let U_d be the unit embedding of H_d, i.e., $U_d := \frac{H_d}{\sqrt{d}}$ and, for all $v \in V$, denote by $\mathbf{u}_v$ the point $\frac{f(v)}{\sqrt{d}}$ of U_d. Each dimension cut in H_d corresponds to a cut in V through the mapping f. In particular, we denote by D_i the cut $\{v \in V : f(v)_i = +1\}$, and $\overline{D}_i := V \backslash D_i$. This defines a set $\mathcal{D} := \{D_1, \ldots, D_d\}$ of bisections of V.

Fix an arbitrary cut player $\mathcal{C}$ which at every round presents a bisection to the matching player $\mathcal{M}^*$ to which $\mathcal{M}^*$ must add a perfect matching. Let $G_t := ([n], E_t)$ denote the graph formed by the union of matchings $M_1, \ldots, M_t$ for some integer $t \geq 0$. $G_0 := ([n], \emptyset)$. Define a potential function $\Phi_t := \mathbf{E}_{D_i \leftarrow \mathcal{D}}\left[\frac{E_t(D_i, \overline{D}_i)}{|D_i|}\right]$ to be the expected expansion in G_t of a cut sampled uniformly at random from $\mathcal{D}$. Note that the matching player $\mathcal{M}^*$ is not random. These expectations are just averages and it is convenient to use this language for the proof.

FACT 4.3. $\Phi_t = \frac{1}{2n} \sum_{\{h,k\} \in E_t} \|\mathbf{u}_h - \mathbf{u}_k\|^2$

PROOF.

$$\begin{aligned}
\Phi_t &= \mathbf{E}_{D_i \leftarrow \mathcal{D}}\left[\frac{|E_t(D_i, \overline{D_i})|}{|D_i|}\right] = \mathbf{E}_{D_i \leftarrow \mathcal{D}}\left[\frac{\vec{x}_{D_i}^{\mathrm{T}} L(G_t) \vec{x}_{D_i}}{4|D_i|}\right] \\
&= \frac{1}{d} \sum_{i=1}^{d} \frac{\vec{x}_{D_i}^{\mathrm{T}} L(G_t) \vec{x}_{D_i}}{2n} \\
&= \frac{1}{d} \sum_{i=1}^{d} \frac{\sum_{\{h,k\} \in E_t} \left((\vec{x}_{D_i})_h - (\vec{x}_{D_i})_k\right)^2}{2n} \\
&= \frac{1}{2n} \sum_{\{h,k\} \in E_t} \sum_{i=1}^{d} \left(\frac{1}{\sqrt{d}}(\vec{x}_{D_i})_h - \frac{1}{\sqrt{d}}(\vec{x}_{D_i})_k\right)^2 \\
&= \frac{1}{2n} \sum_{\{h,k\} \in E_t} \|\mathbf{u}_h - \mathbf{u}_k\|^2.
\end{aligned}$$

Notice that in the last inequality we used the definition of the cuts $D_1, \ldots, D_d$ as the coordinate cuts of H_d. □

This shows that Φ_t equals the sum of the squared length of the edges of G_t in the hypercube representation U_d of V, scaled by $2n$. Hence, for any $t \geq 1$, we can rewrite the increase in potential at round t as:

$$\Phi_t - \Phi_{t-1} = \sum_{\{i,j\} \in E_t \backslash E_{t-1}} \|\mathbf{u}_i - \mathbf{u}_j\|^2 = \sum_{\{i,j\} \in M_t} \|\mathbf{u}_i - \mathbf{u}_j\|^2$$

At every iteration t, given $\mathcal{C}$'s choice of $(S_t, \overline{S}_t)$, $\mathcal{M}^*$ adds the matching M_t across $(S_t, \overline{S_t})$ which minimizes $\sum_{\{i,j\} \in M_t} \|\mathbf{u}_i - \mathbf{u}_j\|^2$. This only requires a minimum cost matching computation on the complete bipartite graph induced by $(S_t, \overline{S}_t)$. The proof of Theorem 4.1 is based on the following lemma, which is proved at the end of this section.

LEMMA 4.4 (MAIN LEMMA). *For all bisections $(S, \overline{S})$, there exists a perfect matching M across $(S, \overline{S})$ such that* $\sum_{\{i,j\} \in M} \|\mathbf{u}_i - \mathbf{u}_j\|^2 = O\left(\frac{n}{\sqrt{d}}\right)$.

Here we see how the Main Lemma implies the theorem.

PROOF OF THEOREM 4.1. By the Main Lemma, the potential increase in one round is: $\Phi_t - \Phi_{t-1} = O\left(\frac{1}{\sqrt{d}}\right)$. Hence $\Phi_t = O\left(\frac{t}{\sqrt{d}}\right)$. This implies that $\mathbf{E}_{D_i \leftarrow \mathcal{D}}\left[\frac{|E_t(D_i, \overline{D_i})|}{|D_i|}\right] = O\left(\frac{t}{\sqrt{d}}\right)$. Hence, there exists a cut D_i with $\frac{|E_t(D_i, \overline{D_i})|}{|D_i|} = O\left(\frac{t}{\sqrt{d}}\right)$. This shows that $\phi(G_t) = O\left(\frac{t}{\sqrt{d}}\right)$. As the game stops after t iterations only if $\phi(G_t)$ is constant, we must have $t = \Omega\left(\sqrt{d}\right) = \Omega\left(\sqrt{\log n}\right)$. Moreover, this also shows that $\frac{t}{\Phi(G_t)} = \Omega(\sqrt{d}) = \Omega(\sqrt{\log n})$ for all t. □

We now proceed to prove the Main Lemma.

PROOF OF MAIN LEMMA 4.4. Let $c_{ij} := \|\mathbf{u}_i - \mathbf{u}_j\|^2$. Consider the LP relaxation of Figure 1 for computing the minimum cost perfect matching across the cut $(S, \overline{S})$.

$$\begin{array}{llll}
 & \text{Minimize} & \sum_{i \in S, j \in \overline{S}} c_{ij} x_{ij} & \\
\text{Subject to} & & & \\
 & \forall i \in S, & \sum_{j \in \overline{S}} x_{ij} & = 1 \\
 & \forall j \in \overline{S}, & \sum_{i \in S} x_{ij} & = 1 \\
 & \forall i \in S,\ j \in \overline{S}, & x_{ij} & \geq 0
\end{array}$$

Figure 1: LP for Bipartite Min-Cost Matching

By the integrality of the bipartite perfect matching polytope (see [Papadimitriou and Steiglitz, 1982]), the objective of this program is the minimum of $\sum_{\{i,j\} \in M} \|\mathbf{u}_i - \mathbf{u}_j\|^2$ over all perfect matchings M across $(S, \overline{S})$. In Figure 2 we consider a formulation of the dual of this LP.

$$\begin{array}{llll}
 & \text{Maximize} & \sum_{i \in S} y_i - \sum_{j \in \overline{S}} y_j & \\
\text{Subject to} & & & \\
 & \forall i \in S, j \in \overline{S}, & y_i - y_j & \leq c_{ij} \\
 & \forall i \in V, & y_i \in \mathbb{R} &
\end{array}$$

Figure 2: The dual of the LP for Bipartite Min-Cost Matching

A feasible solution for this LP can be seen an embedding $\{y_i\}_{i\in[n]}$ of $[n]$ on the real line such that no pair i, j with $i \in S$ and $j \in \overline{S}$ and $y_i \geq y_j$ can be further away in ℓ_1 distance than its ℓ_2^2 distance in the hypercube embedding U_d. We now prove the following two properties of solutions to the dual LP:

1. If $\{y_i\}_{i\in[n]}$ is a feasible solution of value Y, then for any $c \in \mathbb{R}$, $\{y'_i = y_i + c\}_{i\in[n]}$ is a feasible solution of value $Y' = Y$.

2. In any optimal dual solution, we must have, for all pairs $i, j \in [n]$, $|y_i - y_j| \leq c_{ij} = \|\mathbf{u}_i - \mathbf{u}_j\|^2$.

Proof of Property 1: The shifted solution is feasible as for all $i \in S, j \in \overline{S}$:

$$y'_i - y'_j = y_i + c - y_j - c = y_i - y_j \leq c_{ij}$$

The value of this solution is:

$$Y' = \sum_{i\in S} y'_i - \sum_{j\in\overline{S}} y'_j = \sum_{i\in S}(y_i + c) - \sum_{j\in\overline{S}}(y_j + c)$$
$$= \sum_{i\in S} y_i + \frac{cn}{2} - \sum_{j\in\overline{S}} y_j - \frac{cn}{2} = \sum_{i\in S} y_i - \sum_{j\in\overline{S}} y_j = Y$$

Proof of Property 2: Notice that the costs c_{ij}'s respect the triangle inequality as the ℓ_2^2 distance on the hypercube is a metric. To prove the statement, we need to handle the three remaining cases:

1. $i \in S, j \in S$. Assume $y_i \geq y_j$ without loss of generality. As the solution is optimal, it is not possible to increase y_j to obtain a larger dual objective. This implies that there must exist $k \in \overline{S}$ such that $y_j - y_k = c_{jk}$. But we must have $c_{ik} \geq y_i - y_k = (y_i - y_j) + (y_j - y_k) = y_i - y_j + c_{jk}$. As $c_{ik} \leq c_{ij} + c_{jk}$, we have $y_i - y_j \leq c_{ij}$.

2. $i \in \overline{S}, j \in \overline{S}$. This is handled as in the previous case.

3. $i \in S, j \in \overline{S}$ such that $y_j \geq y_i$. Because the solution is optimal there must exists $j' \in S$ and $i' \in \overline{S}$ such that $y_{j'} - y_j = c_{j'j}$ and $y_i - y_{i'} = c_{ii'}$. But, by the dual constraint, we must have $c_{i'j'} \geq y_{j'} - y_{i'} = (y_{j'} - y_j) + (y_j - y_i) + (y_i - y_{i'}) = c_{j'j} + y_j - y_i + c_{ii'}$. By triangle inequality, $c_{i'j'} \leq c_{j'j} + c_{ji} + c_{ii'}$, so that $y_j - y_i \leq c_{ji}$ as required.

Application of vertex iso-perimetry of H_d: Now we use these properties of an optimal dual solution together with the vertex iso-perimetry of the hypercube to obtain an upper bound on the dual optimal. By Property 1, we can translate any optimal dual solution preserving optimality. Hence, we may consider an optimal solution $\{y_i\}_{i\in[n]}$ such that at most $\frac{n}{2}$ vertices are mapped to positive value and at most $\frac{n}{2}$ are mapped to negative values. Notice that, as $\max_{i,j} \|\mathbf{u}_i - \mathbf{u}_j\|^2 = 4$, we have $y_k \in [-4, 4]$ for all $k \in [n]$. Now define sets $R_1, \ldots, R_{4d} \subseteq [n]$ as follows:

$$R_i := \left\{k \in [n] : y_k \in \left(\frac{i-1}{d}, \frac{i}{d}\right]\right\}.$$

Similarly, for the negative side we can define $L_1, \ldots, L_{4d}$:

$$L_i := \left\{k \in [n] : y_k \in \left[-\frac{i}{d}, -\frac{i-1}{d}\right)\right\}.$$

We also define $A_i := \bigcup_{k=i}^{4d} R_i$ and $B_i := \bigcup_{k=i}^{4d} L_i$. By our assumption on $\{y_i\}_{i\in[n]}$ we know that, for all i, $|A_i|, |B_i| \leq \frac{n}{2}$.

Consider now any $k \in A_i$ for $i \geq 2$. Consider any $h \notin A_i$ such that $\|\mathbf{u}_h - \mathbf{u}_k\|^2 = \frac{1}{d}$, i.e., $\mathbf{u}_h$ is a neighbor of $\mathbf{u}_k$ in the hypercube graph. Notice that h must lie in L_{i-1}, as, by Property 2, $|y_k - y_h| \leq \frac{1}{d}$ and $h \notin A_i$. Hence, all vertices which are outside of A_i and adjacent to A_i in the hypercube must belong to L_{i-1}. Because $|A_i| \leq \frac{n}{2}$, by the vertex iso-perimetry of the hypercube, there are at least $\gamma(H_d)|A_i|$ such vertices and, for $i \geq 2$:

$$|L_{i-1}| \geq \Omega\left(\frac{1}{\sqrt{d}}\right)|A_i|.$$

This implies that for $i \geq 2$,

$$|A_{i-1}| \geq \left(1 + \Omega\left(\frac{1}{\sqrt{d}}\right)\right)|A_i|.$$

Since $|A_1| \leq \frac{n}{2}$,

$$|A_i| \leq \frac{n}{2}\left(1 + \Omega\left(\frac{1}{\sqrt{d}}\right)\right)^{-(i-1)}.$$

The same reasoning can be applied to B_i to deduce that

$$|B_i| \leq \frac{n}{2}\left(1 + \Omega\left(\frac{1}{\sqrt{d}}\right)\right)^{-(i-1)}.$$

Now notice that the cost of the dual solution $\{y_k\}_{k\in[n]}$ is upper bounded by

$$\begin{aligned}
\frac{1}{d}\left(\sum_{i=1}^{4d} i|L_i| + \sum_{i=1}^{4d} i|R_i|\right) &\leq \frac{1}{d}\left(\sum_{i=1}^{4d}|A_i| + \sum_{i=1}^{4d}|B_i|\right) \\
&= \frac{n}{d}\sum_{i=1}^{4d}\left(1 + \Omega\left(\frac{1}{\sqrt{d}}\right)\right)^{-(i-1)} \\
&= \frac{n}{d}O\left(\sqrt{d}\right) = O\left(\frac{n}{\sqrt{d}}\right).
\end{aligned}$$

But, by strong duality, the primal optimum equals the the dual optimum. Hence, we complete the proof by noticing that

$$\sum_{\{i,j\}\in M} \|\mathbf{u}_i - \mathbf{u}_j\|^2 = O\left(\frac{n}{\sqrt{d}}\right).$$

□

5. OPEN PROBLEMS

The main remaining open question is whether it is possible to construct a cut player running in almost linear time and achieving a number of rounds to expansion ratio of $O(\sqrt{\log n})$. Another direction of study is to try to reduce the running time of the algorithms presented in this paper by eliminating some of the polylog factors: this may help in making these algorithms competitive against the best heuristics for the SPARSEST CUT problem. Finally, the Main Lemma used in the proof of lower bound raises the following interesting combinatorial question, which, to our knowledge, is unresolved: What is the bisection of the hypercube which maximizes the minimum over all pairings of vertices across the bisection of the average Hamming distance between paired vertices?

Acknowledgments

We would like to thank Satish Rao for several illuminating discussions and for sharing his insights into the realm of graph partitioning.

6. REFERENCES

Arora, S., Hazan, E., and Kale, S. (2004a). $O(\sqrt{\log n})$ approximation to sparsest cut in $\tilde{O}(n^2)$ time. *Proceedings, IEEE Symposium on Foundations of Computer Science*, 00:238–247.

Arora, S. and Kale, S. (2007). A combinatorial, primal-dual approach to semidefinite programs. In *STOC '07: Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, pages 227–236, New York, NY, USA. ACM.

Arora, S., Rao, S., and Vazirani, U. (2004b). Expander flows, geometric embeddings and graph partitioning. In *STOC '04: Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pages 222–231, New York, NY, USA. ACM.

Bhatia, R. (1996). *Matrix Analysis (Graduate Texts in Mathematics)*. Springer.

Chung, F. R. (1997). *Spectral Graph Theory (CBMS Regional Conference Series in Mathematics, No. 92)*. American Mathematical Society.

Devanur, N. R., Khot, S. A., Saket, R., and Vishnoi, N. K. (2006). Integrality gaps for sparsest cut and minimum linear arrangement problems. In *STOC '06: Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 537–546, New York, NY, USA. ACM.

Goldberg, A. V. and Rao, S. (1998). Beating the flow decomposition barrier. *Journal of the ACM (JACM)*, 45:783–797.

Khandekar, R., Rao, S., and Vazirani, U. (2006). Graph partitioning using single commodity flows. In *STOC '06: Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 385–390, New York, NY, USA. ACM.

Khandekar, R. M., Khot, S., Orecchia, L., and Vishnoi, N. K. (2007). On a cut-matching game for the sparsest cut problem. Technical Report EECS-2007-177, EECS Department, University of California, Berkeley, CA.

Leighton, T. and Rao, S. (1999). Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms. *J. ACM*, 46(6):787–832.

Papadimitriou, C. H. and Steiglitz, K. (1982). *Combinatorial Optimization: Algorithms and Complexity*. Prentice-Hall.

Shmoys, D. (1996). Cut problems and their application to divide and conquer. In Hochbaum, D., editor, *Approximation algorithms for NP-hard problems*, pages 192–235. PWS Publishing Co., Boston, MA, USA.

APPENDIX

A. MATRIX INEQUALITIES

The following facts follow from the definition of the trace function. They are proved in [Bhatia, 1996].

FACT A.1 (CYCLIC SHIFT INVARIANCE OF TRACE). *Let $X, Y \in \mathbb{R}^{n\times n}$. Then,* $\mathrm{Tr}[XY] = \mathrm{Tr}[YX]$.

THEOREM A.2 (SYMMETRIC REARRANGEMENT). *Let $X, Y \in \mathbb{R}^{n\times n}$ be symmetric matrices. Then for any positive integer k,*

$$\mathrm{Tr}\left[(XYX)^{2^k}\right] \le \mathrm{Tr}\left[X^{2^k}Y^{2^k}X^{2^k}\right].$$

The following claim uses the fact that, for a matching M, $M^2 = I$ to simplify the expression for a power of a lazy random walk across a single matching.

FACT A.3. *Let* $\lambda := \frac{1}{2} - \frac{1}{2}\left(1 - \frac{2}{d+1}\right)^{4d+4}$. *Then,* $N_{t+1}^{4d+4} = I - \lambda(I - M_{t+1})$.

PROOF. For this proof, let $N := N_{t+1}, M := M_{t+1}$. Note that since $M^2 = I$, $(I-M)^2 = 2(I-M)$. Hence, $(I-M)^j = 2^{j-1}(I-M)$ for $j \ge 1$. We use this observation below to complete the proof of the fact.

$$\begin{aligned}
N^{4d+4} &= \left(I - \frac{1}{d+1}(I-M)\right)^{4d+4} \\
&= I + \sum_{j=1}^{4d+4} (-1)^j \frac{1}{(d+1)^j}\binom{4d+4}{j}(I-M)^j \\
&= I + \frac{I-M}{2}\sum_{j=1}^{4d+4} (-1)^j \frac{2^j}{(d+1)^j}\binom{4d+4}{j} \\
&= I - \frac{I-M}{2}\left(1 - \left(1 - \frac{2}{d+1}\right)^{4d+4}\right)
\end{aligned}$$

□

The following fact is a matrix version of the simple scalar inequality $\mathrm{e}^{-\varepsilon x} \le (1 - (1-\mathrm{e}^{-\varepsilon})x)$ for all $x \in [0,1]$ and $\varepsilon \in (0,1)$.

FACT A.4. *Let $X \in \mathbb{R}^{n\times n}$ be symmetric matrix such that $0 \preceq X \preceq I$ and $\varepsilon \in (0,1)$. Then,*

$$e^{-\varepsilon X} \preceq (I - (1-\mathrm{e}^{-\varepsilon})X).$$

B. PROJECTION LEMMA

The results in this section essentially appear in [Khandekar et al., 2006]. We include them here for completeness.

FACT B.1 (GAUSSIAN BEHAVIOR OF PROJECTIONS). *If $\mathbf{v}$ is a vector of length l in $\mathbb{R}^m$ and $\mathbf{u}$ is a random vector in $\mathbb{S}^{m-1}$. Then*

1. $\mathbf{E}_{\mathbf{u}}\left[\langle \mathbf{v}, \mathbf{u}\rangle^2\right] = \frac{l^2}{m}$,
2. *For* $x \le m/16$, $\mathbf{Pr}_{\mathbf{u}}\left[\langle \mathbf{v}, \mathbf{u}\rangle^2 \ge xl^2/m\right] \le e^{-x/4}$.

LEMMA B.2. *Let $\{\mathbf{v}_i\}_{i=1}^n$ be vectors in $\mathbb{R}^{n-1}$ such that*

1. $\sum_i \mathbf{v}_i = 0$.
2. *For all i, $\|\mathbf{v}_i\|^2 \le B$ for some constant B.*
3. $\Phi := \sum \|\mathbf{v}_i\|^2 \ge \frac{1}{\mathrm{poly}(n)}$.

Let $\mathbf{r}$ be a random unit vector in $\mathbb{R}^{n-1}$ and for all i set $u_i := \langle \mathbf{v}_i, \mathbf{r}\rangle$. Let S be the partition of $[n]$ such $|S| = n/2$ and for all $i \in S$ and $j \in \overline{S}$ $u_i \ge u_j$. Consider any matching M of the indices $[n]$ across $(S, \overline{S})$. Then,

$$\mathbf{E}_{\mathbf{r}}\left[\sum_{\{i,j\}\in M} \|\mathbf{v}_i - \mathbf{v}_j\|^2\right] = \Omega\left(\frac{1}{\log n}\right)\Phi.$$

PROOF. Define the event

$$\mathcal{E}_{ij} := \left\{ (u_i - u_j)^2 \leq \frac{c \log n}{n-1} \|\mathbf{v}_i - \mathbf{v}_j\|^2 \right\}$$

for some constant $c > 0$. Let $\mathcal{E} := \bigcap_{i,j} \mathcal{E}_{ij}$. By the Fact B.1 we have that $\mathbf{Pr}[\overline{\mathcal{E}}_{ij}] \leq n^{-c/4}$. Hence, by a union bound, $\mathbf{Pr}[\overline{\mathcal{E}}] \leq n^{-c/4+2}$. Then:

$$\begin{aligned} & \mathbf{E_r} \left[\sum_{\{i,j\} \in M} \|\mathbf{v}_i - \mathbf{v}_j\|^2 \right] \geq \\ \frac{n-1}{c \log n} \; & \mathbf{E_r} \left[\sum_{\{i,j\} \in M} (u_i - u_j)^2 \mid \mathcal{E} \right] \cdot \mathbf{Pr}[\mathcal{E}]. \end{aligned}$$

Let a be the real number such that $u_i \geq a \geq u_j$ for all $i \in S, j \in \overline{S}$. Hence, for $\mathbf{u} := (u_1, \ldots, u_n)$, $\sum_{i=1}^n u_i = \sum_{i=1}^n \langle \mathbf{v}_i, \mathbf{r} \rangle = \left\langle (\sum_{i=1}^n \mathbf{v}_i), \mathbf{r} \right\rangle = 0$. Hence,

$$\begin{aligned} \sum_{\{i,j\} \in M} (u_i - u_j)^2 & \geq \sum_{i=1}^n (u_i - a)^2 \\ & = \|\mathbf{u}\|^2 - 2a \left(\sum_{i=1}^n u_i \right) + na^2 \\ & = \|\mathbf{u}\|^2 + na^2 \geq \|\mathbf{u}\|^2. \end{aligned}$$

Hence, we have

$$\mathbf{E_r} \left[\sum_{\{i,j\} \in M} \|\mathbf{v}_i - \mathbf{v}_j\|^2 \right] \geq \mathbf{E_r} \left[\|\mathbf{u}\|^2 \mid \mathcal{E} \right] \cdot \mathbf{Pr}[\mathcal{E}].$$

To obtain a lower bound on the r.h.s., notice that

$$\mathbf{E_r}[\|\mathbf{u}\|^2] = \sum_{i=1}^n \mathbf{E_r}[u_i^2] = \sum_{i=1}^n \frac{\|\mathbf{v}_i\|^2}{n-1} = \frac{\Phi}{n-1}.$$

Since $|u_i|^2 \leq B$ for all i, $\|\mathbf{u}\|^2 \leq Bn$. Hence,

$$\begin{aligned} \mathbf{E_r} \left[\|\mathbf{u}\|^2 \mid \mathcal{E} \right] \cdot \mathbf{Pr}[\mathcal{E}] & \geq \mathbf{E_r}[\|\mathbf{u}\|^2] - \mathbf{E_r} \left[\|\mathbf{u}\|^2 \mid \overline{\mathcal{E}} \right] \cdot \mathbf{Pr}[\overline{\mathcal{E}}] \\ & \geq \frac{\Phi}{n-1} - Bn^{-c/4+3}. \end{aligned}$$

Since $\Phi \geq \frac{1}{\mathrm{poly}(n)}$, by picking c to be a large enough constant, one obtains

$$\mathbf{E_r} \left[\|\mathbf{u}\|^2 \mid \mathcal{E} \right] \cdot \mathbf{Pr}[\mathcal{E}] \geq \frac{\Phi}{2(n-1)}.$$

Hence, we can complete the proof of the lemma by concluding that

$$\mathbf{E_r} \left[\sum_{\{i,j\} \in M} \|\mathbf{v}_i - \mathbf{v}_j\|^2 \right] \geq \frac{\Phi}{2c \log n}.$$

□