Spectral Embedding of k-Cliques, Graph Partitioning and k-Means

Pranjal Awasthi
Rutgers University
pa336@cs.rutgers.edu

Ravishankar
Krishnaswamy
Microsoft Research
ravishan@cs.cmu.edu

Moses Charikar
Stanford University
moses@cs.stanford.edu

Ali Kemal Sinop
Simons Institute for the Theory
of Computing
asinop@cs.cmu.edu

ABSTRACT

We introduce and study a new notion of graph partitioning, intimately connected to spectral clustering and k-means clustering. Formally, given a graph G on n vertices, we ask to find a graph H that is the union of k cliques on n vertices, such that $L_G \succeq \lambda L_H$ where λ is maximized. Here L_G and L_H are the (normalized) Laplacians of the graphs G and H respectively. Informally, our graph partitioning objective asks for the *optimal spectral simplification* of a given graph as a disjoint union of k cliques.

We justify this objective function in several ways. First and foremost, we show that a commonly used spectral clustering algorithm *implicitly optimizes* this objective function, up to a factor of $O(k)$. Using this connection, we immediately get an $O(k)$ -approximation algorithm to our new objective function by simply using the spectral clustering algorithm on G . Next, we demonstrate another application of our objective function: we use it as a means to proving that simple spectral clustering algorithms can solve some well-studied graph partitioning problems (such as partitioning into expanders). Additionally, we also show that (a relaxation of) this optimization problem naturally arises as the dual problem to the question of finding the worst-case integrality gap instance for the classical k -means SDP. Finally, owing to these close connection between some classical clustering techniques (such as k -means and spectral clustering), we argue that a more complete understanding of this optimization problem could lead to new algorithmic insights and techniques for the area of graph partitioning.

1. INTRODUCTION

Graph partitioning is a fundamental problem in computer science with various applications in computer vision, machine learning and bioinformatics, among others. Such problems have been studied extensively in theoretical computer science (e.g. sparsest cut, multicut, multiway cut, etc) [28, 16, 18, 13]. Designing a good objective function for graph partitioning is a balancing act between several, often conflicting, desiderata. For instance, the objective function formulation should be succinct and mathematically tractable. In traditional graph partitioning objectives, this is captured by formulating the function as a combinatorial measure of the number of edges cut. From a practitioner's point of view, the hope is that the study of the objective function will naturally lead to good algorithms which succeed on practical instances of the problem, if not on all. In this paper, we introduce and study a new spectral notion of graph partitioning, intimately connected to spectral clustering and k -means clustering. We make the claim, and formally justify, that our objective function has many of the desirable properties mentioned above.

Formally, we study the following problem (see Section 2 and 3 for definitions of these concepts): Given a graph G with normalized Laplacian L_G , the goal is to find a partition of the vertices $\Gamma = \{C_1, \dots, C_k\}$ into k normalized cliques with associated Laplacian matrix K_Γ whose (i, j) th entry is $1 - \frac{1}{|C_\ell|}$ if $i = j \in C_\ell$, $\frac{1}{|C_\ell|}$ if both $i, j \in C_\ell$, and 0 otherwise. The objective is to find a partition Γ such that $L_G \succeq \lambda K_\Gamma$ and λ is maximized. Informally, our graph partitioning objective asks for the optimal spectral simplification of the graph as a disjoint union of cliques. It can be seen as a spectral variant of the problem of graph partitioning into expanders (where expansion is just measured w.r.t the induced graph in each piece of the decomposition); in contrast, our objective function also captures the edges going across pieces, and can be loosely viewed as partitioning a graph into weakly embedded expanders. Moreover, our notion of embedding cliques is a one-sided version of the notion of spectral similarity of graph Laplacians used in the recent influential sequence of work on spectral sparsification [38, 36, 9]. It is also akin to the notion of spectrally thin trees that have been studied before [19], except that we look for a decomposition into cliques that is spectrally thin, instead of trees. Finally, we would like to mention that a very similar notion has been used in [8] where the authors address the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.

ITCS'16, January 14–16, 2016, Cambridge, MA, USA.

© 2016 ACM. ISBN 978-1-4503-4057-1/16/01 ...\$15.00.

DOI: <http://dx.doi.org/10.1145/2840728.2840751>.

question of whether a metric space which does not coarsely embed into a Hilbert space necessarily contains a weakly embedded expander.

1.1 Why Study Another Objective Function?

A) Precise Equivalence Between Partitioning and Clustering. Our first main result is an equivalence between *optimizing our objective and optimizing the clustering cost¹ of the “resistive embedding” of the given graph*. That is, we show that a partitioning of the vertices into k pieces is good for our objective function if and only if the same partitioning is a good clustering of the resistive embedding vectors (which are the column vectors of $L^{\dagger/2}$). Our result also suggests that a natural variant of spectral clustering is a good algorithm for our objective function. Typically, the graph Laplacian is projected to the top k eigenvectors, and then the resulting row vectors are clustered. In our case, we project the resistive embedding vectors into a k dimensional space, and run a standard clustering algorithm on these vectors. Using the equivalence, we show that the resulting graph partitioning is a good solution — an $O(k)$ approximation — for our new objective function, thereby giving us provable guarantees, and more intuition, on the types of clusterings our algorithm finds.

We next show that we can improve the provable guarantees, if we are allowed to output a distribution over partitionings. In Section 4, we show how we can use the matrix multiplicative weights algorithm of Arora and Kale [4] to obtain a *distribution of partitions* that, in expectation, can embed into G with a constant factor approximation in terms of our objective function. Interestingly, we obtain this result by exhibiting a connection between our problem and that of finding the worst-case integrality gap instance for a natural SDP relaxation for the classical k -means problem.

B) New Smoothing Step. While effective resistances (i.e. the values) themselves have been used [36] in spectral sparsification, and the resistive distance embedding has been used in graph partitioning heuristics [24, 25, 34], we believe that this is the first provable justification of their use in graph partitioning algorithms. Moreover, by understanding how our algorithm works on our objective function, we also suggest an additional iterative step in the process, called *smoothing*: iteratively *average* each vector with the vectors corresponding to the neighboring vertices in the original graph, and cluster the smoothed vectors. We are able to get provably better guarantees in certain situations, and also empirically get better guarantees in some large datasets. Going forward, we believe that an analysis of multiple steps of smoothing will lead to improved guarantees for our objective function in theory and in practice.

C) Applications. Then, in Section 5, in order to illustrate the power of our new notion of graph decomposition, we show that approximation algorithms for our new objective can be used as a black box to find a partition of a graph into k pieces with sufficiently large gaps between internal and external conductance. Partitions with such properties were studied by [22] (see also [17]); they also arise in the work of Ng, Jordan and Weiss [33]. Given the promise that

¹Formally, the equivalence holds if the metric used to evaluate a clustering is the spectral radius objective, i.e., $\min_C \|A - C\|_2^2$ where the rows of A are the data points, and the rows of C are the cluster centers each data point is assigned to; in contrast, $\|A - C\|_F^2$ is the k -means cost.

such a partition exists, any good algorithm for our objective directly produces a partition that is guaranteed to have small symmetric difference with this unknown partition. In particular, we show that our variant of spectral clustering (using effective resistance vectors) works! Finally in Section 6, we further strengthen the connection to spectral clustering by showing that the most popular variant of this approach also gives guarantees for our new objective function. Combining the above connections and applications, we thus believe our new graph partitioning objective function and associated algorithm makes a meaningful contribution to the existing body of work trying to motivate and justify spectral clustering algorithms, which we detail in Section 1.3.

1.2 Related Work

Graph partitioning objectives have received a lot of attention in theoretical computer science. Most notable among them is the sparsest cut objective [28, 7, 6] which attempts to find a 2-partition with minimum conductance. Conductance is also closely related to the second smallest eigenvalue of the graph Laplacian [12, 1, 3]. Generalizations of this to k -partitions aim at finding k clusters with small conductance [27, 30] as well as high conductance within each cluster [22, 17]. These works exploit gaps in the higher eigenvalues of the graph Laplacian to design algorithms which achieve a good clustering. The most relevant to our work is the result of [17] which uses the spectrum of the graph Laplacian to design a good clustering if one exists (See Section 6 for a result of similar flavor using our new objective). However as opposed to our algorithm, the algorithm proposed in [17] does not fit naturally into the framework of spectral clustering as used in practice. See the next section for details. A sequence of work has also shown that spectral partitioning works well in geometric settings [37, 23, 10].

1.3 Spectral clustering and k -means

Spectral clustering approaches in practice consist of two main steps: a) transforming the original set of points to be clustered using the spectrum of the data matrix (or an associated matrix), and b) optimizing the k -means objective in the new space to obtain a k -partition. The choice of using the k -means objective in the new space is not arbitrary and several prior works have hinted at a deeper connection. The simplest scenario consists of the original set of points in $\mathbb{R}^d$ such that there is a true unknown k -partition with the mean vectors of each partition being far away from each other. In this case, Kumar and Kannan [26] show that performing a linear embedding of the points onto the span of the top k singular vectors of the data matrix and optimizing for k -means in the new space leads to good approximations to the true mean vectors. This approach also has applications in the planted partition model of Mcsherry [31].

In many real world applications however, one does not have a true clustering with well separated mean vectors. In such scenarios spectral clustering approaches prove to be immensely successful again. The idea is to now look at non-linear embeddings. This approach pioneered by the work of Weiss, Meila and Shi and Ng, Jordan and Weiss [40, 32, 33] looks at the spectrum of the Laplacian L of the affinity graph. The affinity graph A is an $n \times n$ matrix where the entry $A(i, j) \propto \exp(-\|x_i - x_j\|^2)$. The k dimensional embedding of a point is obtained by using the corresponding coordinates in the top k eigenvectors of L . Ng, Jordan and Weiss

[33] justify this approach by showing that if the true clustering is well separated in the sense that it has densely connected clusters with sparse connections across them then in the new space the optimal k means clustering will have a small cost. In many instances, it has been observed that replacing the Laplacian based embedding with the resistive embedding (using the eigen vectors of $L^{\frac{1}{2}}$) works better as it induces a stronger block structure in the induced distance metric [34]. As an application of our result we strengthen this connection by showing in Section 5 how to recover a good point wise approximation to a well separated clustering using our algorithm which is based on resistive embeddings followed by solving k -means. Spectral approaches have also been used with great success in partitioning problems where one is naturally interested in optimizing a cut based objective function [32, 35, 42]. It has also been observed that such approaches have connections, and in some cases are equivalent to optimizing a kernel k -means objective in the original space [14, 15]. Finally, our approach of modifying the metric for the clustering problem (via our smoothing step) can be loosely connected with that of learning the metric for clustering [41].

2. PRELIMINARIES

2.1 Linear Algebra

Let $\mathbb{R}^{m \times n}$ be the set of all $m \times n$ real matrices; $\mathbb{S}^m$ be the set of $m \times m$ symmetric matrices; $\mathbb{S}_+^m \subset \mathbb{S}^m$ be the set of $m \times m$ positive semidefinite (PSD) matrices. Alternatively, we will use $A \succeq 0$ to denote $A \in \mathbb{S}_+^m$. The $m \times m$ identity matrix is denoted by I_m . For any subset $C \subseteq [n]$, $\mathbf{e}_C \in \mathbb{R}^n$ is the indicator vector for C so that $(\mathbf{e}_C)_i = 1$ iff $i \in C$; 0 else. Define $J_C \stackrel{\text{def}}{=} \mathbf{e}_C \mathbf{e}_C^T$ as the matrix whose entries are 0 except the principal minor on C , which is all-1's. For any $A \in \mathbb{R}^{m \times n}$, we will use $A^\Pi, A^\perp \in \mathbb{S}_+^m$ to refer to the projection matrix onto the column span of A and its orthogonal complement, respectively. Note that $A^\Pi + A^\perp = I_m$. Finally, we will use $\sigma_{\max}(A)$ and $\sigma_{\min}(A)$ to denote the maximum and minimum singular values of A . Given $A \in \mathbb{S}^m$, we define $A^\dagger$ to be the pseudo-inverse of A ; and $\lambda_{\min}(A)$, $\lambda_{\max}(A)$ as the the minimum and maximum eigenvalues of A , respectively: $\lambda_{\min}(A) \stackrel{\text{def}}{=} \min_{q \neq 0} \frac{q^T A q}{\|q\|^2}$; $\lambda_{\max}(A) \stackrel{\text{def}}{=} \max_{q \neq 0} \frac{q^T A q}{\|q\|^2}$. Likewise, given $A, B \in \mathbb{S}_+^m$, let $\lambda_{\min}(A, B)$ be the minimum generalized eigenvalue of A and B , $\lambda_{\min}(A, B) \stackrel{\text{def}}{=} \min_{q: Bq \neq 0} \frac{q^T A q}{q^T B q}$. An equivalent definition is $\lambda_{\min}(A, B) \stackrel{\text{def}}{=} \max\{\lambda \mid A \succeq \lambda B\}$.

2.2 Graphs

Given a graph G with adjacency matrix A , let D be the diagonal matrix with $D_{ii} = d_i$ where d_i is the weighted degree of node i . We define the Laplacian matrix of G as $L_G = D - A$. We define the normalized adjacency matrix to be $D^{-1/2} A D^{-1/2}$, and the normalized Laplacian to be $I - D^{-1/2} A D^{-1/2}$. Also, let $K \in \mathbb{S}_+^n$ be the Laplacian matrix of a normalized clique, $K \stackrel{\text{def}}{=} I - \frac{1}{n} J$.

2.3 Partitions

Define $\Gamma_k(n)$ to be the set of all proper k -partitions of $[n]$:

$$\Gamma_k(n) \stackrel{\text{def}}{=} \left\{ \Gamma = \{S_1, \dots, S_k\} \mid S_i \neq \emptyset, S_1 \uplus \dots \uplus S_k = [n] \right\}.$$

We can construct an associated graph for Γ by placing a normalized clique on each $S \in \Gamma$. We will use K_Γ to denote the Laplacian matrix of corresponding graph. Finally, given a subset S , we use $\mathbf{e}_S$ to denote the indicator vector for S so that $[\mathbf{e}_S]_i = 1$ iff $i \in S$; we also use $\mathbf{e}$ for the all-1's vector.

3. SPECTRAL EMBEDDING OF CLIQUES

We now formally define the spectral embedding we study in this work. Given as input a connected graph G with the corresponding normalized Laplacian matrix L_G , the goal is to find a partition Γ into k normalized cliques while optimizing the following objective:

$$\max \lambda \text{ st } L_G \succeq \lambda K_\Gamma.$$

Before we get into the technical details of our algorithm, we present some simple observations about the objective function. To begin with, note that $\lambda \geq 0$ because $L \succeq 0$. Next observe that when $k = 1$, the optimal value OPT of our problem will be the second smallest eigenvalue of the Laplacian, which is the algebraic expansion of the graph (in words, it captures how well we can embed a complete graph into G). Thus our objective can be seen as a generalization of algebraic expansion to k -way partitionings. It is also a smooth transition between completely combinatorial ways of measuring a good k -way partitioning (such as normalized cuts), which are often hard to compute, and purely algebraic ways (such as measuring the k^{th} smallest eigenvalue), which are more difficult to understand and relate to k -way partitionings. Indeed, if the graph G is a disjoint union of k pieces, then OPT is the smallest algebraic expansion of the different pieces. Moreover, as mentioned above, OPT is also related to λ_{k+1} of the graph — by relaxing K_Γ to be any $n - k$ dimensional projection, we get that $\text{OPT} \leq \lambda_{k+1}$. It is also easy to show that $\text{OPT} \geq \lambda_2$. To see this, fix any partitioning Γ and only consider vectors x such that $\sum_i x_i = 0$ for each cluster in Γ . Then

$$\frac{x^T L_G x}{x^T K_\Gamma x} = \frac{x^T L_G x}{x^T x} = \lambda_2$$

Finally, OPT only increases with larger k , because $K_\Gamma \succeq K_{\Gamma'}$ whenever Γ' is a refinement of Γ .

In order to understand this objective better consider the following simple scenario first. Given a Laplacian matrix L and k -partition, for some λ , how can we even verify $L \succeq \lambda K_\Gamma$? From an algorithmic perspective, this is easy: Compute the minimum generalized eigenvalue of L and K_Γ and compare it against λ . From an analysis perspective on the other hand, this yields little to no insight for us on how to certify that the graph admits a good spectral k -clique-embedding. There is no analogue of Cheeger's inequality for the generalized case, and indeed there is strong evidence to believe none exists [39]. As a result, we first present a characterization of k -clique embeddability and, using this, derive a sufficient condition for the existence of one. This condition then leads to a natural algorithm for finding such clustering. We would like to mention that the above formulation has also been studied in the context of expansion relative to a sequence of subgroups [8]. This notion of relative expansion has also been used to show that certain finite semi-direct products form an expander [2].

Good and bad cases: As mentioned above, if the graph is a disjoint union of k pieces, then the optimal value of λ will be the smallest algebraic expansion of any piece. It is also an

easy calculation to show that the partitioning corresponding to the optimal λ will correspond to each piece being a clique. In general, our objective function will represent well cases when there is a ground truth partitioning with internal connectivity more than the external connectivity of each piece. We formalize this in Section 5. On the other hand, if there is no “clear” k -partitioning of the graph, our objective may fail to capture the instance. Consider, for example the case of grid graphs with $k = 2$. From the above discussion, we know that the optimal value of our objective will lie between λ_2 and λ_3 . However, for grid graphs these two values are the same and hence according to our objective, any 2-partitioning will be equally good!

3.1 Necessary and Sufficient Conditions

Our starting point is relating the minimum generalized eigenvalue to a standard maximum eigenvalue problem. While there are other bounds for this [11], we prove one most amenable to us.

LEMMA 3.1. *Given a connected graph G and a k -partition Γ of V , we have $\max\{\lambda \mid L_G \succeq \lambda K_\Gamma\} = \frac{1}{\lambda_{\max}(K_\Gamma L_G^\dagger K_\Gamma)}$. Furthermore the maximum eigenvector p of $K_\Gamma L_G^\dagger K_\Gamma$ satisfies $K_\Gamma p = p$.*

For simplicity in notation, we define $L \stackrel{\text{def}}{=} L_G$, $M \stackrel{\text{def}}{=} K_\Gamma$ and $K \stackrel{\text{def}}{=} L \cdot L^\dagger$. Additionally, let $\lambda_{\min}(L, M) = \lambda_{\min}$ and $\lambda_{\max}(ML^\dagger M) = \lambda_{\max}$. Since any cluster $C \in \Gamma$ is connected in G , $\lambda_{\min} > 0$.

The first identity (i) follows immediately from the definition of $\lambda_{\min}$: $\lambda_{\min}(L, M) = \min_{q: Mq \neq 0} \frac{q^T L q}{q^T M q}$. So for all q , $q^T L q \geq \lambda_{\min} q^T M q$ implying $L \succeq \lambda_{\min} M$. We prove (ii) in two parts.

PROOF OF $\lambda_{\max} \geq 1/\lambda_{\min}$. Given any g such that $Lg = \lambda_{\min} M g$, we can assume $Kg = g$, since Kg also satisfies this identity. Therefore:

$$g = Kg = L^\dagger L g = \lambda_{\min} L^\dagger M g.$$

Multiplying with $\frac{1}{\lambda_{\min}} M$ on both sides,

$$\frac{1}{\lambda_{\min}} M g = M L^\dagger M g.$$

For $p \stackrel{\text{def}}{=} M g$, together with the fact that $M^2 = M$, this becomes

$$\frac{1}{\lambda_{\min}} p = M L^\dagger M p.$$

Therefore $\lambda_{\max}(M L^\dagger M) \geq \frac{1}{\lambda_{\min}}$. $\square$

PROOF OF $\lambda_{\max} \leq 1/\lambda_{\min}$. We always have $\lambda_{\min} \geq 0$ so we only need to consider the case of $\lambda_{\max} > 0$. Given corresponding eigenvector p with $M(L^\dagger M p) = \lambda_{\max} p$, it is easy to see that $M p = p$. Therefore

$$M L^\dagger p = \lambda_{\max} p.$$

Eigenvalues of a matrix and its transpose are the same, thus

$$(M L^\dagger)^T = L^\dagger M$$

has maximum eigenvalue $\lambda_{\max}$ with eigenvector g , so that $L^\dagger M g = \lambda_{\max} g$. Multiplying both sides by L and observing that $LL^\dagger = K$ with $KM = M$,

$$\lambda_{\max} \cdot L g = LL^\dagger M g = K M g = M g.$$

Hence

$$\lambda_{\min}(L, M) \leq \frac{g^T L g}{g^T M g} = \frac{1}{\lambda_{\max}}.$$

$\square$

Equipped with Theorem 3.1, our goal is now much simpler: Instead of lower bounding the minimum generalized eigenvalue of a pair of matrices, we want to upper bound the maximum eigenvalue of a single matrix. Indeed, the largest eigenvalue of $K_\Gamma L_G^\dagger K_\Gamma$ is equal to the $\|L_G^{\dagger/2} K_\Gamma\|_2^2$, the largest singular value of the matrix. However, $L_G^{\dagger/2} K_\Gamma$ is the matrix $L_G^{\dagger/2} - C_\Gamma$, where we think of the columns of $L_G^{\dagger/2}$ as being the data points in Euclidean space, and C_Γ is the center matrix of the corresponding centroid each point/column is assigned to, depending on the cluster it belongs to in Γ . Therefore, finding a good Γ for our partitioning objective is equivalent to clustering the columns of $L_G^{\dagger/2}$, in the spectral norm sense (note that we want to bound the $\|L_G^{\dagger/2} - C_\Gamma\|_2^2$; in relation, $\|L_G^{\dagger/2} - C_\Gamma\|_F^2$ is the k -means cost of clustering the same points). This is precisely the setting considered in Kumar-Kannan [26, 20], where they give an $O(k)$ approximation for this problem using SVD and k -means clustering. However, we give an alternate proof of this result which is more amenable, and in fact, leads naturally to our smoothing step.

THEOREM 3.2. *Given a connected graph G with normalized adjacency (normalized Laplacian) matrix A (L resp.), suppose Γ is a k -partition. Then, for all $\tau \in \mathbb{Z}_+$, we have:*

$$\lambda_{\min}(L, K_\Gamma) \geq \left\{ \max(\lambda^{-1}, 2\tau) + \text{Tr} \left[K_\Gamma A^\tau (L^\dagger)_k A^\tau \right] \right\}^{-1}.$$

Here λ is the k^{th} smallest eigenvalue of L , and $(L^\dagger)_k$ is the matrix $L^\dagger$ projected onto its top k -eigenvectors.² In particular, for the choice of $\tau \leftarrow 0$:

$$\lambda_{\min}(L, K_\Gamma) \geq \left\{ \frac{1}{\lambda_k} + \text{Tr} [K_\Gamma (L^\dagger)_k] \right\}^{-1}.$$

PROOF. Let λ_i and q_i be the i^{th} smallest eigenvalue and corresponding eigenvector of L . Suppose G has c -connected components. Then $0 = \lambda_1 = \dots = \lambda_c < \lambda_{c+1} \leq \dots \leq \lambda_n \leq 2$. Moreover if we define $K \stackrel{\text{def}}{=} L \cdot L^\dagger$ (the projection matrix onto the complement of connected components of G), then $K = \sum_{j>c} q_j q_j^T$. By our assumption, $K_\Gamma K = K K_\Gamma = K_\Gamma$.

For any $x \in [0, 2]$, $\frac{1}{x} = \frac{1}{1-(1-x)} \leq 2\tau + \frac{(1-x)^{2\tau}}{x}$. Using this, we can now upper bound $L^\dagger$:

$$\begin{aligned} L^\dagger &= \sum_{j>c} \frac{1}{\lambda_j} q_j q_j^T \preceq \sum_{c < j < k} \left(2\tau + \frac{(1-\lambda_i)^{2\tau}}{\lambda_i} \right) q_j q_j^T \\ &\quad + \frac{1}{\lambda_r} \sum_{j \geq r} q_j q_j^T \\ &\preceq \max(2\tau, \lambda_r^{-1}) K + A^\tau (L)_k A^\tau. \end{aligned}$$

$$K_\Gamma L^\dagger K_\Gamma \preceq \max(2\tau, \lambda_r^{-1}) K_\Gamma + K_\Gamma A^\tau (L)_k A^\tau K_\Gamma.$$

²In words, for y_i 's being the columns of $[L^\dagger]_k^{1/2} A^\tau$, a simple calculation shows that the trace term is equal to the k -means cost of clustering these vectors using Γ .

Since $\lambda_{\max}(X + Y) \leq \lambda_{\max}(X) + \lambda_{\max}(Y)$,

$$\begin{aligned} \lambda_{\max}(K_{\Gamma} L^{\dagger} K_{\Gamma}) &\leq \max(2\tau, \lambda_r^{-1}) \underbrace{\lambda_{\max}(K_{\Gamma})}_{=1} \\ &\quad + \underbrace{\lambda_{\max}(K_{\Gamma} A^{\tau} (L)_k A^{\tau} K_{\Gamma})}_{\geq 0} \\ &\leq \max(2\tau, \lambda_r^{-1}) + \text{Tr}(K_{\Gamma} A^{\tau} (L)_k A^{\tau} K_{\Gamma}). \end{aligned}$$

The proof is complete by using the identity from Theorem 3.1. $\square$

3.2 Main Algorithm

The lower bound presented in Theorem 3.2 corresponds to a natural algorithm for finding Γ . We state the algorithm only for connected graphs. The disconnected case can easily be handled by recursing on each component separately. Even though this algorithm has polynomial running time, it can be implemented much more efficiently, and we leave the details to the final version.

Input..

Number of clusters k , normalized adjacency and Laplacian matrices $A, L = I - A$ respectively, and maximum smoothing count $\tau_{\max}$.

Output..

λ and k -partition Γ such that $L \succeq \lambda K_{\Gamma}$.

1. (*Resistive Embedding*) For every $i \in V$, let $Y_i^{(0)} \leftarrow (L^{\dagger/2})_k \mathbf{e}_i$ be the resistive embedding of node i projected onto top k -eigenvalues.
2. (*Best k -Partition So Far*) Let Γ_{best} be an arbitrary k -partition. $\lambda_{\text{best}} \leftarrow 0$.
3. For $\tau \leftarrow 0$ to $\tau_{\max}$ do:

- (a) Find a k -means solution Γ for

$$Y^{(t)} = [Y_i^{(t)} \mid i \in V]$$

$$\lambda_{\text{cur}} \leftarrow \left[\max(2\tau, \lambda_k^{-1}) + \|Y^{(t)} K_{\Gamma}\|_F^2 \right]^{-1}.$$

- (b) If $\lambda_{\text{cur}} > \lambda_{\text{best}}$, then $\Gamma_{\text{best}} \leftarrow \Gamma$, $\lambda_{\text{best}} \leftarrow \lambda_{\text{cur}}$.

- (c) (*Smoothing*) $\forall i \in V$, $Y_i^{(t+1)} \leftarrow \sum_j A_{ij} Y_j^{(t)}$.

In the algorithm above, steps (b) and (c) are optional (our current proof does not use any power derived from these steps); however, we believe that they could help in getting an improved quality of solution w.r.t our objective function. In words, the algorithm (without the smoothing step) simply projects the resistive embedding vectors onto the top k eigenvectors, and runs a k -means algorithm on the projected points.

THEOREM 3.3. (Correctness) *The output of this algorithm, Γ_{best} and λ_{best} , satisfies $L \succeq \lambda_{\text{best}} K_{\Gamma_{\text{best}}}$. Furthermore, we have that $\lambda_{\text{best}} \geq \frac{\lambda_{\text{OPT}}}{O(k)}$, where λ_{OPT} is the optimum.*

PROOF. Correctness follows immediately from Theorem 3.2. For the approximation guarantee, let Γ_{opt} be an optimal solution with $L \succeq \lambda_{\text{OPT}} K_{\Gamma_{\text{opt}}}$. Consider the projection matrix onto top k -eigenvectors of $L^{\dagger}$, Q_k : $Q_k = \sum_{j \leq k} q_j q_j^T$. Note that Q_k commutes with L . Thus, if we multiply the first expression on both sides with $Q_k L^{\dagger/2} Q_k = (Y^{(0)})^T (Y^{(0)})$:

$$Q_k \succeq \lambda_{\text{OPT}} \cdot (Q_k L^{\dagger/2} Q_k) K_{\Gamma_{\text{opt}}} (Q_k L^{\dagger/2} Q_k).$$

Taking the trace, $k \geq \lambda_{\text{OPT}} \|Y^{(0)} K_{\Gamma_{\text{opt}}}\|_F^2$. Therefore at time $\tau = 0$, there exists a k -means solution of cost $\leq \frac{k}{\lambda_{\text{OPT}}}$. Hence the algorithm will find some Γ_{best} with

$$\|Y^{(0)} K_{\Gamma_{\text{best}}}\|_F^2 \leq \frac{O(k)}{\lambda_{\text{OPT}}}.$$

Consequently $\lambda_{\text{best}} \leq \frac{1}{\lambda_k} + \frac{O(k)}{\lambda_{\text{OPT}}} \leq \frac{O(k)}{\lambda_{\text{OPT}}}$ where we used the fact that $\lambda_{\text{OPT}} \leq \lambda_k$. $\square$

Remark: Connection with Spectral Clustering. In machine learning, data mining and similar fields, a very common approach for clustering is to apply k -means on either the resistive embedding, or the embedding obtained by the smallest k -eigenvectors. In these cases, our algorithm above, or more the discussion following Theorem 3.1, could be seen to offer an explanation for *what kind of k -partitions* such methods implicitly seek. For example, if an algorithm of the above types finds a k -partition with small cost in the resistive embedding, then it means the underlying clusters are better connected to each other than across clusters, i.e., they have high value in our objective function. We give a more rigorous connection in Section 6. Given this connection, an intriguing practical problem is whether the applying the smoothing step helps clustering in such domains.

3.3 Smoothing for Multiple Steps

We end this section with a small remark regarding the use of single versus multiple steps of smoothing. While we don't require smoothing for our current guarantee of a factor of $O(k)$, we leave open the question of whether smoothing can provably yield better bounds in practical settings. To support our belief, we empirically demonstrate that the performance of traditional spectral clustering itself improves with a small number of steps of smoothing on large-scale real-world datasets [43].

Dataset	n	k	SC	SC+smoothing
Musk	6598	2	74.63%	83.2%(1)
Gamma	19020	2	69.36%	75.09%(2)
USCI	285779	2	93.9%	93.76%(1)
Segmentation	19020	7	33.23%	37.4%(5)
Poker	1000000	3	47.7%	49.8%(3)

Table 1: The clustering accuracy of the spectral clustering method used in [43] as compared to the same method applied with smoothing. The quantity in bracket denotes the number of rounds of smoothing which provides the best result. The datasets are from the UCI repository [29]. SC stands for Spectral Clustering.

4. SPECTRALLY EMBEDDING A DISTRIBUTION OF PARTITIONINGS

In this section, we show that we can get much better approximation guarantees if we allow ourselves to find a distribution over k -partitions.

THEOREM 4.1. *Given a feasible fractional solution X to the k -partitioning problem, we can efficiently find a distribution $\mathcal{X}$ over k -partitions such that $L_X \succeq \alpha E_{\Gamma \sim \mathcal{X}}[K_\Gamma]$ where $\alpha > \frac{1}{216}$ is a constant.*³

A nice additional property this says is that if X is a fractional solution for the Euclidean k -means problem, then we can round X **obliviously** into an integral clustering *even without looking at the point-set, or their distances*, and still achieve the best possible approximation factor.

We prove this by writing down the SDP for the best convex combination of k -partitions into X , and analyzing its dual. Consider the following SDP, which, given a feasible fractional k -means solution, tries to find the best convex combination of K_Γ 's which embed into X , and its dual. Here, let Γ denote the set of all possible k -partitions of G .

$$(\text{Primal}) \quad \max \sum_{\Gamma} w_{\Gamma} \text{ st } \sum_{\Gamma} w_{\Gamma} K_{\Gamma} \preceq L_X.$$

$$(\text{Dual}) \quad \min L_X \cdot Y \text{ st } K_{\Gamma} \cdot Y \geq 1, \quad \forall \Gamma \in \Gamma, Y \succeq 0,$$

Clearly, notice that if we show that the primal solution has objective value at least α , then we're done (by scaling by $1/\sum_{\Gamma} w_{\Gamma}$, we'll get a convex combination which embeds into $(1/\alpha)L_X$). In what follows, we'll first show that, indeed, this is true, and subsequently show how to approximately solve this SDP using the Multiplicative Weights framework of Arora and Kale [5].

To show that the objective value is at least α , consider the dual. We show that the dual optimal has value at least α . Indeed, what is the dual trying to solve? Upon careful inspection, it is trying to find, given X , the *worst-case* set of points in Euclidean space, for which the k -means LP has the largest integrality gap. Indeed, suppose the set of points $\{y_1, y_2, \dots, y_n\}$ are such that their gram matrix is Y , then $K_{\Gamma} \cdot Y = \text{Tr}(K_{\Gamma} Y)$ is precisely the k -means cost of the data set according to clustering Γ . So the dual asks for all the true clusterings to have cost at least 1 while minimizing the fractional cost of the k -means LP. But this is precisely the integrality gap instance! From existing rounding algorithms (most relevant to our work is that of Jain and Vazirani [21]), we know that the integrality gap is bounded by a small constant $c_{JV} \leq 216$, and hence the dual objective is at least a constant $1/c_{JV} \geq \alpha$. We hence know that the dual SDP has optimal value at least $\alpha = \frac{1}{216}$. Hence, the primal objective has value at least α , which completes the existential result. It remains to show that we can efficiently construct the distribution $\mathcal{X}$. Indeed, we show that we can achieve this using the Matrix Multiplicative Weights framework of Arora and Kale [5]. Perhaps not surprisingly, the "oracle" needed in their algorithm amounts to running the Jain-Vazirani approximation algorithm for k -means. We provide complete details in Appendix A.

5. AN APPLICATION: PARTITIONING OF WELL SEPARATED GRAPHS

In this section, we demonstrate the power of our objective by presenting an application in a traditional k -way clustering problem. In order to simplify the exposition, we assume

³For the curious reader, $1/\alpha$ is the approximation ratio (more specifically, integrality gap) of the best known rounding algorithm for the Euclidean k -means problem using the natural LP. If running time is not a concern, then $1/\alpha$ is the true integrality gap of the natural k -means LP.

G is regular, with normalized adjacency and Laplacian matrices given by A and $L = I - A$, respectively. Indeed, suppose a graph G has a good k -partition into expanders, such that each cluster has low external sparsity and every cluster is internally an expander. Then we show that, as long as the internal expansion is sufficiently more than the external sparsity⁴, we can recover a clustering which is ϵ -close in symmetric difference, by simply running our spectral clustering algorithm from Section 3. Before we delve into the details, let us introduce some notation.

DEFINITION 5.1. *Given a graph G and a k -partition Γ , we say that Γ is a (k, ϕ, λ) partition for G provided the following: (i) Every $S \in \Gamma$ has small sparsity in G , i.e., $\phi_G(S) \leq \phi$. (ii) Every $S \in \Gamma$ induces an algebraic expander, i.e., $\lambda_2(L[S]) \geq \lambda$.*

A similar notion appeared in the work of Gharan and Trevisan [17]. Unfortunately a direct comparison of both algorithms is not possible, as the goals are different: They first proved the existence of such a (k, λ, ϕ) clustering when there is a (sufficiently large) gap between λ_k and λ_{k+1} , and their constructive algorithm to efficiently find such a clustering requires an even larger gap between λ_k and λ_{k+1} . However, our algorithm works simply by assuming the existence of such a (k, λ, ϕ) partition and finding one which is close to this ground truth.

THEOREM 5.2. *There exists a constant $1 > \alpha > 0$ such that the following holds. Given graph G with (k, ϕ, λ) -partition $\Gamma_{\text{opt}} = \{T_1, \dots, T_k\}$ and $\epsilon \stackrel{\text{def}}{=} \frac{k\phi}{\lambda} \leq \alpha$, our algorithm will output a k -partition Γ of the form $\Gamma = \{S_1, \dots, S_k\}$ such that: $\forall i \in [k] : |S_i \Delta T_i| \leq O(\epsilon) \min(|S_i|, |T_i|)$.*

We devote the remainder of this section to the proof of Theorem 5.2. We use $\Gamma_{\text{opt}} = \{T_1, \dots, T_k\}$ to denote a (k, ϕ, λ) -partition of G with $\frac{k\phi}{\lambda} = \epsilon$ for some $\epsilon \leq O(1)$. We will use $\Gamma = \{S_1, \dots, S_k\}$ to refer to the partition found by our algorithm. Unless noted otherwise, we use S (solution we found) to refer to the clusters in Γ and T (ground truth) to refer to the clusters in Γ_{opt} .

For further convenience, we define $A_{\Gamma_{\text{opt}}}$ and A_{Γ} as the normalized adjacency matrices for the union of cliques on Γ_{opt} and Γ , respectively:

$$A_{\Gamma_{\text{opt}}} \stackrel{\text{def}}{=} \sum_T \frac{1}{|T|} J_T \text{ and } A_{\Gamma} \stackrel{\text{def}}{=} \sum_S \frac{1}{|S|} J_S.$$

Note that $K_{\Gamma_{\text{opt}}} = I - A_{\Gamma_{\text{opt}}}$ and $K_{\Gamma} = I - A_{\Gamma}$.

We begin with some useful preliminary definitions pertaining to how we measure the proximity of a partitioning to the ground truth (which is stronger than the commonly used notion of total number of mis-clustered points which can heavily be biased towards larger clusters and ignoring small clusters altogether), and a theorem connecting nearby partitionings and the corresponding closeness of their indicator matrices.

DEFINITION 5.3. *Given two k -partitions $\Gamma_1, \Gamma_2 \in \Gamma_k(n)$; we say Γ_1 and Γ_2 are ϵ -close if there is a perfect matching $M \subseteq \Gamma_1 \times \Gamma_2$ such that any matched pair $(S, T) \in M$ has $|S \Delta T| \leq \epsilon \min(|S|, |T|)$.*

⁴To the best of our knowledge, all known constructive algorithms need some such assumption which may appear in different manifestations, like an eigenvalue gap, for instance.

THEOREM 5.4. Given two k -partitions $\Gamma_1, \Gamma_2 \in \Gamma_k(n)$, consider the matrices $\mathbf{\Gamma}_1, \mathbf{\Gamma}_2 \in \mathbb{R}^{n \times k}$ where $\mathbf{\Gamma}_1$ is the orthonormal basis corresponding to Γ_1 ($\mathbf{\Gamma}_2$ is defined similarly):

$$\mathbf{\Gamma}_1 = \left[\frac{1}{\sqrt{|S|}} \mathbf{e}_S \mid S \in \Gamma_1 \right],$$

Suppose there exists $\varepsilon < 1/2$ with $\sigma_{\min}(\mathbf{\Gamma}_1^T \mathbf{\Gamma}_2) \geq \sqrt{1-\varepsilon}$. Then Γ_1 and Γ_2 are 2ε -close.

We give the proof in Appendix B.

Next, in Theorem 5.5, we will show that L spectrally dominates the union of cliques on Γ_{opt} and Γ , the solution we find after running spectral clustering.

PROPOSITION 5.5. If each $T \in \Gamma_{\text{opt}}$ induces a λ -algebraic expander, then $L \succeq \lambda K_{\Gamma_{\text{opt}}}$ and $L \succeq \frac{\beta\lambda}{k} K_{\Gamma}$ for some universal constant $\beta > 0$.

PROOF. For any pair of subsets A, B , define $L[A, B]$ as the Laplacian matrix induced between A and B including only the edges between A and B . Observe $L[A, B] \succeq 0$. By abusing notation, we can express L as $L = \sum_{i \leq j} L[T_i, T_j] \succeq \sum_i L[T_i]$. Since $L[T]$ is an algebraic expander for every $T \in \Gamma_{\text{opt}}$, $L[T] \succeq \lambda K_T$. Therefore $L \succeq \lambda \sum_T K_T = \lambda K_{\Gamma_{\text{opt}}}$. Second part follows from Theorem 3.3. $\square$

Next, we will prove that $A_{\Gamma} \approx A_{\Gamma_{\text{opt}}}$ so as to relate Γ and Γ_{opt} . To this end, we upper bound the spectral radius of the Laplacian obtained by contracting each cluster in Γ_{opt} .

LEMMA 5.6. Let

$$\phi \stackrel{\text{def}}{=} \max_{T \in \Gamma_{\text{opt}}} \phi_G(T)$$

. Then $\phi \leq \lambda_{\max}(A_{\Gamma_{\text{opt}}} L A_{\Gamma_{\text{opt}}}) \leq 2\phi$.

PROOF. Define $D \in \mathbb{S}_+^k$ as the diagonal matrix with entries $(|T| \mid T \in \Gamma_{\text{opt}})$ and $P \in \{0, 1\}^{n \times k}$ as the matrix whose columns are indicator vectors for each $T \in \Gamma_{\text{opt}}$. Note that $U \stackrel{\text{def}}{=} P D^{-1/2} \in \mathbb{R}^{n \times k}$ is an orthonormal basis, $U^T U = I_k$. Moreover $U U^T = \Pi$. The entry of matrix $\hat{L} \stackrel{\text{def}}{=} P^T L P$ at (S, T) for $S, T \in \Gamma_{\text{opt}}$ is

$$\hat{L}_{S,T} = \begin{cases} \text{Weight } C(S, \bar{S}) \text{ of edges crossing } S & \text{if } S = T, \\ -\text{Weight of edges between } S \text{ and } T & \text{else.} \end{cases}$$

It is easy to see that $\hat{L}$ is a Laplacian matrix and $\hat{L} \preceq 2 \text{diag}(C(T, \bar{T}))_{T \in \Gamma_{\text{opt}}}$. If we multiply with $D^{-1/2}$ on both sides, on RHS we obtain a diagonal matrix with entries $\frac{2C(T, \bar{T})}{|T|} \leq 2\phi_G(T) \leq 2\phi$:

$$U^T L U = D^{-1/2} \hat{L} D^{-1/2} \preceq 2\phi I_k.$$

Again, multiplying with U, U^T on the left and right, respectively:

$$2\phi \Pi = 2\phi U U^T \succeq U U^T L U U^T = A_{\Gamma_{\text{opt}}} L A_{\Gamma_{\text{opt}}}.$$

For the lower bound, consider $q \leftarrow K \mathbf{e}_T$ where

$$T = \operatorname{argmax}_{T \in \Gamma_{\text{opt}}} \phi_G(T)$$

$$\Pi q = q \implies q^T A_{\Gamma_{\text{opt}}} L A_{\Gamma_{\text{opt}}} q = q^T L q$$

$$= \phi_G(T) \frac{|T| \cdot |\bar{T}|}{n} = \phi_G(T) \|q\|^2.$$

$\square$

We are ready to relate $A_{\Gamma_{\text{opt}}}$ and A_{Γ} to each other via Theorem 5.6. Recall $L \succeq \frac{\lambda}{O(k)} K_{\Gamma}$. Multiply with $A_{\Gamma_{\text{opt}}}$:

$$\frac{\lambda}{O(k)} A_{\Gamma_{\text{opt}}} K_{\Gamma} A_{\Gamma_{\text{opt}}} \preceq A_{\Gamma_{\text{opt}}} L A_{\Gamma_{\text{opt}}} \preceq 2\phi A_{\Gamma_{\text{opt}}}$$

where we used Theorem 5.6 in the last step. In particular, for $\mathbf{\Gamma}$ and $\mathbf{\Gamma}_{\text{OPT}}$ being the orthonormal matrices corresponding to Γ and Γ_{OPT} as described in ??, then we see that $I_k - \mathbf{\Gamma}^T \mathbf{\Gamma} = K_{\Gamma}$, $\mathbf{\Gamma}_{\text{OPT}}^T \mathbf{\Gamma}_{\text{OPT}} = A_{\Gamma_{\text{opt}}}$:

$$A_{\Gamma_{\text{opt}}} K_{\Gamma} A_{\Gamma_{\text{opt}}} \preceq O(\varepsilon) A_{\Gamma_{\text{opt}}} \implies$$

$$(1 - O(\varepsilon)) I_k \preceq \mathbf{\Gamma}_{\text{OPT}}^T \mathbf{\Gamma} \mathbf{\Gamma}^T \mathbf{\Gamma}_{\text{OPT}}.$$

Theorem 5.4 immediately implies that Γ and Γ_{opt} are $O(\varepsilon)$ close.

6. CONNECTION WITH SPECTRAL CLUSTERING

Finally, we present yet another connection with a traditional spectral clustering approach often deployed in practice. In this approach, given a graph, the algorithm for graph partitioning is to essentially compute the top k eigenvectors of the graph Laplacian, project every vertex to the top k eigenvectors, and simply run a k -means clustering algorithm on these vectors. We now show that, implicitly, there is a connection to our objective function. Indeed, we show that if the spectral clustering has low cost for some k , then the same clustering is a good solution for our PSD-embedding of k -cliques problem as well!

Given graph G , let (λ_i, q_i) be the pair of i^{th} smallest eigenvalue and corresponding eigenvector. Consider

$$Q_i \stackrel{\text{def}}{=} \sum_{2 \leq j \leq i} q_i q_i^T.$$

Observe that $Q_n = K$. Recall the basic spectral clustering heuristic for k -clusters: Output the partition found by running k -means on Q_k . In the next theorem, we will show that if this heuristic finds a small cost solution, then the solution is spectrally embeddable into G .

THEOREM 6.1. Suppose there exists a k -partition Γ such that: $\text{Tr}(K_{\Gamma} \cdot Q_k) \leq \kappa \frac{\lambda_2}{\lambda_k - \lambda_2}$. Then $L \succeq \frac{\lambda_k}{\kappa+1} K_{\Gamma}$. Moreover this is within factor $\kappa + 1$ of the best possible.

The proof is in Appendix C.

7. CONCLUSIONS

In this paper we propose a new notion of graph partitioning which involves spectrally embedding a disjoint union of k (normalized) cliques into the original graph. We motivate and justify the study of our notion of spectral embedding by exhibiting several interesting connections to k -means, spectral clustering and the use of resistive embeddings which is a common heuristic in practical applications. Along the way, we give a formal connection bringing forth the implicit objective function being optimized by the clustering heuristic involving the resistive embedding for graph partitioning. As

an application of our framework we also show how to recover good partitions on graphs where one exists.

When studying graph partitioning problems, the modeling aspect of choosing a good objective function is often overlooked in favor of the study of more traditional objectives. Our work illustrates that a well chosen objective function can lead to insights into the operation of heuristic algorithms as well as bring up intriguing algorithmic questions. One such question concerns the use of a novel smoothing step that we use in our algorithm. We empirically demonstrate that smoothing helps in practice in conjunction with current spectral clustering heuristics. It would be interesting to make a rigorous claim about the benefits of smoothing.

8. ACKNOWLEDGMENTS

Moses Charikar is supported by National Science Foundation grants CCF-1565581, CCF-1302518 and a Simons Investigator Award.

9. REFERENCES

- [1] N. Alon. Eigenvalues and expanders. *Combinatorica*, 6(2):83–96, 1986.
- [2] N. Alon, A. Lubotzky, and A. Wigderson. Semi-direct product in groups and zig-zag product in graphs: connections and applications. In *Foundations of Computer Science, 2001. Proceedings. 42nd IEEE Symposium on*, pages 630–637. IEEE, 2001.
- [3] N. Alon and V. D. Milman. λ_1 , isoperimetric inequalities for graphs, and superconcentrators. *Journal of Combinatorial Theory, Series B*, 38(1):73–88, 1985.
- [4] S. Arora and S. Kale. A combinatorial, primal-dual approach to semidefinite programs. In *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, pages 227–236. ACM, 2007.
- [5] S. Arora and S. Kale. A combinatorial, primal-dual approach to semidefinite programs. In *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, pages 227–236. ACM, 2007.
- [6] S. Arora, J. Lee, and A. Naor. Euclidean distortion and the sparsest cut. *Journal of the American Mathematical Society*, 21(1):1–21, 2008.
- [7] S. Arora, S. Rao, and U. Vazirani. Expander flows, geometric embeddings and graph partitioning. *Journal of the ACM (JACM)*, 56(2):5, 2009.
- [8] G. Arzhantseva and R. Tessera. Relatively expanding box spaces with no expansion. *arXiv preprint arXiv:1402.1481*, 2014.
- [9] J. Batson, D. A. Spielman, and N. Srivastava. Twice-ramanujan sparsifiers. *SIAM Journal on Computing*, 41(6):1704–1721, 2012.
- [10] P. Biswal, J. R. Lee, and S. Rao. Eigenvalue bounds, spectral partitioning, and metrical deformations via flows. *Journal of the ACM (JACM)*, 57(3):13, 2010.
- [11] E. G. Boman and B. Hendrickson. Support theory for preconditioning. *SIAM Journal on Matrix Analysis and Applications*, 25(3):694–717, 2003.
- [12] J. Cheeger. A lower bound for the smallest eigenvalue of the laplacian. *Problems in analysis*, 625:195–199, 1970.
- [13] E. Dahlhaus, D. S. Johnson, C. H. Papadimitriou, P. D. Seymour, and M. Yannakakis. The complexity of multiterminal cuts. *SIAM Journal on Computing*, 23(4):864–894, 1994.
- [14] I. Dhillon, Y. Guan, and B. Kulis. *A unified view of kernel k-means, spectral clustering and graph cuts*. Citeseer, 2004.
- [15] I. S. Dhillon, Y. Guan, and B. Kulis. Kernel k-means: spectral clustering and normalized cuts. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 551–556. ACM, 2004.
- [16] N. Garg, V. V. Vazirani, and M. Yannakakis. Approximate max-flow min-(multi) cut theorems and their applications. *SIAM Journal on Computing*, 25(2):235–251, 1996.
- [17] S. O. Gharan and L. Trevisan. Partitioning into expanders. In *SODA*, pages 1256–1266. SIAM, 2014.
- [18] O. Goldschmidt and D. S. Hochbaum. Polynomial algorithm for the k-cut problem. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 444–451. IEEE, 1988.
- [19] N. J. Harvey and N. Olver. Pipage rounding, pessimistic estimators and matrix concentration. In *SODA*, pages 926–945. SIAM, 2014.
- [20] J. Hopcroft and R. Kannan. *Foundations of Data Science*. 2014.
- [21] K. Jain and V. V. Vazirani. Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and lagrangian relaxation. *Journal of the ACM (JACM)*, 48(2):274–296, 2001.
- [22] R. Kannan, S. Vempala, and A. Vetta. On clusterings: Good, bad and spectral. *Journal of the ACM (JACM)*, 51(3):497–515, 2004.
- [23] J. A. Kelner. Spectral partitioning, eigenvalue bounds, and circle packings for graphs of bounded genus. *SIAM Journal on Computing*, 35(4):882–902, 2006.
- [24] N. L. D. Khoa and S. Chawla. Large scale spectral clustering using resistance distance and spielman-teng solvers. In *Discovery Science*, pages 7–21. Springer, 2012.
- [25] N. L. D. Khoa and S. Chawla. A scalable approach to spectral clustering with sdd solvers. *Journal of Intelligent Information Systems*, pages 1–20, 2013.
- [26] A. Kumar and R. Kannan. Clustering with spectral norm and the k-means algorithm. In *Foundations of Computer Science (FOCS), 2010 51st Annual IEEE Symposium on*, pages 299–308. IEEE, 2010.
- [27] J. R. Lee, S. Oveis Gharan, and L. Trevisan. Multi-way spectral partitioning and higher-order cheeger inequalities. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 1117–1130. ACM, 2012.
- [28] T. Leighton and S. Rao. Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms. *Journal of the ACM (JACM)*, 46(6):787–832, 1999.
- [29] M. Lichman. UCI machine learning repository, 2013.
- [30] A. Louis, P. Raghavendra, P. Tetali, and S. Vempala. Many sparse cuts via higher eigenvalues. In *Proceedings of the forty-fourth annual ACM*

symposium on Theory of computing, pages 1131–1140. ACM, 2012.

- [31] F. McSherry. Spectral partitioning of random graphs. In *Foundations of Computer Science, 2001. Proceedings. 42nd IEEE Symposium on*, pages 529–537. IEEE, 2001.
- [32] M. Meila and J. Shi. A random walks view of spectral segmentation. 2001.
- [33] A. Y. Ng, M. I. Jordan, Y. Weiss, et al. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 2:849–856, 2002.
- [34] H. Qiu and E. R. Hancock. Clustering and embedding using commute times. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(11):1873–1890, 2007.
- [35] J. Shi and J. Malik. Normalized cuts and image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(8):888–905, 2000.
- [36] D. A. Spielman and N. Srivastava. Graph sparsification by effective resistances. *SIAM Journal on Computing*, 40(6):1913–1926, 2011.
- [37] D. A. Spielman and S.-H. Teng. Spectral partitioning works: Planar graphs and finite element meshes. *Linear Algebra and its Applications*, 421(2):284–305, 2007.
- [38] D. A. Spielman and S.-H. Teng. Spectral sparsification of graphs. *SIAM Journal on Computing*, 40(4):981–1025, 2011.
- [39] L. Trevisan. Is cheeger-type approximation possible for nonuniform sparsest cut? *arXiv preprint arXiv:1303.2730*, 2013.
- [40] Y. Weiss. Segmentation using eigenvectors: a unifying view. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pages 975–982. IEEE, 1999.
- [41] E. Xing, A. Ng, M. Jordan, and S. Russell. Distance metric learning, with application to clustering with side-information. In *Advances in Neural Information Processing Systems*, volume 15, 2003.
- [42] E. Xing, E. P. Xing, M. Jordan, and M. I. Jordan. On semidefinite relaxations for normalized k-cut and connections to spectral clustering. 2003.
- [43] D. Yan, L. Huang, and M. I. Jordan. Fast approximate spectral clustering. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 907–916. ACM, 2009.

APPENDIX

A. PROOF OF Theorem 4.1

In order to constructively solve the SDP in Section 4 (note that the SDP has exponentially many variables), we appeal to the Matrix Multiplicative Weights framework due to Arora and Kale [5]. Let c_{JV} denote the integrality gap of the Jain-Vazirani algorithm for k -means clustering. We will solve the primal using Arora-Kale approach to get a feasible solution of value $(1 - \delta)(1/c_{JV})$ for any constant δ . Formally, we appeal black-box to the following result of Arora and Kale [5]:

THEOREM A.1. *Consider the following SDP optimization problem with target solution value val (that is feasible):*

$$\begin{aligned} & \max \mathbf{b} \bullet \mathbf{y} \\ & \sum_j \mathbf{A}_j y_j \preceq \mathbf{C} \\ & \mathbf{y} \geq 0 \end{aligned}$$

Also suppose there is an efficient “oracle” algorithm for solving the following linear system given any positive semi-definite $\mathbf{Z}$: $\{\mathbf{y} : \mathbf{y} \geq 0; \mathbf{b} \bullet \mathbf{y} \geq \text{val}; \sum_{j=1}^m (\mathbf{A}_j \bullet \mathbf{Z}) y_j - \mathbf{C} \bullet \mathbf{Z} \leq 0\}$. Then, for any δ , there is an efficient algorithm which runs in time polynomial in $n, 1/\text{val}, 1/\delta$ and ρ which finds a feasible $\mathbf{y}$ with objective value at least $(1 - \delta)\text{val}$. Here ρ is the width, i.e., $\max_j \|A_j y_j - C\|$ of the system.

In our case, the $\mathbf{A}_j$ ’s correspond to the K_Γ ’s, the vector $\mathbf{y}$ corresponds to the vector $\mathbf{w}$, the vector $\mathbf{b}$ is the all ones vector, the matrix $\mathbf{C}$ corresponds to $L(X)$, and finally we set val to be $1/c_{JV}$. Indeed, due to the nice structure of both the matrix $L(X)$ and the set of matrices K_Γ , it is easy to see that the width ρ is at most 2, which bounds the overall runtime. Moreover, the oracle algorithm is also simple: plugging in our values of $\mathbf{A}, \mathbf{b}, \mathbf{C}$, we get that it amounts to solving the following system: $\{\mathbf{y} : \mathbf{y} \geq 0; \sum_\Gamma y_\Gamma \geq 1/c_{JV}; \sum_\Gamma (K_\Gamma \bullet \mathbf{Z}) y_\Gamma - L(X) \bullet \mathbf{Z} \leq 0\}$. Indeed, as mentioned earlier, if we view the psd matrix $\mathbf{Z}$ as the gram matrix of a set of n points P in Euclidean space, then $L(X) \bullet \mathbf{Z}$ is precisely the fractional k -means cost of solution X on dataset P . And using the Jain-Vazirani algorithm, we can find an integer clustering Γ' such that its cost $K_{\Gamma'} \bullet \mathbf{Z} \leq c_{JV} L(X) \bullet \mathbf{Z}$, and so we can set $y_{\Gamma'} = 1/c_{JV}$ and satisfy the system of equations we are checking! This completes the proof of our multiplicative-weights based algorithm. For an informal explanation of how the Arora-Kale algorithm works, please read on.

Explanation of Matrix Multiplicative Weights We will now sketch the informal description of how the Matrix Multiplicative Weights algorithm of Arora and Kale [5] works. Readers familiar with the framework can entirely skip this section, as it only provides a rough overview of the steps of the algorithm. The basis of the algorithm is the following identity: $A \succeq B$ if and only if $A \bullet C \geq B \bullet C$ for all PSD-matrices C . So, the Arora-Kale algorithm intuitively views each psd matrix C as an expert, and maintains a distribution $\mathcal{D}_t$ over experts at each time step t (all of this is succinctly implemented in the final algorithm). Initially, $\mathcal{D}_t$ has all its mass on I , and it updates this over time with the goal of in fact trying to show *infeasibility* of the primal SDP! Indeed, suppose it finds a distribution $\mathcal{D}_t$ ⁵ (with expectation $M_t = E_{M \sim \mathcal{D}_t}[M]$) such that the system $\{\mathbf{w} : \sum_\Gamma w_\Gamma \geq 1/c_{JV}, \sum_\Gamma w_\Gamma K_\Gamma \bullet M_t \leq L(X) \bullet M_t\}$ is infeasible, then we would have found our proof of infeasibility. On the other hand, suppose the above system is indeed feasible, and suppose we find a vector $\mathbf{w}_t$ which satisfies these constraints, then the Arora-Kale algorithm updates $\mathcal{D}_t$ to $\mathcal{D}_{t+1}$ by looking at the *reward matrix* $\sum_\Gamma w_{t,\Gamma} K_\Gamma - L(X)$. Indeed, if an expert C is such that $(\sum_\Gamma w_{t,\Gamma} K_\Gamma - L(X)) \bullet C$ is very positive, then we increase its weight a lot, and if its very negative, we decrease its weight.⁶

⁵Of course, the algorithm does all of this implicitly.

⁶Recall that we are still trying to establish a proof of infeasibility, and when we will have failed, we’d have found a good feasible solution!

Then, after T rounds, if we haven't found a proof of infeasibility, then the experts algorithm guarantees that our overall expected reward is almost at least the reward of the best expert in hindsight. Our overall reward is simply,

$$0 \geq \frac{1}{T} \sum_{t=1}^T \left(\sum_{\Gamma} w_{t,\Gamma} K_{\Gamma} - L(X) \right) \bullet M_t$$

Here, this sum is at most 0 because we have assumed that always found a feasible $\mathbf{w}_t$ for all steps of our algorithm. And the reward of any expert C is

$$\frac{1}{T} \sum_{t=1}^T \left(\sum_{\Gamma} w_{t,\Gamma} K_{\Gamma} - L(X) \right) \bullet C$$

The experts algorithm guarantees that, for all experts C , our reward is at least its reward, and in particular,

$$0 \geq \frac{1}{T} \sum_{t=1}^T \left(\sum_{\Gamma} w_{t,\Gamma} K_{\Gamma} - L(X) \right) \bullet C - \delta$$

for some small constant δ . But this says that the solution $\hat{\mathbf{w}} = \frac{1}{T} \sum_{t=1}^T \mathbf{w}_t$ is almost a feasible dual solution. It's dual objective value is at least $1/c_{JV}$ (as the individual $\mathbf{w}_t$'s satisfied this), and moreover, for all psd C , it satisfies

$$\sum_{\Gamma} \hat{w}_{\Gamma} K_{\Gamma} \bullet C \leq L(X) \bullet C + \delta$$

Intuitively, this almost means that $\sum_{\Gamma} \hat{w}_{\Gamma} K_{\Gamma} \preceq L(X) + \delta I$. They also show that T only depends polynomially on n and $1/\delta$. And we can make it strictly feasible by scaling the $\hat{\mathbf{w}}$ by a bit while only losing out on a little in the objective function. Throughout this above analysis, we have assumed that we will always find a feasible $\mathbf{w}_t$ for all of the T steps. Why is that? Indeed, here is where we use the rounding algorithm due to Jain and Vazirani [21]. Let us revisit what the linear system corresponds to. Given a distribution $\mathcal{D}_t$ with expectation $M_t = E_{M \sim \mathcal{D}_t}[M]$ (which is positive semi-definite), it is $\{\mathbf{w} : \sum_{\Gamma} w_{\Gamma} \geq 1/c_{JV}, \sum_{\Gamma} w_{\Gamma} K_{\Gamma} \bullet M_t \leq L(X) \bullet M_t\}$. But if we view M_t as the gram matrix of a set of n points P_t in Euclidean space, then $L(X) \bullet M_t$ is precisely the fractional k -means cost of solution X on dataset P_t . And using the Jain-Vazirani algorithm, we can find an integer clustering Γ_t such that its cost $K_{\Gamma_t} \bullet M_t \leq c_{JV} L(X) \bullet M_t$, and so we can set $w_{\Gamma_t} = 1/c_{JV}$ and satisfy the system of equations we are checking! Essentially, this completes the high level overview of the Arora-Kale algorithm.

B. PROOF OF THEOREM 5.4

We define $\pi_1 : \Gamma_1 \rightarrow \Gamma_2$ and $\pi_2 : \Gamma_2 \rightarrow \Gamma_1$ as the following:

$$\forall S \in \Gamma_1 : \pi_1(S) \stackrel{\text{def}}{=} \operatorname{argmax}_{T \in \Gamma_2} \frac{|S \cap T|}{|T|} \quad \text{and}$$

$$\forall T \in \Gamma_2 : \pi_2(T) \stackrel{\text{def}}{=} \operatorname{argmax}_{S \in \Gamma_1} \frac{|S \cap T|}{|S|}.$$

Consider $M = \{(S, \pi_1(S)) \mid S \in \Gamma_1\}$: By B.2 and B.3, M is indeed a perfect matching between Γ_1 and Γ_2 . Now consider any matched pair $(S, T) \in M$. Without loss of generality, say $|S| \geq |T|$. By B.1, $|S \cap T| \geq (1 - \varepsilon)|S|$. Since $|S \Delta T| = |S| + |T| - 2|S \cap T|$:

$$|S \Delta T| \leq |S| + |T| - 2(1 - \varepsilon)|S| = 2\varepsilon|S| + (|T| - |S|) \leq 2\varepsilon|S|.$$

We finish our proof with proving Theorems B.1, B.2, and B.3 below.

CLAIM B.1. *If $\pi_1(S) = T$, then $|S \cap T| \geq (1 - \varepsilon)|T|$. Similarly, if $\pi_2(T) = S$, then $|S \cap T| \geq (1 - \varepsilon)|S|$.*

PROOF. Consider the matrix $P = \mathbf{\Gamma}_1^T \mathbf{\Gamma}_2 \mathbf{\Gamma}_2^T \mathbf{\Gamma}_1 \in \mathbb{S}_+^k$ so that $\lambda_{\min}(P) = \sigma_{\min}^2(\mathbf{\Gamma}_1^T \mathbf{\Gamma}_2)$. It is easy to notice that $\lambda_{\min}(P) = \sigma_{\min}^2(\mathbf{\Gamma}_1^T \mathbf{\Gamma}_2) \geq 1 - \varepsilon$. In particular, all diagonals of P are at least $1 - \varepsilon$. Consider any diagonal corresponding to $S \in \Gamma_1$:

$$1 - \varepsilon \leq \frac{\mathbf{e}_S^T \mathbf{\Gamma}_2 \mathbf{\Gamma}_2^T \mathbf{e}_S}{|S|} = \sum_{T \in \Gamma_2} \frac{|S \cap T|^2}{|S||T|}$$

$$\leq \left(\max_{T' \in \Gamma_2} \frac{|S \cap T'|}{|T'|} \right) \sum_{T \in \Gamma_2} \frac{|S \cap T|}{|S|} = \max_{T' \in \Gamma_2} \frac{|S \cap T'|}{|T'|},$$

which, by construction, is equal to $\frac{|S \cap \pi_1(S)|}{|\pi_1(S)|}$. This proves the first part of the claim. The second part follows immediately by applying the same argument on Γ_2 and Γ_1 . $\square$

CLAIM B.2. *Both π_1 and π_2 are bijections.*

PROOF. Suppose $\pi_1(S) = \pi_1(S') = T$ for some $S \neq S'$. Since S, S' are disjoint and $\varepsilon < \frac{1}{2}$:

$$|T| \geq |S \cap T| + |S' \cap T| \geq 2(1 - \varepsilon)|T| > |T|,$$

a contradiction. A similar argument shows that π_2 is a bijection as well. $\square$

This completes the proof of the theorem.

CLAIM B.3. $\pi_1 = \pi_2^{-1}$.

PROOF. Suppose not. Since both Γ_1 and Γ_2 are bijections by B.2, there exists a cycle of the form

$$(S_0, T_0, \dots, S_{m-1}, T_{m-1}, S_m = S_0)$$

where $\pi_1(S_i) = T_i$ and $\pi_2(T_i) = S_{i+1}$ for some $m \geq 2$.

By construction, $|S_i \cap T_i| \geq (1 - \varepsilon)|T_i|$ which means $\varepsilon|T_i| \geq |T_i \setminus S_i|$. Since S_i and S_{i+1} are disjoint, $|T_i \setminus S_i| \geq |T_i \cap S_{i+1}|$. Again, by construction, $|T_i \cap S_{i+1}| \geq (1 - \varepsilon)|S_{i+1}|$. Therefore:

$$\varepsilon|T_i| \geq (1 - \varepsilon)|S_{i+1}| \implies |T_i| \geq \frac{1 - \varepsilon}{\varepsilon}|S_{i+1}| > |S_{i+1}|$$

The last inequality holds because $\varepsilon < 1/2$. By a similar argument, we can also show that $|S_i| > |T_i|$. Consequently, $|S_0| > |S_1| > \dots > |S_m| = |S_0|$ which is a contradiction. So all cycles have length 2, which implies $\pi_1 = \pi_2^{-1}$. $\square$

C. PROOF OF THEOREM 6.1

For some r (we will fix r to k later) Γ satisfies:

$$\operatorname{Tr}(K_{\Gamma} \cdot Q_r) \leq \kappa \frac{\lambda_2}{\lambda_r - \lambda_2}.$$

It is easy to see that $L^{\dagger} \preceq \frac{1}{\lambda_r} \left(\frac{\lambda_r - \lambda_2}{\lambda_2} Q_r + K \right)$. We want to upper bound $\lambda_{\max}(K_{\Gamma} L^{\dagger} K_{\Gamma})$ which is at most

$$\frac{1}{\lambda_r} \left[\frac{\lambda_r - \lambda_2}{\lambda_2} \operatorname{Tr}(K_{\Gamma} Q_r) + 1 \right] \leq \frac{1}{\lambda_r} (\kappa + 1).$$

By Theorem 3.1, $L \succeq \frac{\lambda_r}{\kappa + 1} K_{\Gamma}$. Substituting $r = k$ yields the first claim. Second one follows from $\lambda_{\min}(L, K_{\Gamma'}) \leq \lambda_k$ for any k -partitioning Γ' .