Thèse n° 8132

Sublinear Algorithms for Spectral Graph Clustering

Présentée le 17 décembre 2021

Faculté informatique et communications
Laboratoire de théorie du calcul 4
Programme doctoral en informatique et communications

pour l'obtention du grade de Docteur ès Sciences

par

Aidasadat MOUSAVIFAR

Acceptée sur proposition du jury

Prof. R. Urbanke, président du jury
Prof. M. Kapralov, directeur de thèse
Prof. C. Sohler, rapporteur
Dr V. Cohen-Addad, rapporteur
Prof. E. Abbé, rapporteur

École
polytechnique
fédérale
de Lausanne

2021

Once you decide where it is you want to be,
you won't stop pushing forward until you get there,
that's how winners are made.
—Gary Raser

To my family for their endless love and devotion
To my Ph.D. advisor, Michael Kapralov for his continuous support

Acknowledgements

Ph.D. is a long journey and this would not have been possible without the support of many people who made my Ph.D. successful and enjoyable.

First, I would like to express my deepest appreciation to my advisor, Michael Kapralov, for his continuous support, supervision, and guidance. He has been an ideal teacher and mentor for me throughout my Ph.D. The working environment Michael has created is bringing out the best qualities in his students and maintaining high standards and strong motivation. His guidance was crucial during my whole time at EPFL, whether in my research or my academic career. I endeavour to use the valuable lessons I learned from Michael throughout my life.

I would like to thank the other members of my thesis committee: Prof. Christian Sohler, Prof. Emmanuel Abbé, Dr. Vincent Cohen-Addad, and Prof. Rüdiger Urbanke for reading this work and the insightful discussions we had during the defense.

I am immensely thankful to Dr. Silvio Lattanzi for hosting me at Google Research and for his engaging discussions about my research. I am also grateful to the people who have guided me at various points of my career (at the University of Tehran, EPFL, and Google Research), from whom I have learned a lot. It has been a true honour to write papers with many amazing co-authors: Sanjeev Khanna, Ola Svensson, Christian Sohler, Silvio Lattanzi, Elisa Celis, Christopher Musco, Cameron Musco, Aaron Sidford, Yuval Peres, Slobodan Mitrović, Jakub Tarnawski, Grzegorz Gluch, Jakab Tardos, Ashkan Norouzi-Fard, Navid Nouri and Amir Zandieh.

It has been a great pleasure to be part of the Theory of Computation lab and to share an office with many wonderful and smart people. I thank all my friends in Lausanne for their great company and all the fun times we had in the past five years.

I would also like to express my sincere gratitude to the administrative staff, especially Pauline Raffestin. She was always there when I needed help and when I had any questions.

I would like to thank Google for giving me the Google PhD Fellowship and generously funding my Ph.D. since 2019.

Finally, and most importantly, I want to express my boundless gratitude to my family for their endless love, devotion and support. None of this would have been possible without their enormous encouragement, help and support.

Lausanne, August 13, 2021 A. M.

Abstract

This thesis focuses on designing spectral tools for graph clustering in sublinear time. With the emergence of big data, many traditional polynomial time, and even linear time algorithms have become prohibitively expensive. Processing modern datasets requires a new set of algorithms for computing with extremely constrained resources, i.e., *sublinear algorithms.* Clustering is one of the well-known techniques for solving large-scale optimization problems in a wide variety of domains, including machine learning, data science, and graph analysis [ABM16, RAK$^+$16, GLMY11]. Efficient sublinear solutions for fundamental graph clustering problems require going well beyond classic techniques.

In this thesis, we present an *optimal* sublinear-time algorithm for *testing k-clusterability problem*, i.e., quickly determining whether the graph can be partitioned into at most k expanders, or is far from any such graph. This is a generalization of a well-studied problem of testing graph expansion. The classic results on testing k-clusterability either consider the testing expansion problem (i.e, $k = 1$ vs $k \geq 2$) [KS11, NS10], or address the problem for larger values of k under the assumption that the gap between conductances of accepted and rejected graphs is at least logarithmic in the size of the graph [CPS15]. We overcome these barriers by developing novel spectral techniques based on analyzing the spectrum of the Gram matrix of random walk transition probabilities. We complement our algorithm with a matching lower bound on the query complexity of testing k-clusterability, which improves upon the long-standing previous lower bound for testing graph expansion.

Furthermore, we extend our previous result from the *property testing* framework to an efficient clustering algorithm in the *local computation algorithm* (LCA) model. We focus on a popular variant of graph clustering where the input graph can be partitioned into k expanders with outer conductance bounded by ϵ. We construct a small space data structure that allows quickly classifying vertices of G according to the cluster they belong to in sublinear time. Our spectral clustering oracle provides $O(\epsilon \log k)$ error per cluster for any $\epsilon \ll 1/\log k$. Our main contribution is a sublinear time oracle that provides dot product access to the spectral embedding of the graph. We estimate dot products with high precision using an appropriate linear transformation of the Gram matrix of random walk transition probabilities. Finally, using dot product access to the spectral embedding we design a spectral clustering oracle. At a high level, our approach amounts to hyperplane partitioning in the spectral embedding of the graph but crucially operates on a nested sequence of carefully defined subspaces in the spectral embedding to achieve per cluster recovery guarantees.

Keywords: Sublinear Algorithms, Graph Clustering, Spectral Methods, Property Testing Framework, Local Computation Algorithms

Zusammenfassung

Diese Arbeit konzentriert sich auf die Entwicklung von spektralen Werkzeugen für das Clustering von Graphen in sublinearer Zeit. Mit dem Aufkommen von Big Data sind viele traditionelle Algorithmen mit polynomialer Zeit und sogar linearer Zeit unerschwinglich geworden. Die Verarbeitung moderner Datensätze erfordert einen neuen Satz von Algorithmen für das Rechnen mit extrem eingeschränkten Ressourcen, d. h. sublineare Algorithmen. Clustering ist eine der bekannten Techniken zur Lösung großer Optimierungsprobleme in einer Vielzahl von Domänen, einschließlich maschinelles Lernen, Datenwissenschaft und Graphenanalyse [ABM16, RAK$^+$16, GLMY11]. Effiziente sublineare Lösungen für fundamentale Graphen-Clustering-Probleme erfordern es, weit über klassische Techniken hinauszugehen.

In dieser Arbeit stellen wir einen *optimalen* Algorithmus mit sublinearer Zeit für das k-Clusterbarkeitsproblem vor, d. h. die schnelle Bestimmung, ob der Graph in höchstens k Expander partitioniert werden kann oder weit von einem solchen Graphen entfernt ist. Dies ist eine Verallgemeinerung eines gut untersuchten Problems der Prüfung der Graphenexpansion. Die klassischen Ergebnisse zum Testen der k-Clusterbarkeit betrachten entweder das Problem des Testens der Expansion (d.h. $k = 1$ vs. $k \geq 2$) [KS11, NS10], oder behandeln das Problem für größere Werte von k unter der Annahme, dass der Abstand zwischen den Leitwerten von akzeptierten und verworfenen Graphen mindestens logarithmisch in der Größe des Graphen ist [CPS15]. Wir überwinden diese Barrieren durch die Entwicklung neuartiger spektraler Techniken, die auf der Analyse des Spektrums der Gram-Matrix von Übergangswahrscheinlichkeiten des Random Walk. Wir ergänzen unseren Algorithmus mit einer passenden unteren Schranke für die Abfragekomplexität des Testens von k-Clusterbarkeit, die die seit langem bestehende untere Schranke für das Testen von Graphenexpansion verbessert.

Darüber hinaus erweitern wir unser vorheriges Ergebnis aus dem *Eigenschaftstest*-Rahmen auf einen effizienten Clustering-Algorithmus im *lokalen Berechnungsalgorithmus* (LCA) Modell. Wir konzentrieren uns auf eine populäre Variante des Graphenclustering, bei der der Eingabegraph in k Expander partitioniert werden kann, deren äußerer Leitwert durch ϵ begrenzt ist. Wir konstruieren eine kleinräumige Datenstruktur, die es erlaubt, Scheitelpunkte von G schnell und in sublinearer Zeit nach dem Cluster zu klassifizieren, dem sie angehören. Unser spektrales Clustering-Orakel liefert $O(\epsilon \log k)$ Fehler pro Cluster für jedes $\epsilon \ll 1/\log k$. Unser Hauptbeitrag ist ein Orakel mit sublinearer Zeit, das Punktprodukt-Zugriff auf die spektrale Einbettung des Graphen bietet. Wir schätzen Punktprodukte mit hoher Genauigkeit, indem wir eine geeignete lineare Transformation der Gram-Matrix der Übergangswahrscheinlichkeiten des Random Walk verwenden. Schließlich entwerfen wir unter Verwendung des Punktproduktzugriffs auf die spektrale Einbettung ein spektrales Clustering-Orakel. Auf hohem Niveau läuft unser Ansatz auf eine Hyperebenen-Partitionierung in der spektralen Einbettung des Graphen hinaus, operiert aber auf einer verschachtelten Sequenz von sorgfältig definierten Unterräumen in der spektralen Einbettung, um Wiederherstellungsgarantien pro Cluster zu erreichen.

Schlüsselwörter: Sublineare Algorithmen, Graphen-Clustering, Spektrale Methoden, Property Testing Framework, Lokale Berechnungsalgorithmen

Contents

1 Introduction

The growth of modern datasets, coming from diverse sources such as social networks, sensor networks, and video streams, prompt us to reconsider the efficiency of traditional algorithms. In many modern scenarios, the input is so large that it does not fit on a single machine. It is often challenging to partition the data (and process it in parallel) in a communication-efficient manner. As a result, classic polynomial time/space algorithms are prohibitively expensive, and often even linear algorithms are too slow. Hence, more efficient solutions, i.e., *sublinear time/space* algorithms are needed. Sublinear algorithms require resources that are substantially smaller than the size of the input: Such algorithms often return a solution after inspecting only a minuscule fraction of the data (*sublinear-time algorithms*), or scan the input data stream while maintaining a small but accurate summary of the data, and compute the answer based on that (*sketching algorithms*).

Sublinear algorithms for basic statistical estimation problems, such as finding the most frequent items in data streams, are now routinely used in the practice of massive data analysis (e.g., HYPERLOGLOG and COUNTSKETCH [HNH13]). However, modern data analysis needs techniques for answering more complex queries on inputs that often come in the form of large graphs or matrices: For example, given a large graph (e.g., a nearest-neighbor graph), to quickly retrieve an approximate clustering of the graph, or to maintain a small summary of a dynamically updated graph (e.g., the Twitter network) that enables retrieval of *spectral* properties of the graph (e.g., random walks, or personalized PageRank needed for recommendation engines). Efficient sublinear solutions to these fundamental questions require going well beyond classic techniques.

Clustering is one of the well-known problems in discrete optimization and clustering techniques are widely used for solving large-scale optimization problems in many areas, including machine learning, data analysis, social sciences, and statistics. Clustering methods have various applications: for example graph partitioning in Google Maps driving directions [ABM16], finding treatment groups for experimental design [RAK$^+$16], and community detection for topic discovery on YouTube [GLMY11].

Clustering is the task of partitioning data points into several groups, called clusters, such that the data points in the same group are similar to each other and dissimilar to the points in other groups. One way of representing the data points is in the form of a *similarity graph*, where we connect two data points (vertices) by an edge if they are similar. There are several constructions of the similarity graph for modeling the local neighborhood relationships between the data points (e.g., ϵ-neighborhood graph, k-nearest neighbor graph). The goal of graph clustering is to partition the vertices of a graph (e.g., the similarity graph) into disjoint subgraphs based on the well-connectivity of the vertices. One well-studied measure to evaluate the quality of a cluster is *conductance* [KVV04a]. The conductance of a cluster assesses the well-connectivity of the vertices inside a cluster and leads to a natural graph clustering objective, i.e., partitioning the vertices of a graph into clusters with high internal conductance and sparse boundaries.

One of the most popular graph-clustering techniques is *spectral clustering*; it is broadly used in practice due to its simple implementation by linear-algebra methods. Spectral clustering algorithms significantly outperform traditional clustering algorithms such as k-means or single linkage [VL07, Jai10]. These techniques make use of the few bottom eigenvectors of graph Laplacian to define a feature vector for data points, and to perform dimensionality reduction. Spectral clustering has various applications in a wide range of problems, for example, image segmentation [SM00], speech separation [BJ06], and clustering of protein sequences [PCS06]. Conductance is closely related to the *spectral* properties of the graph, behaviour of random walks, and connectivity. The focus of this thesis is to understand the power of spectral algorithms in discovering the cluster structure of graphs in sublinear time.

Following this, we summarize the contributions of this thesis and give a general outline of the remaining chapters.

1.1 Overview of our contributions

In this section, we give an overview of our results and techniques. First, we define the notion of conductance, and then we formulate our main contributions using Definition 1.1.

Definition 1.1 (**Internal and external conductance**)**.** Let $G = (V, E)$ be a graph. Let $\deg(x)$ be the degree of vertex x. For a set $S \subseteq V$, let $\mathrm{vol}(S) = \sum_{x \in S} \deg(x)$ denote the *volume* of set S. For a set $S \subseteq C \subseteq V$, let $E(S, C \setminus S)$ be the set of edges with one endpoint in S and the other in $C \setminus S$. The *conductance of a set S within C* is $\phi_C^G(S) = \frac{|E(S,C\setminus S)|}{\mathrm{vol}(S)}$. The *external-conductance* of set C is defined to be $\phi_V^G(C) = \frac{|E(C,V\setminus C)|}{\mathrm{vol}(C)}$. The *internal-conductance* of set $C \subseteq V$, denoted by $\phi^G(C)$, is $\min_{S\subseteq C, 0<\mathrm{vol}(S)\le \frac{\mathrm{vol}(C)}{2}} \phi_C^G(S)$ if $|C| > 1$ and one otherwise.

Intuitively, the inner conductance of a cluster measures the strength of connections across any cut inside the cluster relative to the strength of connections inside the smaller of the cut. Roughly speaking, vertices of a set with large inner conductance are well-connected, and a set with small outer conductance has few connections to the outside. The conductance measure leads to a natural graph clustering objective, which is to partition the vertices of a graph into clusters with large inner conductance and small outer conductance. Oveis Gharan and Trevisan [GT14a] and Zhu et al. [ZLM13] proposed to combine both and inner conductance and outer conductance for characterizing a cluster. Inspired by this, we consider a popular version of the spectral clustering problem where one assumes the existence of a planted solution, namely that the input graph can be partitioned into a disjoint union of k induced expanders with outer conductance bounded by $\epsilon \ll 1$ (Definition 1.2).

Definition 1.2 (**(k,φ,ϵ)-clustering**)**.** Let $G = (V, E)$ be a graph. A (k,φ,ϵ)-clustering of G is a partition of vertices V into disjoint subsets $C_1,\dots,C_k$ such that for all $i \in [k]$, $\phi^G(C_i) \geq \varphi$, $\phi^G_V(C_i) \leq \epsilon$. Graph G is called (k,φ,ϵ)-clusterable if there exists a (k,φ,ϵ)-clustering for G.

An average-case version of this problem where the clusters induce Erdos-Renyi graphs has been studied in the literature on the stochastic block model (SBM) [Abb18]. We study the worst-case version of the problem and the goal is to recover clusters with small misclassification error. Towards this objective, many graph clustering algorithms have been developed [KVV04b, NJW02, SM00, VL07]. Any algorithm that outputs such a partition necessarily requires $\Omega(n)$ time even to output the solution, where n denote the number of vertices in the input graph. However, on very large-scale graphs, even linear time algorithms might be too slow and consequently, there has been considerable recent interest in learning the cluster structure of graphs in sublinear time.

The most basic question towards understanding the cluster structure is quickly determining the number of clusters in the graph (i.e., k). Formally, we would like to test whether the input graph can be partitioned into at most k clusters with inner conductance at least φ (i.e., (k,φ)-clusterable), or is *far* from any such graph. First, in Chapter 2, we present a sublinear algorithm for testing graph-clusterability in the *property testing* framework. For testing (k,φ)-clusterable graphs, we do not require the outer conductance of clusters to be small. Next, in Chapter 3, we extend the testing result to an efficient *local computation algorithm* that given a (k,φ,ϵ)-clusterable graph, recovers the clusters up to a small error.

In the following subsection, we briefly outline the recent results on *testing graph properties* and present our main contributions on testing graph clusterability.

1.1.1 Testing graph clusterability

Goldreich and Ron [GR02] initiated the framework of testing graph properties. In this framework, the algorithm is given oracle access to the graph and has to decide whether the graph has a certain property (YES case) or is *far* from having that property (NO case). Here the notion of ϵ-far means that one needs to modify at least an ϵ-fraction of edges to convert one graph into another. There has been an extensive line of work on testing various graph properties in the framework of Goldreich and Ron. Several properties known to be testable in this model such as bipartiteness [GR99], degree distribution moments [ERS17b], number of triangles [ELRS17], and number of k-cliques [ERS17a]. Czumaj et al. designed algorithms for testing several properties including cycle-freeness [CGR$^+$14]. Newman et al. develop algorithms to test hyperfinite properties [NS13], whereas Eden et al. designed algorithms to test arboricity [ELR18]. In our study, we use a standard graph exploration model for sublinear algorithms which allows us to perform *neighbor queries* to the graph. Using this model one can perform various graph exploration processes on the graph such as running random walks.

We study the problem of testing k-clusterability where the goal is to distinguish graphs that are (k,φ)-clusterable from graphs that are ϵ-far from admitting such clustering. This is a generalization of a well-studied problem of testing graph expansion, in which the task is to distinguish an expander ($k = 1$), from a graph having a sparse cut ($k \geq 2$). Goldreich and Ron [GR02] showed that $\Omega(\sqrt{n})$ queries are necessary to differentiate between expanders and graphs that are far from expanders. Then [KS11, NS10] developed algorithms to distinguish an expander with inner conductance at least φ from a graph that is far from having expansion $\gamma \cdot \varphi^2$ in time $n^{1/2+O(\gamma)}$. The problem of *testing k-clusterability* for $k > 1$ has recently been considered in the literature and a recent work of [CPS15] gave an ingenious sublinear time algorithm for testing k-clusterability in time $\tilde{O}(n^{1/2} \cdot k^{O(1)})$. Their algorithm implicitly embeds a random sample of vertices into Euclidean space, and then partition the samples into clusters based on Euclidean distances between the points. This yields a very efficient testing algorithm, but only works if the cluster structure is very pronounced: it is necessary to assume that the gap between conductances of graphs in YES case and NO case is at least logarithmic in the size of the graph G. In particular, the algorithm requires that the accepted be (k,φ)-clusterable, while the rejected graphs are ϵ-far from being $(k,\varphi^2/\log n)$-clusterable. Thus the quality of clusters in the NO case has to be weakened by factor $\log n$.

In Chapter 2, we develop an optimal sublinear-time algorithm for this problem that can differntiate (k,φ)-clusterable graphs from graphs that are far from from being $(k,\gamma \cdot \varphi^2)$-clusterable in time $n^{1/2+O(\gamma)}$. Our tester works even when the separation between the conductance of accepted and rejected graphs i.e γ is a sufficiently large constant which is a considerable improvement comparing to [CPS15] that requires $\gamma \ll \frac{1}{\text{poly}(k,\log n)}$. Our algorithm works based on the singular value decomposition of a Gram matrix of the random walk transition probabilities from a small sample of seed nodes. Instead of classifying vertices based on pairwise Euclidean distances, our tester decides to accept or reject the graph by estimating the $(k+1)$-th eigenvalue of the Gram matrix which turns out to be a more robust tester.

We complemented our algorithm with a matching lower-bound of $n^{1/2+\Omega(\gamma)}$ on the query complexity of testing k-clusterability, which improves upon the long-standing previous lower-bound of $\Omega(n^{1/2})$ for testing graph expansion (i.e., $k = 1$ vs $k \geq 2$). Our lower bound is based on a novel property testing problem, which we analyze using Fourier analytic tools. As a byproduct of our techniques, we also achieve new lower bounds for the problem of approximating MAX-CUT value in sublinear time.

1.1.2 Sublinear time clustering oracles

In Chapter 3, we extend our previous result from the property testing framework to an efficient clustering algorithm in the Local Computation Algorithms (LCA) model. Local computation algorithms proposed by Ronitt et al. [RTVX11] are used to model sublinear algorithms. In many modern scenarios, the input and the output solution are so large that even writing the entire output is prohibitively expensive. However, at any point in time, only a small portion of the output is required to be queried. In this model, the algorithm should answer all the queries to the output solution consistently. In particular, a local computation algorithm for graph clustering should provide consistent query access with respect to a certain partition of the graph and to determine the cluster membership for the set of queried vertices.

We study a popular version of the problem where the input graph admits a (k,φ,ϵ)-clustering (Definition 1.2) and the goal is to recover every cluster up to an $O(\epsilon)$ misclassification error. We construct a small space data structure that allows quickly classifying vertices of G according to the cluster they belong to in sublinear time. We refer to this data structure as a *spectral clustering oracle.*

Spectral clustering techniques often first compute the spectral embedding of vertices and then partition the vertices according to their embedding. Since computing the singular value decomposition on very large graphs is computationally expensive, this approach seems to be highly non-local. Our main contribution in this work is a sublinear time oracle that provides query access to dot products in the spectral embedding of G. Our dot product oracle has $n^{1/2+O(\epsilon)}$ preprocessing time and query time. Using this data structure we develop a sublinear time spectral clustering oracle with $k^{O(1)} \cdot n^{1/2+O(\epsilon)}$ query time and $2^{O(1/\epsilon \cdot k^4 \cdot \log^2 k)} \cdot n^{1/2+O(\epsilon)}$ preprocessing time that guarantees $O(\epsilon \log k)$ misclassification error per cluster for any $\epsilon \ll \frac{1}{\log k}$. We show that the query time can be reduced by increasing the preprocessing time as long as the product is about $n^{1+O(\epsilon)}$. This, in particular, gives a nearly linear time primitive for spectral clustering.

Our dot product oracle works based on estimating distributions of random walks from few sampled vertices in G. The distributions themselves provide a poor approximation to the spectral embedding, but we use an appropriate linear transformation to achieve high precision dot product access. We analyze this estimator by spectral perturbation bounds and novel tail bounds on the spectral embedding of a k-clusterable graph. Our spectral clustering oracle performs hyperplane partitioning in the spectral embedding of G, and it crucially operates on a nested sequence of carefully defined subspaces in the spectral embedding to achieve per cluster recovery guarantees.

Organization: The remaining chapters of the thesis contain the full results. We present our results in separate self-contained chapters that can be read independently. Each chapter has its own introduction that covers the prior work and a detailed discussion on the new techniques we develop.

2 Testing Graph Clusterability: Algorithms and Lower Bounds

This chapter is based on a joint work with Ashish Chiplunkar, Michael Kapralov, Sanjeev Khanna and Yuval Peres. It has been accepted to the 59th Annual IEEE Symposium on Foundation of Computer (FOCS'18) [CKK$^+$18].

2.1 Introduction

Graph clustering is the problem of partitioning vertices of a graph based on the connectivity structure of the graph. It is a fundamental problem in many application domains where one wishes to identify groups of closely related objects, for instance, communities in a social network. The clustering problem is, thus, to partition a graph into vertex-disjoint subgraphs, namely clusters, such that each cluster contains vertices that are more similar to each other than the rest of the graph. There are many natural measures that have been proposed to assess the quality of a cluster; one particularly well-studied and well-motivated measure for graph clustering is conductance of a cluster [KVV04a]. Roughly speaking, conductance of a graph measures the strength of connections across any partition of vertices relative to the strength of connections inside the smaller of the two parts. The higher the conductance inside a cluster, the harder it is to split it into non-trivial pieces. The conductance measure lends itself to a natural graph clustering objective, namely, partition the vertices of a graph into a small number of clusters such that each cluster has large conductance in the graph induced by it (the inner conductance of the cluster). Towards this objective, many efficient graph partitioning algorithms have been developed that partition vertices of a graph into a specified number of clusters with approximately high conductance (when possible). Any algorithm that outputs such a partition necessarily requires $\Omega(n)$ time – simply to output the solution, and usually $\Omega(m)$ time, where n and m respectively denote the number of vertices and edges in the input graph. On very large-scale graphs, even linear-time algorithms may prove to be computationally prohibitive, and consequently, there has been considerable recent interest in understanding the cluster structure of a graph in sublinear time. Specifically, given a target number of clusters, say k, and a measure ϕ of desired cluster quality, how much exploration of the input graph is needed to distinguish between graphs that can be partitioned into at most k clusters with inner conductance at least ϕ from graphs that are far from admitting such clustering? The focus of this paper is to understand the power of sublinear algorithms in discovering the cluster structure of a graph.

In our study, we use by now a standard model of graph exploration for sublinear algorithms, where at any step, the algorithm can either sample a uniformly at random vertex, query the degree $d(u)$ of a vertex u, or specify a pair (u,i) and recover the i^{th} neighbor of u for any $i \in [1..d(u)]$. For any positive $\epsilon > 0$, we say a pair of graphs is ϵ-far if one needs to modify at least an ϵ-fraction of edges to convert one graph into another.

The simplest form of the cluster structure problem is the case $k = 1$: how many queries to the graph are needed to distinguish between graphs that are expanders (YES case) from graphs that are $\Omega(1)$-far from being expanders (NO case)? A formal study of this basic question was initiated in the work of Goldreich and Ron [GR02] where they showed that even on bounded degree graphs, $\Omega(\sqrt{n})$ queries to the input graph are necessary to distinguish between expanders and graphs that are far from expanders. On the positive side, it is known that a bounded degree expander graph with conductance at least ϕ can be distinguished from a graph that is $\Omega(1)$-far from a graph with conductance $\gamma \times \phi^2$ for some positive constant γ, using only $n^{\frac{1}{2}+O(\gamma)}$ queries [KS11, NS10]. Thus, even the simplest setting of the graph clustering problem is not completely understood – the known algorithmic results require additional separation in the conductance requirements of YES and NO instances. Furthermore, even with this separation in conductance requirements, the best algorithmic result requires polynomially more queries than suggested by the lower bound.

Lifting algorithmic results above for the case $k = 1$ to larger values of k turned out to be a challenging task. A breakthrough was made by Czumaj, Peng, and Sohler [CPS15] who designed an algorithm that differentiates between bounded degree graphs that can be clustered into k clusters with good inner conductance (YES case) from graphs that are far from such graphs (NO case), using only $\tilde{O}(n^{\frac{1}{2}}\text{poly}(k))$ queries. This striking progress, however, required an even stronger separation between YES and NO instances of the problem. In particular, the algorithm requires that in the YES case, the graph can be partitioned into k clusters with inner conductance at least ϕ, while in the NO case, the graph is ϵ-far from admitting k clusters with conductance $\phi^2/\log n$. Thus the cluster quality in the NO case needs to be weakened by a factor that now depends on the size of the input graph.

The current state of the art raises several natural questions on both algorithmic and lower bound fronts. On the algorithmic front, does sublinear testing of cluster structure of a graph fundamentally require such strong separation between the cluster structures of YES and NO cases? On the lower bound front, is there a stronger barrier than the current $\Omega(\sqrt{n})$ threshold for differentiating between the YES and NO cases? Even for the case of distinguishing an expander for a graph that is far from expander, the known algorithmic results require $n^{\frac{1}{2}+\Omega(1)}$ queries when the conductance guarantees of YES and NO cases are separated by only a constant factor.

In this work, we make progress on both questions above. On the algorithmic side, we present a new sublinear testing algorithm that considerably weakens the separation required between the conductance of YES and NO instances. In particular, for any fixed k, our algorithm can distinguish between instances that can be partitioned into k clusters with conductance at least ϕ from instances that are $\Omega(1)$-far from admitting k clusters with conductance $\gamma\phi^2$, using $n^{\frac{1}{2}+O(\gamma)}$ queries. This generalizes the results of [KS11, NS10] for $k = 1$ to any fixed k and arbitrary graphs. Similar to [CPS15] our algorithm is based on sampling a small number of vertices and gathering information about the transition probabilities of suitably long random walks from the sampled points. However, instead of classifying points as pairwise similar or dissimilar based on ℓ_2 similarity between the transition probability vectors, our approach is based on analyzing the structure of the Gram matrix of these transition probability vectors, which turns out to be a more robust mechanism for separating the YES and NO cases.

On the lower bound side, we show that arguably the simplest question in this setting, namely, differentiating a bounded degree expander graph with conductance $\Omega(1)$ from a graph that is $\Omega(1)$-far from a graph with conductance γ for some positive constant γ, already requires $n^{\frac{1}{2}+\Omega(\gamma)}$ queries. This improves upon the long-standing previous lower bound of $\Omega(n^{\frac{1}{2}})$. Going past the $n^{\frac{1}{2}}$ threshold requires us to introduce new ideas to handle non-trivial dependencies that manifest due to unavoidable emergence of cycles once an $\omega(n^{\frac{1}{2}})$-sized component is uncovered in an expander. We use a Fourier analytic approach to handle emergence of cycles and create a distribution where $n^{\frac{1}{2}+\Omega(\gamma)}$ queries are necessary to distinguish between YES and NO cases. We believe our lower bound techniques are of independent interest and will quite likely find applications to other problems. As one illustrative application, we show that our approach yields an $n^{\frac{1}{2}+\Omega(1)}$ query complexity lower bound for the problem of approximating the max-cut value in graph to within a factor better than 2, improving the previous best lower bound of $\Omega(n^{\frac{1}{2}})$.

In what follows, we formally define our clustering problem, present our main results, and give an overview of our techniques.

2.1.1 Problem statement

We start by introducing basic definitions, then proceed to define the problems that we design algorithms for (namely **PartitionTesting** and testing clusterability) in Section 2.1.2, and finally discuss the communication game that we use to derive query complexity lower bounds (namely the **NoisyParities** game) and state our results on lower bounds in Section 2.1.3.

Definition 2.1 (**Internal and external conductance**). Let $G = (V_G, E_G)$ be a graph. Let $\deg(v)$ be the degree of vertex v. For a set $S \subseteq V_G$, let $\mathrm{vol}(S) = \sum_{v\in S} \deg(v)$ denote the *volume* of set S. For a set $S \subseteq C \subseteq V_G$, the *conductance of S within C*, denoted by $\phi_C^G(S)$, is the number of edges with one endpoint in S and the other in $C \setminus S$ divided by $\mathrm{vol}(S)$. Equivalently, $\phi_C^G(S)$ is the probability that a uniformly random neighbor, of a vertex in S selected with probability proportional to degree, is in $C \setminus S$. The *internal conductance* of C, denoted by $\phi^G(C)$, is defined to be $\min_{S\subseteq C, 0<\mathrm{vol}(S)\leq \frac{\mathrm{vol}(C)}{2}} \phi_C^G(S)$ if $|C|>1$ and one otherwise. The *external conductance* of C is defined to be $\phi_{V_G}^G(C)$.

Based on the conductance parameters, clusterability and unclusterability of graphs is defined as follows.

Definition 2.2 (**Graph clusterability**)**.** Graph $G=(V_G,E_G)$ is defined to be (k,φ)*-clusterable* if V_G can be partitioned into $C_1,\ldots,C_h$ for some $h\leq k$ such that for all $i=1,\ldots,h$, $\phi^G(C_i)\geq\varphi$. Graph G is defined to be (k,φ,β)*-unclusterable* if V_G contains $k+1$ pairwise disjoint subsets $C_1,\ldots,C_{k+1}$ such that for all $i=1,\ldots,k+1$, $\mathrm{vol}(C_i)\geq\beta\cdot\frac{\mathrm{vol}(V_G)}{k+1}$, and $\phi_{V_G}^G(C_i)\leq\varphi$.

The following algorithmic problem was implicitly defined in [CPS15]:

Definition 2.3. PartitionTesting $(k,\varphi_{\mathrm{in}},\varphi_{\mathrm{out}},\beta)$ is the problem of distinguishing between the following two types of graphs.

1. The YES case: graphs which are $(k,\varphi_{\mathrm{in}})$-clusterable
2. The NO case: graphs which are $(k,\varphi_{\mathrm{out}},\beta)$-unclusterable

The ultimate problem that we would like to solve is the **Clusterability** problem, defined below:

Definition 2.4. Clusterability$(k,\varphi,k',\varphi',\varepsilon)$ is the problem of distinguishing between the following two types of graphs.

1. The YES case: graphs which are (k,φ)-clusterable
2. The NO case: graphs which are ε-far from (k',φ')-clusterable.

Here, a graph $G=(V,E)$ is ε-far from (k',φ')-clusterable if there does not exist a (k',φ')-clusterable graph $G'=(V,E')$ such that $|E\oplus E'|\leq\varepsilon\cdot|E|$ ($\oplus$ denotes the symmetric difference, or equivalently, the Hamming distance).

Note that in the clusterability problem considered by Czumaj et al. [CPS15], the YES instances were required to have clusters with small outer conductance, whereas we have no such requirement.

Queries and Complexity. We assume that the algorithm has access to graph G via the following queries.

1. Vertex query: returns a uniformly random vertex $v\in V_G$
2. Degree query: outputs degree $\deg(v)$ of a given $v\in V_G$.

3. Neighbor query: given a vertex $v \in V_G$, and $i \in [n]$, returns the i-th neighbor of v if $i \leq \deg(v)$, and returns *fail* otherwise.

The complexity of the algorithm is measured by number of access queries.

2.1.2 Algorithmic results

Theorem 2.1. *Suppose $\varphi_{out} \leq \frac{1}{480}\varphi_{in}^2$. Then there exists a randomized algorithm for* **Partition-Testing** $(k, \varphi_{in}, \varphi_{out}, \beta)$ *which gives the correct answer with probability at least* $2/3$*, and which makes* $poly(1/\varphi_{in}) \cdot poly(k) \cdot poly(1/\beta) \cdot poly\log(m) \cdot m^{1/2+O(\varphi_{out}/\varphi_{in}^2)}$ *queries on graphs with m edges.*

Observe that even when the *average* degree of the vertices of the graph is constant, the dependence of query complexity on n, the number of vertices, is $\tilde{O}(n^{1/2+O(\varphi_{\text{out}}/\varphi_{\text{in}}^2)})$. We also note that our current analysis of the tester is probably somewhat loose: the tester likely requires no more than $\tilde{O}(n^{1/2+O(\varphi_{\text{out}}/\varphi_{\text{in}}^2)})$ for graphs of arbitrary volume (specifically, the variance bound provided by Lemma 2.19 can probably be improved).

Theorem 2.1 allows us to obtain the following result on testing clusterability, which removes the logarithmic gap assumption required for the results in [CPS15] in the property testing framework.

Theorem 2.2. *Suppose $\varphi' \leq \alpha_{4.5}\varepsilon$, (for the constant $\alpha_{4.5} = \Theta(\min(d^{-1}, k^{-1}))$ from Lemma 4.5 of [CPS15], where d denotes the maximum degree), and $\varphi' \leq c'\varepsilon^2\varphi^2/k^2$ for some small constant c'. Then there exists a randomized algorithm for* **Clusterability**$(k, \varphi, k, \varphi', \varepsilon)$ *problem on degree d-bounded graphs that gives the correct answer with probability at least* $2/3$*, and which makes* $poly(1/\varphi) \cdot poly(k) \cdot poly(1/\varepsilon) \cdot poly(d) \cdot poly\log(n) \cdot n^{1/2+O(\varepsilon^{-2}k^2 \cdot \varphi'/\varphi^2)}$ *queries on graphs with n vertices.*

The proof of the theorem follows by combining Theorem 2.1 and Lemma 4.5 of [CPS15]. The details of the proof are provided in Section 2.5.

Furthermore, we strengthen Lemma 4.5 of [CPS15] to reduce the dependence of the gap between inner and outer conductance to logarithmic in k, albeit at the expense of a bicriteria approximation. This gives us the following theorem, whose proof is provided in Section 2.5.

Theorem 2.3. *Let $0 \leq \varepsilon \leq \frac{1}{2}$. Suppose $\varphi' \leq \alpha$, (for $\alpha = \min\{\frac{c_{exp}}{150d}, \frac{c_{exp}\cdot\varepsilon}{1400\log(\frac{16k}{\varepsilon})}\}$, where d denotes the maximum degree), and $\varphi' \leq c \cdot \varepsilon^2\varphi^2/\log(\frac{32k}{\varepsilon})$ for some small constant c. Then there exists a randomized algorithm for* **Clusterability**$(k, \varphi, 2k, \varphi', \varepsilon)$ *problem on degree d-bounded graphs that gives the correct answer with probability at least* $2/3$*, and which makes* $poly(1/\varphi) \cdot poly(k) \cdot poly(1/\varepsilon) \cdot poly(d) \cdot poly\log(n) \cdot n^{1/2+O(\varepsilon^{-2}\log(\frac{32k}{\varepsilon})\cdot\varphi'/\varphi^2)}$ *queries on graphs with n vertices.*

2.1.3 Lower bound results

Our lower bounds are based on the following communication problem that we refer to as the **NoisyParities** (d,ε):

Definition 2.5. **NoisyParities** (d,ε) is the problem with parameters $d \geq 3$ and $\varepsilon \leq 1/2$ defined as follows. An adversary samples a random d-regular graph $G = (V,E)$ from the distribution induced by the *configuration model* of Bollobás [Bol80]. The adversary chooses to be in the YES case or the NO case with probability 1/2, and generates a vector of binary edge labels $Y \in \{0,1\}^E$ as follows:

YES case: The vector Y is chosen uniformly at random from $\{0,1\}^E$, that is, the labels $Y(e)$ for all edges $e \in E$ are independently 0 or 1 with probability 1/2;

NO case: A vector $X \in \{0,1\}^V$ is sampled uniformly at random. Independently, a "noise" vector $Z \in \{0,1\}^E$ is sampled such that all the $Z(e)$'s are independent Bernoulli random variables which are 1 with probability ε and 0 with probability $1-\varepsilon$. The label of an edge $e = (u,v) \in E$ is given by $Y(e) = X(u) + X(v) + Z(e)$.

The algorithm can query vertices $q \in V$ in an adaptive manner deterministically. Upon querying a vertex $q \in V$, the algorithm gets the edges incident on q together with their labels as a response to the query, and must ultimately determine whether the adversary was in the YES or the NO case.

Our main result is a tight lower bound on the query complexity of **NoisyParities** . Before stating our lower bound we note that it is easy to see that unless the set of edges that the algorithm has discovered contains a cycle, the algorithm cannot get any advantage over random guessing. Indeed, if the set of discovered edges were a path $P = (e_1,\ldots,e_T)$ where $e_i = (v_{i-1},v_i)$, the label $Y(e_i) = X(v_1) + X(v_{i-1}) + Z(e_i)$ of e_i in the NO case is uniform and independent of the labels of $e_1,\ldots,e_{i-1}$, because $X(v_i)$ is uniform and independent of $Y(e_1)\ldots,Y(e_{i-1})$. A similar argument holds when the set of discovered edges is a forest. Thus, the analysis must, at the very least, prove that $\Omega(\sqrt{n})$ queries are needed in our model for the algorithm to discover a cycle in the underlying graph G. In the noisy case (i.e. when $\varepsilon > 0$) detecting a single cycle does not suffice. Indeed, a natural test would be to add up the labels over the edges of a cycle C, that is, consider $\sum_{e\in C} Y(e)$. In the YES case, this is uniformly 0 or 1, whereas in the NO case, it is equal to $\sum_{e\in C} Z(e)$, which is 0 with probability $(1/2)\cdot(1+(1-2\varepsilon)^{|C|})$ and 1 with probability $(1/2)\cdot(1-(1-2\varepsilon)^{|C|})$. Thus, the deviation of the distribution of $\sum_{e\in C} Y(e)$ from uniform is $n^{-\Theta(\varepsilon)}$, even in the NO case, if $|C| = \Theta(\log n)$.

Theorem 2.4. *Any deterministic algorithm that solves the* **NoisyParities** *problem correctly with probability at least* 2/3 *must make at least* $n^{1/2+\Omega(\varepsilon)}$ *queries on* n*-vertex graphs, for constant* d.

We note that this lower bound is tight up to constant factors multiplying ε in the exponent. For example, it suffices to find $n^{\Theta(\varepsilon)}$ disjoint cycles. This can be done as follows. Sample $n^{\Theta(\varepsilon)}$

vertices in G uniformly at random, and run $\approx \sqrt{n}$ random walks from each of them. With at least constant probability, for most of the seed nodes the walks will intersect. Then for each cycle C_i, compute $\zeta_i = \sum_{e \in C_i} Y(e)$. In the YES case, this is uniformly 0 and 1, whereas in the NO case, it is $n^{-\Theta(\varepsilon)}$-far from uniform. Furthermore, since the cycles are disjoint, ζ_i's are independent. The Chernoff bound implies that, with a constant probability, less than $(1/2) \cdot (1 + n^{-\Theta(\varepsilon)}/2)$ fraction of the ζ_i's will be zero in the YES case, and more than $(1/2) \cdot (1 + n^{-\Theta(\varepsilon)}/2)$ fraction of the ζ_i's will be zero in the NO case.

As a consequence of Theorem 2.4 and appropriate reductions, we derive the following lower bounds.

Theorem 2.5. *Any algorithm that distinguishes between a $(1, \varphi_{in})$-clusterable graph (that is, a φ_{in}-expander) and a $(2, \varphi_{out}, 1)$-unclusterable graph on n vertices (in other words, solves* **PartitionTesting**$(1, \varphi_{in}, \varphi_{out}, 1)$*) correctly with probability at least $2/3$ must make at least $n^{1/2+\Omega(\varphi_{out})}$ queries, even when the input is restricted to regular graphs, for constant φ_{in}.*

Theorem 2.6. *Any algorithm that approximates the maxcut of n-vertex graphs within a factor $2 - \varepsilon'$ with probability at least $2/3$ must make at least $n^{1/2+\Omega(\varepsilon'/\log(1/\varepsilon'))}$ queries.*

Remark 2.1. *After posting our paper on arXiv, we learnt that the above result was already known due to Yoshida (Theorem 1.2 of [Yos11]; the proof appears in the full version [Yos10]). We note, however, that our proof is very different from Yoshida's proof, and may be of independent interest.*

2.1.4 Our techniques

In this section we give an overview of the new techniques involved in our algorithm and lower bounds.

Algorithms

We start by giving an outline the approach of [CPS15], outline the major challenges in designing robust tester of graph cluster structure, and then describe our approach.

As [CPS15] show, the task of distinguishing between (k, φ)-clusterable graphs and graphs that are ϵ-far from (k, φ')-clusterable reduces to **PartitionTesting** $(k, \varphi_{\text{in}}, \varphi_{\text{out}}, \beta)$, where $\beta = \text{poly}(\epsilon)$. In this problem we are given query access to a graph G, and would like to distinguish between two cases: either the graph can be partitioned into at most k clusters with inner conductance at least φ_{in} (the **YES** case, or 'clusterable' graphs) or there exists at least $k+1$ subsets $C_1, \ldots, C_{k+1}$ with outer conductance at most φ_{out}, and containing nontrivial (i.e. no smaller than $\beta n/(k+1)$) number of nodes (the **NO** case, or 'non-clusterable' graphs). Here $\varphi_{\text{in}} = \varphi$, and φ_{out} is a function of the conductance φ', the number of nodes k, and the precision parameter ϵ.

A very natural approach to **PartitionTesting** $(k, \varphi_{\text{in}}, \varphi_{\text{out}}, \beta)$ is to sample $10k$ nodes, say, run random walks of appropriate length from the sampled nodes, and compare the resulting distributions: if a pair of nodes is in the same cluster, then the distributions of random walks should be 'close', and if the nodes are in different clusters, the distributions of random walks should be 'far'. The work of [CPS15] shows that this high level approach can indeed be made to work: if one compares distributions in ℓ_2 norm, then for an appropriate separation between φ_{in} and φ_{out} random walks whose distributions are closer than a threshold θ in ℓ_2 sense will indicate that the starting nodes are in the same cluster, and if the distributions are further than 2θ apart in ℓ_2, say, then the starting points must have been in different clusters. Using an ingenious analysis [CPS15] show that one can construct a graph on the sampled nodes where 'close' nodes are connected by an edge, and the original graph is clusterable if and only if the graph on the sampled nodes is a union of at most k connected components. The question of estimating ℓ_2 norm distance between distributions remains, but this can be done in about $\sqrt{n}$ time by estimating collision probabilities (by the birthday paradox), or by using existing results in the literature. The right threshold θ turns out to be $\approx 1/\sqrt{n}$.

The main challenge. While very beautiful, the above approach unfortunately does not work unless the cluster structure in our instances is very pronounced. Specifically, the analysis of [CPS15] is based on arguing that random walks of $O(\log n)$ length from sampled nodes that come from the same cluster mostly don't leave the cluster, and this is true only if the outer conductance of the cluster is no larger than $1/\log n$. This makes the approach unsuitable for handling gaps between conductances that are smaller than $\log n$ (it is not hard to see that the walk length must be at least logarithmic in the size of the input graph, so shortening the walk will not help).

One could think that this is a question of designing of a more refined analysis of the algorithm of [CPS15], but the problem is deeper: it is, in general, not possible to choose a threshold θ that will work even if the gap between conductances is constant, and even if we want to distinguish between 2-clusterable and far from 2-clusterable graphs (such a choice is, in fact, possible for $k = 1$). The following simple example illustrates the issue. First consider a d-regular graph G composed of two $\Omega(1)$-expanders A and B, each of size $\frac{n}{2}$. Further, suppose that the outer conductance of both A and B is upper bounded by $\frac{1}{4d}$, i.e. at most one quarter of the nodes in each of the clusters have connections to nodes on the other side. Let $t = C \log n$ for a constant $C > 0$, and let $\mathbf{p}_u^t$ denote the probability distribution of t-step random walk starting from vertex u. Lemma C.1 of [CPS15] implies that for at least one of these two clusters (say A), $||\mathbf{p}_u^t - \mathbf{p}_v^t||_2 = \Omega(d^{-2}n^{-1/2})$ for all u, v in some large subset $\tilde{A}$ of A. Thus, any tester that considers two sampled vertices close when their Euclidean distance is at most θ must use $\theta > d^{-2}n^{-1/2}$. On the other hand, consider the following 3-clusterable instance.

Fix $\epsilon \in (0,1)$, and let C be regular graph with degree $\frac{1}{\epsilon} - 3$, inner conductance $\Omega(1)$ and size $\frac{n}{3}$. Let $G' = (V, E)$ be a $1/\epsilon$-regular graph composed of three copies of C (say C_1, C_2, C_3), where for each vertex $u \in C$, its three copies in C_1, C_2, C_3 are pairwise connected (i.e. form a triangle), and each vertex has a self-loop. We say that an edge is bad if it is a triangle edge or self-loop, otherwise we call it good. Notice that at any step the random walk takes a bad edge with probability 3ϵ, and takes an edge inside one of the copies with probability $1-3\epsilon$. We can think that at any step, the random walk first decides to take a good edge or a bad edge, and then takes a random edge accordingly. With probability $(1-3\epsilon)^t$ the random walk never decides to take a bad edge and mixes inside the starting copy. On the other hand, if the random walk does decide to take a bad edge, it is thereafter equally likely to be in any of the three copies of any vertex of C. Let $t = c\log n$ for a constant $c > 0$, and let $\mathbf{p}_u^t$ denote the probability distribution of t-step random walk starting from vertex u. We are interested in bounding $||\mathbf{p}_u^t - \mathbf{p}_v^t||_2$ for a pair of nodes u, v in different clusters (say $u \in C_1$ and $v \in C_2$). For $u \in C_1$ and $a \in V$, let $\mathbf{q}_u^{t'}(a)$ denote the probability that a t'-step random walk in C_1 starting from u ends up in the copy of a in C_1. Notice that since C_1 is constant-expander, if $t' = \Theta(t)$, then for every a we have $\mathbf{q}_u^{t'}(a) \simeq \frac{1}{n/3}$. Consider a t-step random walk from u in G, and let t' denote the number good edges taken. Notice that t' is binomially distributed with parameters $(t, 3\epsilon)$, so that $t' \simeq t(1-3\epsilon)$ with high probability. Now, for $a \in C_1$, we have,

$$\mathbf{p}_u^t(a) = (1-3\epsilon)^t \cdot \mathbf{q}_u^t(a) + \frac{1}{3}(1-(1-3\epsilon)^t)\mathbb{E}_{t'}[\mathbf{q}_u^{t'}(a)|t'>0] \simeq 3n^{-1-3c\epsilon} + \left(1-n^{-3c\epsilon}\right)n^{-1},$$

while for $b \notin C_1$, we have $\mathbf{p}_u^t(b) = \frac{1}{3}(1-(1-3\epsilon)^t)\mathbb{E}_{t'}[\mathbf{q}_u^{t'}(b)|t'>0] \simeq \left(1-n^{-3c\epsilon}\right)n^{-1}$. A symmetric argument holds for v. Hence we have $||\mathbf{p}_u^t - \mathbf{p}_v^t||_2^2 \le \Theta(n^{-1-6c\epsilon})$. Therefore for a pair of nodes u, v in different clusters one has $||p_u^t - p_v^t||_2 \le n^{-1/2-\Omega(\epsilon)} \ll d^{-2}n^{-1/2}$ for constant d and ϵ. Thus, in order to ensure that vertices in different clusters will be considered far, one must take the threshold θ to be smaller than $d^{-2}n^{-1/2}$. Thus, no tester that uses a fixed threshold can distinguish between the two cases correctly. To summarize, euclidean distance between distributions is no longer a reliable metric if one would like to operate in a regime close to theoretical optimum, and a new proxy for clusterability is needed.

Our main algorithmic ideas. Our main algorithmic contribution is a more geometric approach to analyzing the proximity of the sampled points: instead of comparing ℓ_2 distances between points, our tester considers the Gram matrix of the random walk transition probabilities of the points, estimates this matrix entry-wise to a precision that depends on the gap between φ_{in} and φ_{out} in the instance of **PartitionTesting** $(k, \varphi_{\text{in}}, \varphi_{\text{out}}, \beta)$ that we would like to solve, and computes the $(k+1)$-st largest eigenvalue of the matrix. This quantity turns out to be a more robust metric, yielding a tester that operates close to the theoretical optimum, i.e. able to solve **PartitionTesting** $(k, \varphi_{\text{in}}, \varphi_{\text{out}}, \beta)$ as long as the gap $\varphi_{\text{out}}/\varphi_{\text{in}}^2$ is smaller than an absolute constant.[1] Specifically, our tester (see Algorithms 1 and 2 in Section 2.3.1 for the most basic version) samples a multiset S of $s \approx \text{poly}(k)\log n$ vertices of the graph G independently and with probability proportional to the degree distribution (this can be achieved in $\approx \sqrt{n}$ time per sample using the result of Eden and Rosenbaum [ER18]), and computes the matrix

$$A := (D^{-\frac{1}{2}}M^tS)^T(D^{-\frac{1}{2}}M^tS), \tag{2.1}$$

[1] Note that our runtime depends on $\varphi_{\text{out}}/\varphi_{\text{in}}^2$ as opposed to $\varphi_{\text{out}}/\varphi_{\text{in}}$ due to a loss in parameters incurred through Cheeger's inequality. This loss is quite common for spectral algorithms.

where M is the random walk transition matrix of the graph G, and D is the diagonal matrix of degrees. Note that this is the Gram matrix of the t-step distributions of random walks from the sampled nodes in G, for a logarithmic number of steps walk. Intuitively, the matrix A captures pairwise collision probabilities of random walks from sampled nodes, weighted by inverse degree. The algorithm accepts the graph if the $(k+1)$-st largest eigenvalue of the matrix A is below a threshold, and rejects otherwise. Specifically, the algorithm accepts if $\mu_{k+1}(A) \lesssim \text{vol}(V_G)^{-1-\Theta(\varphi_{\text{out}}/\varphi_{\text{in}}^2)}$ and rejects otherwise. Before outlining the proof of correctness for the tester, we note that, of course, the tester above cannot be directly implemented in sublinear time, as computing the matrix A exactly is expensive. The actual sublinear time tester approximately computes the entries of the matrix A to additive precision about $\frac{1}{\text{poly}(k)}\text{vol}(V_G)^{-1-\Theta(\varphi_{\text{out}}/\varphi_{\text{in}}^2)}$ and uses the eigenvalues of the approximately computed matrix to decide whether to accept or reject. Such an approximation can be computed in about $\text{vol}(V_G)^{\frac{1}{2}+\Theta(\varphi_{\text{out}}/\varphi_{\text{in}}^2)}$ queries by rather standard techniques (see Section 2.3.2).

We now outline the proof of correctness of the tester above (the detailed proof is presented in Section 2.3.1). It turns out to be not too hard to show that the tester accepts graphs that are (k, φ_{in})-clusterable. One first observes that Cheeger's inequality together with the assumption that each of the k clusters is a φ_{in}-expander implies that the $(k+1)$-st eigenvalue of the normalized Laplacian of G is at least $\varphi_{\text{in}}^2/2$ (Lemma 2.10). It follows that the matrix M^t of t-step random walk transition probabilities, for our choice of $t = (C/\varphi_{\text{in}}^2)\log n$, is very close to a matrix of rank at most k, and thus the $(k+1)$-st eigenvalue of the matrix A above (see (2.1)) is smaller than $1/n^2$, say. The challenging part is to show that the tester rejects graphs that are $(k, \varphi_{\text{out}}, \beta)$-unclusterable, since in this case we do not have any assumptions on the inner structure of the clusters $C_1, \ldots, C_{k+1}$. The clusters $C_1, \ldots, C_{k+1}$ could either be good expanders, or, for instance, unions of small disconnected components. The random walks from nodes in those clusters behave very differently in these two cases, but the analysis needs to handle both. Our main idea is to consider a carefully defined $k+1$-dimensional subspace of the eigenspace of the normalized Laplacian of G that corresponds to small (smaller than $O(\varphi_{\text{out}})$) eigenvalues, and show that our random sample of points is likely to have a well-concentrated projection onto this subspace. We then show that this fact implies that the matrix A in (2.1) has a large $(k+1)$-st eigenvalue with high probability. The details of the argument are provided in Section 2.3.1: the definition of matrix U and projection operator P_h at the beginning of Section 2.3.1 yield the $(k+1)$-dimensional subspace in question (this subspace is the span of the columns of U projected onto the first h eigenvectors of the normalized Laplacian of G), and a central claims about the subspace in question are provided by Lemmas 2.13, 2.14 and 2.15. The assumption that vertices in S are sampled with probabilities proportional to their degrees is crucial to making the proof work for general (sparse) graphs.

One consequence of the fact that our algorithm for **PartitionTesting** $(k, \varphi_{\text{in}}, \varphi_{\text{out}}, \beta)$ estimates the entries of the Gram matrix referred to above to additive precision $\approx n^{-1-\Theta(\varphi_{\text{out}}/\varphi_{\text{in}}^2)}$ is that the runtime $\approx n^{1/2+\Theta(\varphi_{\text{out}}/\varphi_{\text{in}}^2)}$. If $\varphi_{\text{out}} \ll \varphi_{\text{in}}^2/\log n$, then we recover the $\approx \sqrt{n}$ runtime of [CPS15], but for any constant gap between φ_{out} and φ_{in}^2 our runtime is polynomially larger than $\sqrt{n}$. Our main contribution on the lower bound side is to show that this dependence is necessary. We outline our main ideas in that part of the paper now.

The lower bound

We show that the $n^{1+\Omega(\varphi_{\text{out}}/\varphi_{\text{in}}^2)}$ runtime is necessary for **PartitionTesting** $(k,\varphi_{\text{in}},\varphi_{\text{out}},\beta)$ problem, thereby proving that our runtime is essentially best possible for constant k. More precisely, we show that even distinguishing between an expander and a graph that contains a cut of sparsity ϵ for $\epsilon \in (0,1/2)$ requires $n^{1+\Omega(\epsilon)}$ adaptive queries, giving a lower bound for the query complexity (and hence runtime) of **PartitionTesting** $(1,\Omega(1),\epsilon,1)$ that matches our algorithm's performance.

The NoisyParities problem. Our main tool in proving the lower bound is a new communication complexity problem (the **NoisyParities** problem) that we define and analyze: an adversary chooses a regular graph $G=(V,E)$ and a hidden binary string $X \in \{0,1\}^V$, which can be thought of as encoding a hidden bipartition of G. The algorithm can repeatedly (and adaptively) query vertices of G. Upon querying a vertex v, the algorithm receives the edges incident on v and a binary label $Y(e)$ on each edge e. In the **NO** case the labels $Y(e)$ satisfy $Y(e)=X(u)+X(v)+Z(e)$, where $Z(e)$ is an independent Bernoulli random variable with expectation ϵ (i.e. the algorithm is told whether the edge crosses the hidden bipartition, but the answer is noisy). In the **YES** case each label $Y(e)$ is uniformly random in $\{0,1\}$. The task of the algorithm is to distinguish between the two cases using the smallest possible number of queries to the graph G.

It is easy to see that if $\epsilon=0$, then the algorithm can get a constant advantage over random guessing as long as it can query all edges along a cycle in G. If G is a random d-regular graph unknown to the algorithm, one can show that this will take at least $\Omega(\sqrt{n})$ queries, recovering the lower bound for expansion testing due to Goldreich and Ron [GR02]. In the noisy setting, however, detecting a single cycle is not enough, as cycles that the algorithm can locate in a random regular graph using few queries are generally of logarithmic length, and the noise added to each edge compounds over the length of the cycle, leading to only advantage of about $n^{-O(\epsilon)}$ over random guessing that one can obtain from a single cycle. Intuitively. this suggests that the algorithm should find at least $n^{\Omega(\epsilon)}$ cycles in order to get a constant advantage. Detecting a single cycle in an unknown sparse random graphs requires about $\sqrt{n}$ queries, which together leads to the $n^{1/2+\Omega(\epsilon)}$ lower bound. Turning this intuition into a proof is challenging, however, as **(a)** the algorithm may base its decisions on labels that it observes on its adaptively queried subgraph of G and **(b)** the algorithm does not have to base its decision on observed parities over cycles. We circumvent these difficulties by analyzing the distribution of labels on the edges of the subgraph that the algorithm queries in the **NO** case and proving that this distribution is close to uniformly random in total variation distance, with high probability over the queries of the algorithm. We analyze this distribution using a Fourier analytic approach, which we outline now.

Suppose that we are in the **NO** case, i.e. the edge labels presented to the algorithm are an

ϵ-noisy version of parities of the hidden boolean vector $X \in \{0,1\}^n$, and suppose that the algorithm has discovered a subset $E_{\text{query}} \subseteq E_G$ of edges of the graph G (recall that the graph G, crucially, is not known to the algorithm) together with their labels. The central question that our analysis needs to answer in this situation turns out to be the following: given the observed labels on edges in E_{query} and an edge $e = (a,b) \in E_G$ what is the posterior distribution of $X(a) + X(b)$ given the information that the algorithm observed so far? For example, if E_{query} does not contain any cycles (i.e. is a forest), then $X(a) + X(b)$ is a uniformly random Bernoulli variable with expectation $1/2$ if the edge (a,b) does not close a cycle when added to E_{query}. If it does close a cycle but E_{query} is still a forest, then one can show that if the distance in E_{query} from a to b is large (at least $\Omega(\log n)$), then the posterior distribution of $X(a) + X(b)$ is still $n^{-\Omega(\epsilon)}$ close, in total variation distance, to a Bernoulli random variable with expectation $1/2$. Our analysis needs to upper bound this distance to uniformity for a 'typical' subset E_{query} that arises throughout the interaction process of the algorithm with the adversary, and contains two main ideas. First, we show using Fourier analytic tools (see Theorem 2.12 in Section 2.4.3) that for 'typical' subset of queried edges E_{query} and any setting of observed labels, one has that the bias of $X(a) + X(b)$, i.e. the absolute deviation of the expectation of this Bernoulli random variable from $1/2$, satisfies

$$\text{bias}(X(a) + X(b)) \lesssim \sum_{E' \subseteq E_{\text{query}} \text{ s.t. } E' \cup \{a,b\} \text{ is Eulerian}} (1-2\epsilon)^{|E'|}. \tag{2.2}$$

Note that for the special case of E_{query} being a tree, the right hand side is exactly the $(1-2\epsilon)^{\text{dist}(a,b)}$, where $\text{dist}(a,b)$ stands for the shortest path distance from a to b in T. Since 'typical' cycles that the algorithm will discover will be of $\Omega(\log n)$ length due to the fact that G is a constant degree random regular graph, this is $n^{-\Omega(\epsilon)}$, as required. Of course, the main challenge in proving our lower bound is to analyze settings where the set of queried edges E_{query} is quite far from being a tree, and generally contains many cycles, and control the sum in (2.2). In other words, we need to bound the weight distribution of Eulerian subgraphs of E_{query}. The main insight here is the following structural claim about 'typical' sets of queried edges E_{query}: we show that for typical interaction scenarios between the algorithm and the adversary one can decompose E_{query} as $E_{\text{query}} = F \cup R$, where F is a forest and R is a small (about $n^{O(\epsilon)}$ size) set of 'off-forest' edges that further satisfies the property that the endpoints of edges in R are $\Omega(\log n)$-far from each other in the shortest path metric induced by F. This analysis relies on basic properties of random graphs with constant degrees and is presented in Section 2.4.3. Once such a decomposition of $E_{\text{query}} = T \cup F$ is established, we get a convenient basis for the cycle space of E_{query}, which lets us control the right hand side in (2.2) as required (see Section 2.4.3). The details of the lower bound analysis are presented in Section 2.4.3.

Finally, our lower bound on the query complexity of **NoisyParities** yields a lower bound for **PartitionTesting** $(1, \Omega(1), \epsilon, 1)$ (Theorem 2.5), as well as a lower bound for better than factor 2 approximation to MAX-CUT value in sublinear time (Theorem 2.6). Both reductions are presented in Section 2.4.2. The reduction to MAX-CUT follows using rather standard techniques (e.g. is very similar to [KKSV17]; see Section 2.4.2). The reduction to **PartitionTesting**

$(1, \Omega(1), \epsilon, 1)$ is more delicate and novel: the difficulty is that we need to ensure that the introduction of random noise Z_e on the edge labels produces graphs that have the expansion property (in contrast, the MAX-CUT reduction produces graphs with a linear fraction of isolated nodes). This reduction is presented in Section 2.4.2.

2.1.5 Related work

Goldreich and Ron [GR02] initiated the framework of testing graph properties via neighborhood queries. In this framework, the goal is to separate graphs having a certain property from graphs which are "far" from having that property, in the sense that they need many edge additions and deletions to satisfy the property. The line of work closest to this paper is the one on testing expansion of graphs [GR00, NS10, CS10a, KS11] which proves that expansion testing can be done in about $\tilde{O}(\sqrt{n})$ queries, and $\Omega(\sqrt{n})$ queries are indeed necessary. Going beyond expansion (that is, 1-clusterability), Kannan et al. [KVV04a] introduced (internal) conductance as a measure of how well a set of vertices form a cluster. In order to measure the quality of a clustering, that is, a partition of vertices into clusters, Zhu et al. [ALM13a] and Oveis Gharan and Trevisan [GT14b] proposed bi-criteria measures which take into account the (minimum) internal conductance and the (maximum) external conductance of the clusters. Considering this measure, Czumaj et al. [CPS15] defined the notion of clusterable graphs parameterized requirements on the minimum internal expansion and by the maximum external expansion, and gave an algorithm for testing clusterability.

There has been an extensive work on testing many other graph properties in the framework of Goldreich and Ron. For instance, Czumaj et al. [CGR+14] give algorithms for testing several properties including cycle-freeness, whereas Eden et al. [ELR18] design algorithms to test arboricity. Estimation of graph parameters such as degree distribution moments [ERS17b], number of triangles [ELRS17], and more generally, number of k-cliques [ERS17a] has also received attention recently.

A closely related model of property testing is the one where the graph arrives as a random order stream and the property testing algorithm is required to use sublinear space. Although this appears to be a less powerful model because the algorithm no longer has the ability to execute whatever queries it wants, interestingly, Peng and Sohler [PS18] show that sublinear property testing algorithms give rise to sublinear space algorithms for random order streams.

Other graph property testing models include extension to dense graphs [GR10, GR11a] where the algorithm queries the entries of the adjacency matrix of the graph, and the non-deterministic property testing model [LV13, GS13], where the algorithm queries the graph and a certificate, and must decide whether the graph satisfies the property. We refer the reader to [CPS15] for a more comprehensive survey of the related work.

2.2 Preliminaries

Let $G = (V_G, E_G)$ be a graph and let A be its adjacency matrix.

Definition 2.6. The *normalized adjacency matrix* $\overline{A}$ of G is $D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$, where D is the diagonal matrix of the degrees. The *normalized Laplacian* of G is $L = I - \overline{A}$.

Definition 2.7. The *random walk associated with* G is defined to be the random walk with transition matrix $M = \frac{I+AD^{-1}}{2}$. Equivalently, from any vertex v, this random walk takes every edge of G incident on v with probability $\frac{1}{2\cdot\deg(v)}$, and stays on v with probability $\frac{1}{2}$. We can write the transition matrix as $M = D^{\frac{1}{2}}\overline{M}D^{-\frac{1}{2}}$, where $\overline{M} = I - \frac{L}{2}$.

To see the equivalence of the two definitions of M above, observe that the transition matrix is $M = \frac{I+AD^{-1}}{2} = (\frac{D+A}{2})D^{-1}$ and $\overline{M} = I - \frac{L}{2} = \frac{I+\overline{A}}{2} = D^{-\frac{1}{2}}(\frac{D+A}{2})D^{-\frac{1}{2}}$. Hence, $M = D^{\frac{1}{2}}\overline{M}D^{-\frac{1}{2}}$.

Our algorithm and analysis use spectral techniques, and therefore, we setup the following notation.

- $0 \le \lambda_1 \le \ldots \le \lambda_n \le 2$ are the eigenvalues of L, the normalized Laplacean of G. Λ is the diagonal matrix of these eigenvalues in ascending order.
- $(v_1, \ldots, v_n)$ is an orthonormal basis of eigenvectors of L, with $Lv_i = \lambda_i v_i$ for all i. $V \in \mathbb{R}^{V_G \times [n]}$ is the matrix whose columns are the orthonormal eigenvectors of L arranged in increasing order of eigenvalues. Thus, $LV = V\Lambda$.
- Observe that each v_i is also an eigenvector of $\overline{M}$, with eigenvalue $1 - \frac{\lambda_i}{2}$. Σ is the diagonal matrix of the eigenvalues of $\overline{M}$ in descending order. Then $\Sigma = I - \Lambda/2$ and $\overline{M}V = V\Sigma$.
- For a vertex $a \in V_G$, $\mathbb{1}_a \in \mathbb{R}^{V_G}$ denotes the indicator of a, that is, the vector which is 1 at a and 0 elsewhere. Fix some total order on V_G. For a (multi) set $S = \{a_1, \ldots, a_s\}$ of vertices from V_G where $a_1, \ldots, a_s$ are sorted, we abuse notation and also denote by S the $V_G \times s$ matrix whose i^{th} column is $\mathbb{1}_{a_i}$.
- For a symmetric matrix B, $\mu_h(B)$ (resp. $\mu_{\max}(B)$ $\mu_{\min}(B)$) denotes the h^{th} largest (resp. maximum, minimum) eigenvalue of B.

Claim 2.1. *Let $V \in \mathbb{R}^{V_G \times [n]}$ be the matrix whose columns are the orthonormal eigenvectors of $\overline{M}$ arranged in descending order of eigenvalues. Let Σ denote the diagonal matrix of the eigenvalues of $\overline{M}$. Then*

$$V^T D^{-\frac{1}{2}} M = \Sigma V^T D^{-\frac{1}{2}} \textit{ and, } M^T D^{-\frac{1}{2}} V = D^{-\frac{1}{2}} V \Sigma.$$

Proof. Notice that for each v_i, we can write $v_i^T D^{-\frac{1}{2}} M$ as $v_i^T D^{-\frac{1}{2}}(D^{\frac{1}{2}}\overline{M}D^{-\frac{1}{2}}) = v_i^T \overline{M} D^{-\frac{1}{2}} = (1 - \frac{\lambda_i}{2}) v_i^T D^{-\frac{1}{2}}$. Hence, $v_i^T D^{-\frac{1}{2}}$ is a left eigenvector of M with eigenvalue $1 - \frac{\lambda_i}{2}$. Similarly, $D^{-\frac{1}{2}} v_i$ is a right eigenvector of M^T with eigenvalue $1 - \frac{\lambda_i}{2}$. Then we have $V^T D^{-\frac{1}{2}} M = \Sigma V^T D^{-\frac{1}{2}}$ and $M^T D^{-\frac{1}{2}} V = D^{-\frac{1}{2}} V \Sigma$. □

We will use the following standard results on matrix norms and eigenvalues.

Lemma 2.1. *Frobenius norm $\|\cdot\|_F$ (resp. spectral norm $\mu_{\max}(\cdot)$) is submultiplicative on all (resp. positive semidefinite) matrices. That is, for any two $m \times m$ (positive semidefinite) matrices A and B, $\|AB\|_F \le \|A\| \cdot \|B\|$ (resp. $\mu_{\max}(AB) \le \mu_{\max}(A) \cdot \mu_{\max}(B)$).*

The following is a result from [HJ90] (Theorem 1.3.20 on page 53).

Lemma 2.2. *For any $m \times n$ matrix A and any $n \times m$ matrix B, the multisets of nonzero eigenvalues of AB and BA are equal. In particular, if one of AB and BA is positive semidefinite, then $\mu_h(AB) = \mu_h(BA)$.*

Lemma 2.3 (Weyl's Inequality)**.** *Let A and E be symmetric $m \times m$ matrices. Then for all $i = 1, \ldots, m$, $\mu_i(A) + \mu_{\min}(E) \le \mu_i(A+E) \le \mu_i(A) + \mu_{\max}(E)$.*

The next linear algebraic lemma will be useful in our analysis. The (simple) proof is given in Appendix A.1.

Lemma 2.4. *Let A be an $m \times n$ matrix, V be a $m \times p$ matrix with orthonormal columns, and U be a $n \times q$ matrix with orthonormal columns. Then for all $h = 1, \ldots, n$,*

1. $\mu_h(A^T A) \ge \mu_h(A^T V V^T A)$.
2. $\mu_h(A^T A) \ge \mu_h(U^T A^T A U)$.

Lemma 2.5 (Courant-Fischer)**.** *Let A be a symmetric $n \times n$ matrix with eigenvalues $\lambda_1 \ge \lambda_2 \ge \ldots \ge \lambda_n$ If $\mathcal{V}_k$ denotes the set of subspaces of $\mathbb{R}^n$ of dimension k, then*

$$\lambda_k = \max_{W \in \mathcal{V}_k} \min_{x \in W, w \neq 0} \frac{x^T A x}{x^T x}.$$

Lemma 2.6 (Gershgorin Circle Theorem)**.** *Let Q be a $n \times n$ matrix, with entries q_{ij}. For $i \in [n]$, let $R_i = \sum_{j \neq i} |q_{ij}|$ be the sum of the absolute values of the non-diagonal entries in the i-th row. Let $D(q_{ii}, R_i)$ be the closed disc centered at q_{ii} with radius R_i. Such a disc is called a Gershgorin disc. Every eigenvalue of Q lies within at least one of the Gershgorin discs $D(q_{ii}, R_i)$.*

2.3 Algorithm for partition testing

The goal of this section is to present an algorithm for the **PartitionTesting** problem, analyze it, and hence, prove Theorem 2.1. We restate this theorem here for the reader's convenience, and its proof appears at the end of Section 2.3.2.

Theorem 2.1 (restated). Suppose $\varphi_{\text{out}} \le \frac{1}{480}\varphi_{\text{in}}^2$. Then there exists a randomized algorithm for **PartitionTesting** $(k, \varphi_{\text{in}}, \varphi_{\text{out}}, \beta)$ which gives the correct answer with probability at least $2/3$, and which makes $\text{poly}(1/\varphi_{\text{in}}) \cdot \text{poly}(k) \cdot \text{poly}(1/\beta) \cdot \text{polylog}(m) \cdot m^{1/2+O(\varphi_{\text{out}}/\varphi_{\text{in}}^2)}$ queries on graphs with m edges.

Towards proving this theorem, we first make the following simplifying assumption. We assume that we have the following oracle at our disposal: the oracle takes a vertex a as input, and returns $D^{-\frac{1}{2}}M^t\mathbb{1}_a$, where D is the diagonal matrix of the vertex degrees, M is the transition matrix of the lazy random walk associated with the input graph, and $\mathbb{1}_a$ is the indicator vector of a. We first present and analyze, in Section 2.3.1, an algorithm for **PartitionTesting** which makes use of this oracle. Following this, in Section 2.3.2, we show how the oracle can be (approximately) simulated, and thereby, get an algorithm for **PartitionTesting**.

We remark that our algorithms use the value of $\mathrm{vol}(V_G)$, which is not available directly through the access model described in Section 2.1.1. However, by the result of [Ses15], it is possible to approximate the value of $\mathrm{vol}(V_G)$ with an arbitrarily small multiplicative error using $\tilde{O}(\sqrt{|V_G|})$ queries.

2.3.1 The algorithm under an oracle assumption

Let $G = (V_G, E_G)$ be a graph and let A be its adjacency matrix. Recall that in Definition 2.7 we associated with such a graph the random walk given by the transition matrix $M = \frac{I+AD^{-1}}{2}$. That is, from any vertex, the walk takes each edge incident on the vertex with probability $\frac{1}{2\cdot\deg(v)}$, and stays at the same vertex with probability $\frac{1}{2}$. Fix t, the length of the random walk. For this section, we assume that we have the following oracle at our disposal: the oracle takes a vertex $a \in V_G$ as input, and returns $D^{-\frac{1}{2}}M^t\mathbb{1}_a$. Our algorithm for **PartitionTesting** is given by Algorithm 2 called PARTITIONTEST. The goal of this section is to prove guarantees about this algorithm, as stated in the following theorem.

Theorem 2.7. *Suppose $\varphi_{in}^2 > 480\varphi_{out}$. For every graph G, integer $k \geq 1$, and $\beta \in (0,1)$,*

1. *If G is (k, φ_{in})-clusterable (YES case), then* PARTITIONTEST*$(G, k, \varphi_{in}, \varphi_{out}, \beta)$ accepts.*
2. *If G is $(k, \varphi_{out}, \beta)$-unclusterable (NO case), then* PARTITIONTEST*$(G, k, \varphi_{in}, \varphi_{out}, \beta)$ rejects with probability at least $\frac{2}{3}$.*

Algorithm 1 ESTIMATE(G, k, s, t, η)

```
Sample $s$ vertices from $V_G$ independently and with probability proportional to the degree
    of the vertices at random with replacement using sampler$(G,\eta)$ (See Lemma 2.7). Let $S$ be
    the multiset of sampled vertices.
Compute $D^{-\frac{1}{2}}M^tS$ using the oracle.
Return $\mu_{k+1}((D^{-\frac{1}{2}}M^tS)^T(D^{-\frac{1}{2}}M^tS))$.
```

Algorithm PARTITIONTEST calls the procedure ESTIMATE given by Algorithm 1, compares the value returned with a threshold, and then decides whether to accept or reject. Procedure ESTIMATE needs to draw several samples of vertices, where each vertex of the input graph is sampled with probability proportional to its degree. This, by itself, is not allowed in the query model under consideration defined in Section 2.1.1. Therefore, procedure ESTIMATE makes

Algorithm 2 $\textsc{PartitionTest}(G, k, \varphi_{\text{in}}, \varphi_{\text{out}}, \beta)$ ▷ Need: $\varphi_{\text{in}}^2 > 480\varphi_{\text{out}}$

```
$\eta := 0.5$.
$s := 1600(k+1)^2 \cdot \ln(12(k+1)) \cdot \ln(\text{vol}(V_G))/(\beta(1-\eta))$.
$c := \frac{20}{\varphi_{\text{in}}^2}$, $t := c\ln(\text{vol}(V_G))$                ▷ Observe: $c > 0$.
$\mu_{\text{thres}} := \frac{1}{2} \cdot \frac{8(k+1)\ln(12(k+1))}{\beta\cdot(1-\eta)} \times \text{vol}(V_G)^{-1-120c\varphi_{\text{out}}}$.
if $\textsc{Estimate}(G, k, s, t, \eta) \le \mu_{\text{thres}}$ then
    Accept $G$.
else
    Reject $G$.
```

use of the following result by Eden and Rosenbaum to (approximately) sample vertices with probabilities proportional to degree.

Lemma 2.7 (Corollary 1.5 of [ER18]). *Let $G = (V_G, E_G)$ be an arbitrary graph, and $\eta > 0$. Let $\mathcal{D}$ denote the degree distribution of G (i.e., $\mathcal{D}(v) = \frac{\deg(v)}{vol(G)}$). Then there exists an algorithm, denoted by **sampler**(G, η), that with probability at least $\frac{2}{3}$ produces a vertex v sampled from a distribution $\mathcal{P}$ over V_G, and outputs "Fail" otherwise. The distribution $\mathcal{P}$ is such that for all $v \in V_G$,*

$$|\mathcal{P}(v) - \mathcal{D}(v)| \le \eta \cdot \mathcal{D}(v).$$

The algorithm uses $\tilde{O}\left(\frac{|V_G|}{\sqrt{\eta \cdot vol(V_G)}}\right)$ vertex, degree and neighbor queries.

The proof of Theorem 2.7 relies on the following guarantees about the behavior of the algorithm in the YES case, and the NO case respectively, whose proofs are given in Section 2.3.1 and Section 2.3.1 respectively.

Theorem 2.8. *Let $\varphi_{in} > 0$ and integer $k \ge 1$. Then for every (k, φ_{in})-clusterable graph $G = (V_G, E_G)$ (see definition 2.2), with $\min_{v \in V_G} deg(v) \ge 1$ the following holds:*

$$\textsc{Estimate}(G, k, s, t, \eta) \le s \cdot \left(1 - \frac{\varphi_{in}^2}{4}\right)^{2t}.$$

Theorem 2.9. *Let $\varphi_{out} > 0$, $\beta \in (0,1)$, and integer $k \ge 1$. Let*

$$s = 1600(k+1)^2 \cdot \ln(12(k+1)) \cdot \ln(vol(V_G))/(\beta \cdot (1-\eta)).$$

Then for every $(k, \varphi_{out}, \beta)$-unclusterable graph $G = (V_G, E_G)$ (see definition 2.2), with $\min_{v \in V_G} deg(v) \ge 1$, the following holds with probability at least $\frac{2}{3}$.

$$\textsc{Estimate}(G, k, s, t, \eta) \ge \frac{8(k+1)\ln(12(k+1))}{\beta \cdot (1-\eta) \cdot vol(V_G)} \times (1 - 30\varphi_{out})^{2t}.$$

Proof of Theorem 2.7. Let $t = c\ln(\mathrm{vol}(V_G))$ for $c = \frac{20}{\varphi_{\mathrm{in}}^2}$. We call the procedure ESTIMATE with

$$s = 1600(k+1)^2 \cdot \ln(12(k+1)) \cdot \ln(\mathrm{vol}(V_G))/(\beta \cdot (1-\eta)),$$

and $t = c\ln(\mathrm{vol}(V_G))$. In the YES case, by Theorem 2.8, ESTIMATE returns a value at most

$$\begin{aligned} s \cdot \left(1 - \frac{\varphi_{\mathrm{in}}^2}{4}\right)^{2t} &\le s \cdot \exp\left(-\frac{\varphi_{\mathrm{in}}^2 t}{2}\right) = s \cdot \exp\left(-\frac{\varphi_{\mathrm{in}}^2 c\ln(\mathrm{vol}(V_G))}{2}\right) \\ &= \frac{1600(k+1)^2 \cdot \ln(12(k+1)) \cdot \ln(\mathrm{vol}(V_G))}{\beta \cdot (1-\eta)} \times \mathrm{vol}(V_G)^{-c\frac{\varphi_{\mathrm{in}}^2}{2}} \\ &\le \frac{1}{2} \cdot \frac{8(k+1)\ln(12(k+1))}{\beta \cdot (1-\eta)} \times \mathrm{vol}(V_G)^{2-c\frac{\varphi_{\mathrm{in}}^2}{2}}. \end{aligned}$$

In the last inequality we use the fact that $k+1 \le \mathrm{vol}(V_G)$, and $|V_G|$ is large enough to insure that $200\ln(\mathrm{vol}(V_G)) \le \mathrm{vol}(V_G)$. In the NO case, by Theorem 2.9, with probability at least $\frac{2}{3}$, ESTIMATE returns a value at least

$$\begin{aligned} \frac{8(k+1)\ln(12(k+1))}{\beta \cdot (1-\eta) \cdot \mathrm{vol}(V_G)} \times (1 - 30\varphi_{out})^{2t} &\ge \frac{8(k+1)\ln(12(k+1))}{\beta \cdot (1-\eta) \cdot \mathrm{vol}(V_G)} \times \exp\left(-120\varphi_{\mathrm{out}} c\ln(\mathrm{vol}(V_G))\right) \\ &\ge \frac{1}{2} \cdot \frac{8(k+1)\ln(12(k+1))}{\beta \cdot (1-\eta)} \times \mathrm{vol}(V_G)^{-1-120c\varphi_{\mathrm{out}}}. \end{aligned}$$

Since $\varphi_{\mathrm{in}}^2 > 480\varphi_{\mathrm{out}}$, the value of $c = \frac{20}{\varphi_{\mathrm{in}}^2}$, chosen in PARTITIONTEST is such that $2 - c\frac{\varphi_{\mathrm{in}}^2}{2} < -1 - 120c\varphi_{\mathrm{out}}$. Therefore for $|V_G|$ large enough, the upper bound on the value returned by ESTIMATE in the YES case is less than $\mu_{\mathrm{thres}} = \frac{1}{2} \cdot \frac{8(k+1)\ln(12(k+1))}{\beta \cdot (1-\eta)} \times \mathrm{vol}(V_G)^{-1-120c\varphi_{\mathrm{out}}}$, which is less than the lower bound on the value returned by ESTIMATE in the NO case. □

Proof of Theorem 2.8 (the YES case)

The main result of this section is a proof of Theorem 2.8, restated below for convenience of the reader:

Theorem 2.8 (restated) *Let* $\varphi_{in} > 0$ *and integer* $k \ge 1$. *Then for every* (k, φ_{in})*-clusterable graph* $G = (V_G, E_G)$ *(see definition 2.2), with* $\min_{v \in V_G} deg(v) \ge 1$ *the following holds:*

$$\mathrm{ESTIMATE}(G, k, s, t, \eta) \le s \cdot \left(1 - \frac{\varphi_{in}^2}{4}\right)^{2t}.$$

Consider the YES case, where the vertices of G can be partitioned into h subsets with $C_1, \ldots, C_h$ for some $h \le k$, such that for each i, $\phi^G(C_i) \ge \varphi_{\mathrm{in}}$. We are interested in bounding $\mu_{k+1}((D^{-\frac{1}{2}}M^tS)^T(D^{-\frac{1}{2}}M^tS))$ from above.

Lemma 2.8. *Let* $\varphi_{in} > 0$, *integer* $k \ge 1$, *and* $G = (V_G, E_G)$ *be a* (k, φ_{in})*-clusterable graph (see* *definition 2.2), with* $\min_{v\in V_G} deg(v) \geq 1$. *Let L be its normalized Laplacian matrix, and M be the transition matrix of the associated random walk. Let S be a (multi)set of s vertices of G. Then*

$$\mu_{k+1}((D^{-\frac{1}{2}}M^tS)^T(D^{-\frac{1}{2}}M^tS)) \leq s\cdot\left(1-\frac{\lambda_{k+1}}{2}\right)^{2t},$$

where λ_{k+1} *is the* $(k+1)$*-st smallest eigenvalue of L.*

Proof. Recall from Section 2.2 that $M = D^{\frac{1}{2}}\overline{M}D^{-\frac{1}{2}}$, hence, $M^t = D^{\frac{1}{2}}\overline{M}^tD^{-\frac{1}{2}}$. Thus we can write

$$\begin{aligned}\mu_{k+1}((D^{-\frac{1}{2}}M^tS)^T(D^{-\frac{1}{2}}M^tS)) &= \mu_{k+1}((D^{-\frac{1}{2}}D^{\frac{1}{2}}\overline{M}D^{-\frac{1}{2}}S)^T(D^{-\frac{1}{2}}D^{\frac{1}{2}}\overline{M}D^{-\frac{1}{2}}S))\\ &= \mu_{k+1}(S^TD^{-\frac{1}{2}}\overline{M}^{2t}D^{-\frac{1}{2}}S).\end{aligned}$$

Recall from Section 2.2 that $1-\frac{\lambda_1}{2} \geq \cdots \geq 1-\frac{\lambda_n}{2}$ are the eigenvalues of $\overline{M}$, Σ is the diagonal matrix of these eigenvalues in descending order, and V is the matrix whose columns are orthonormal eigenvectors arranged in descending order of their eigenvalues. We have $\overline{M}^{2t} = V\Sigma^{2t}V^T$. Let $\Sigma_{1:k}$ be $n\times n$ diagonal matrix with first k entries $1-\frac{\lambda_1}{2} \geq \cdots \geq 1-\frac{\lambda_k}{2}$ and the rest zero and let $\Sigma_{k+1:n}$ denote $n\times n$ diagonal matrix with first k entries zero and rest $1-\frac{\lambda_{k+1}}{2} \geq \cdots \geq 1-\frac{\lambda_n}{2}$. We have $\Sigma^{2t} = \Sigma^{2t}_{1:k} + \Sigma^{2t}_{k+1:n}$, thus we get

$$\begin{aligned}\mu_{k+1}((D^{-\frac{1}{2}}M^tS)^T(D^{-\frac{1}{2}}M^tS)) &= \mu_{k+1}(S^TD^{-\frac{1}{2}}\overline{M}^{2t}D^{-\frac{1}{2}}S)\\ &= \mu_{k+1}(S^TD^{-\frac{1}{2}}(V\Sigma^{2t}V^T)D^{-\frac{1}{2}}S)\\ &= \mu_{k+1}(S^TD^{-\frac{1}{2}}V(\Sigma^{2t}_{1:k}+\Sigma^{2t}_{k+1:n})V^TD^{-\frac{1}{2}}S)\\ &\leq \mu_{k+1}(S^TD^{-\frac{1}{2}}V\Sigma^{2t}_{1:k}V^TD^{-\frac{1}{2}}S) + \mu_{\max}(S^TD^{-\frac{1}{2}}V\Sigma^{2t}_{k+1:n}V^TD^{-\frac{1}{2}}S)\end{aligned}$$

The last inequality follows from Lemma 2.3. Here $\mu_{k+1}(S^TD^{-\frac{1}{2}}V\Sigma^{2t}_{1:k}V^TD^{-\frac{1}{2}}S) = 0$, because the rank of $\Sigma^{2t}_{1:k}$ is k. We are left to bound $\mu_{\max}(S^TD^{-\frac{1}{2}}V\Sigma^{2t}_{k+1:n}V^TD^{-\frac{1}{2}}S)$. By Lemmas 2.2 and 2.1, we have,

$$\begin{aligned}&\mu_{\max}(S^TD^{-\frac{1}{2}}V\Sigma^{2t}_{k+1:n}V^TD^{-\frac{1}{2}}S)\\ &= \mu_{\max}(D^{-\frac{1}{2}}V\Sigma^{2t}_{k+1:n}V^TD^{-\frac{1}{2}}SS^T) && \text{(By Lemma 2.2)}\\ &\leq \mu_{\max}(D^{-\frac{1}{2}}V\Sigma^{2t}_{k+1:n}V^TD^{-\frac{1}{2}})\cdot\mu_{\max}(SS^T) && \text{(By Lemma 2.1)}\\ &= \mu_{\max}(V\Sigma^{2t}_{k+1:n}V^TD^{-1})\cdot\mu_{\max}(SS^T) && \text{(By Lemma 2.2)}\\ &\leq \mu_{\max}(V\Sigma^{2t}_{k+1:n}V^T)\cdot\mu_{\max}(SS^T)\cdot\mu_{\max}(D^{-1}) && \text{(By Lemma 2.1)}\\ &= \mu_{\max}(\Sigma^{2t}_{k+1:n}V^TV)\cdot\mu_{\max}(SS^T)\cdot\mu_{\max}(D^{-1}) && \text{(By Lemma 2.2)}\\ &= \mu_{\max}(\Sigma^{2t}_{k+1:n})\cdot\mu_{\max}(SS^T)\cdot\mu_{\max}(D^{-1}) && \text{(Since } V^TV = I\text{)}\end{aligned}$$

Next, observe that $SS^T \in N\times N$ is a diagonal matrix whose $(a,a)^{\text{th}}$ entry is the multiplicity of vertex a in S. Thus, $\mu_{\max}(SS^T)$ is the maximum multiplicity over all vertices, which is at most s. Also notice that $\mu_{\max}(D^{-1}) = \max_{v\in V_G} \frac{1}{\deg(v)} \leq 1$, and $\mu_{\max}(\Sigma^{2t}_{k+1:n}) = (1-\frac{\lambda_{k+1}}{2})^{2t}$. Thus we get,

$$\mu_{k+1}((D^{-\frac{1}{2}}M^tS)^T(D^{-\frac{1}{2}}M^tS)) \leq \mu_{\max}(S^TD^{-\frac{1}{2}}V\Sigma^{2t}_{k+1:n}V^TD^{-\frac{1}{2}}S) \leq s\cdot\left(1-\frac{\lambda_{k+1}}{2}\right)^{2t}.$$

□

Next, we bound λ_{k+1} from below. For this, we prove a lemma that can be seen as a strengthening of Lemma 5.2 of [CPS15]. Let us first recall Cheeger's inequality that we use later in the proof of Lemma 2.10.

Lemma 2.9 (Cheeger's inequality)**.** *For a general graph G, let L denote the normalized Laplacian of G, and λ_2 be the second smallest eigenvalue of L. Then*

$$\frac{\phi(G)^2}{2} \leq \lambda_2 \leq 2\phi(G).$$

Lemma 2.10. *Let G be any graph which is (k,φ_{in})-clusterable. Let L be its normalized Laplacian matrix, and λ_{k+1} be the $(k+1)$st smallest eigenvalue of L. Then $\lambda_{k+1} \geq \frac{\varphi^2_{in}}{2}$.*

Proof. Let $C_1,\ldots,C_h$ be a partition of V_G which achieves $\phi^G(C_i) \geq \varphi$ for all i, and $h \leq k$. Let G_{in} be the graph consisting of edges of G with endpoints in the same cluster C_i for some i. Let G_{out} be the graph consisting of edges of G with endpoints in different clusters. Let D, D_{in}, and D_{out} be the diagonal matrices of the degrees of the vertices in G, G_{in}, and G_{out} respectively, so that $D = D_{\text{in}} + D_{\text{out}}$. Let A, A_{in}, and A_{out} be the adjacency matrices of G, G_{in}, and G_{out} respectively, so that $A = A_{\text{in}} + A_{\text{out}}$. Recall that λ_{k+1} is the $(k+1)$st eigenvalue of the the normalized Laplacian L. Observe that,

$$L = I - \overline{A} = D^{-\frac{1}{2}}(D-A)D^{-\frac{1}{2}} = D^{-\frac{1}{2}}(D_{\text{in}}-A_{\text{in}})D^{-\frac{1}{2}} + D^{-\frac{1}{2}}(D_{\text{out}}-A_{\text{out}})D^{-\frac{1}{2}}$$

Let $\lambda^{\text{in}}_{k+1}$ be the $(k+1)$st smallest eigenvalue of $D^{-\frac{1}{2}}(D_{\text{in}}-A_{\text{in}})D^{-\frac{1}{2}}$ and λ^{out}_1 be the minimum eigenvalue of $D^{-\frac{1}{2}}(D_{\text{out}}-A_{\text{out}})D^{-\frac{1}{2}}$. Then by Lemma 2.3, $\lambda_{k+1} \geq \lambda^{\text{in}}_{k+1} + \lambda^{\text{out}}_1$. Observe that $\lambda^{\text{out}}_1 \geq 0$, since $D^{-\frac{1}{2}}(D_{\text{out}}-A_{\text{out}})D^{-\frac{1}{2}}$ is positive semi-definite. Therefore, it is sufficient to lower bound $\lambda^{\text{in}}_{k+1}$.

Let graph G'_{in} is obtained from G_{in} by increasing the degree of every $a \in V_G$ by $D(aa) - D_{\text{in}}(aa)$, by adding self-loops. Let A'_{in}, and L'_{in} be the adjacency matrix and the normalized Laplacian of G_{in} respectively. Observe that $D^{-\frac{1}{2}}(D_{\text{in}}-A_{\text{in}})D^{-\frac{1}{2}} = D^{-\frac{1}{2}}(D-A'_{\text{in}})D^{-\frac{1}{2}} = L'_{\text{in}}$.

Consider the graph G'_{in}. It is composed of h disconnected components, each of which has internal expansion φ_{in}. Thus, by applying Cheeger's inequality to each component, we get that, the second smallest eigenvalue of the normalized Laplacian of each component is at least $\frac{\varphi_{\text{in}}^2}{2}$. Now the set of eigenvalues of L'_{in} is the multi-union of the sets of eigenvalues of the components. Thus, we have $\lambda_1^{\text{in}} = \cdots = \lambda_h^{\text{in}} = 0$ and $\frac{\varphi_{\text{in}}^2}{2} \leq \lambda_{h+1}^{\text{in}} \leq \ldots \leq \lambda_{k+1}^{\text{in}}$. This implies $\lambda_{k+1} \geq \frac{\varphi_{\text{in}}^2}{2}$, as required. □

Proof of Theorem 2.8. Follows from Lemma 2.8 and Lemma 2.10. □

Proof of Theorem 2.9 (the NO case)

The main result of this section is a proof of Theorem 2.9, restated below for convenience of the reader:

Theorem 2.9 (restated) Let $\varphi_{\text{out}} > 0$, $\beta \in (0,1)$, and integer $k \geq 1$. Let

$$s = 1600(k+1)^2 \cdot \ln(12(k+1)) \cdot \ln(\text{vol}(V_G))/(\beta \cdot (1-\eta)).$$

Then for every $(k, \varphi_{\text{out}}, \beta)$-unclusterable graph $G = (V_G, E_G)$ (see definition 2.2), with $\min_{v \in V_G} \deg(v) \geq 1$, the following holds with probability at least $\frac{2}{3}$.

$$\textsc{Estimate}(G, k, s, t, \eta) \geq \frac{8(k+1)\ln(12(k+1))}{\beta \cdot (1-\eta) \cdot \text{vol}(V_G)} \times (1 - 30\varphi_{out})^{2t}.$$

Consider the NO case, where the vertex set of G contains $k+1$ subsets $C_1, \ldots, C_{k+1}$ of volume at least $\frac{\beta}{k+1}\text{vol}(V_G)$ each, such that for each i, $\phi_{V_G}^G(C_i) \leq \varphi_{\text{out}}$. We are interested in bounding the quantity $\mu_{k+1}((D^{-\frac{1}{2}} M^t S)^T (D^{-\frac{1}{2}} M^t S))$ from below. Let θ be a large enough absolute constant (say $\theta = 60$). Let h be the largest index such that $\lambda_h < \theta\varphi_{\text{out}}$.

Recall that $(v_1, \ldots, v_n)$ is an orthonormal basis of eigenvectors of L. $V \in \mathbb{R}^{V_G \times [n]}$ is the matrix whose columns are the orthonormal eigenvectors of L arranged in increasing order of eigenvalues. Let $P_h = V_{1:h} V_{1:h}^T$ and $P_h^{\perp} = V_{h+1:n} V_{h+1:n}^T$, so that for any vector $v \in \mathbb{R}^{V_G}$, $P_h v$ is the projection of v onto the span of $\{v_1, \ldots, v_h\}$, and $P_h^{\perp} v$ is its projection on the span of $\{v_{h+1}, \ldots, v_n\}$, that is, the orthogonal complement of the span of $\{v_1, \ldots, v_h\}$. Also, observe that $P_h + P_h^{\perp} = I$, $P_h^2 = P_h$, and $(P_h^{\perp})^2 = P_h^{\perp}$. Let $P = D^{-\frac{1}{2}} P_h$ and $P^{\perp} = D^{-\frac{1}{2}} P_h^{\perp}$. Let $U \in \mathbb{R}^{V_G \times [k+1]}$ be the matrix with orthonormal columns, where for $a \in V_G$, and $1 \leq i \leq k+1$, the (a,i)-th entry of U has $\sqrt{\frac{\deg(a)}{\text{vol}(C_i)}}$ if $a \in C_i$, and zero otherwise.

Lemma 2.11. *Let $G = (V_G, E_G)$ be a graph with $\min_{v \in V_G} deg(v) \geq 1$, and with normalized Laplacian L (Definition 2.6), and M be the transition matrix of the random walk associated with G (Definition 2.7). Let $C_1 \ldots, C_{k+1}$ be pairwise disjoint subsets of vertices of G. Let $U \in \mathbb{R}^{V_G \times [k+1]}$ be the matrix with orthonormal columns, where for $a \in V_G$, and $1 \leq i \leq k+1$, the (a,i)-th entry of U has $\sqrt{\frac{deg(a)}{vol(C_i)}}$ if $a \in C_i$, and zero otherwise. For $\theta > 0$ and $\varphi_{out} \geq 0$, let h be the largest index such that λ_h, the h^{th} smallest eigenvalue of L, is less than $\theta\varphi_{out}$. Let $P = D^{-\frac{1}{2}} P_h$. Let S be any* *multiset of vertices. (Recall our abuse of notation from Section 2.2.) Then*

$$\mu_{k+1}((D^{-\frac{1}{2}}M^tS)^T(D^{-\frac{1}{2}}M^tS)) \geq \left(1-\frac{\theta\varphi_{out}}{2}\right)^{2t}\cdot \min_{z\in\mathbb{R}^{k+1},\,\|z\|_2=1}\|S^TPUz\|_2^2.$$

Proof. We can write $\mu_{k+1}((D^{-\frac{1}{2}}M^tS)^T(D^{-\frac{1}{2}}M^tS))$ as

$$\begin{aligned}\mu_{k+1}((D^{-\frac{1}{2}}M^tS)^T(D^{-\frac{1}{2}}M^tS)) &= \mu_{k+1}((M^tS)^TD^{-1}(M^tS))\\ &\geq \mu_{k+1}((M^tS)^TD^{-\frac{1}{2}}V_{1:h}V_{1:h}^TD^{-\frac{1}{2}}(M^tS)) && \text{By Lemma 2.4}\\ &= \mu_{k+1}((M^tS)^TD^{-\frac{1}{2}}V_{1:h}V_{1:h}^TV_{1:h}V_{1:h}^TD^{-\frac{1}{2}}(M^tS)) && \text{Since } V_{1:h}^TV_{1:h}=I\\ &= \mu_{k+1}(S^T{M^t}^TPP^TM^tS).\end{aligned}$$

Recall from Section 2.2 that $v_i^TD^{-\frac{1}{2}}$ is a left eigenvector of M, and $D^{-\frac{1}{2}}v_i$ is a right eigenvector of M^T with eigenvalue $1-\frac{\lambda_i}{2}$. Thus we can write $V_{1:h}^TD^{-\frac{1}{2}}M^t=\Sigma_{1:h}^tV_{1:h}^TD^{-\frac{1}{2}}$ and ${M^t}^TD^{-\frac{1}{2}}V_{1:h}=D^{-\frac{1}{2}}V_{1:h}\Sigma_{1:h}^t$, where $\Sigma_{1:h}^t$ is a $h\times h$ diagonal matrix with entries $(1-\frac{\lambda_1}{2})^t,\ldots,(1-\frac{\lambda_h}{2})^t$. Observe that

$$(M^t)^TP=(M^t)^TD^{-\frac{1}{2}}V_{1:h}V_{1:h}^T=D^{-\frac{1}{2}}V_{1:h}\Sigma_{1:h}^tV_{1:h}^T$$

Thus we have,

$$\begin{aligned}\mu_{k+1}((D^{-\frac{1}{2}}M^tS)^T(D^{-\frac{1}{2}}M^tS)) &\geq \mu_{k+1}(S^T{M^t}^TPP^TM^tS)\\ &= \mu_{k+1}(S^TD^{-\frac{1}{2}}V_{1:h}\Sigma_{1:h}^tV_{1:h}^TV_{1:h}\Sigma_{1:h}^tV_{1:h}^TD^{-\frac{1}{2}}S)\\ &= \max_U\{\quad \min_y\{\|V_{1:h}\Sigma_{1:h}^tV_{1:h}^TD^{-\frac{1}{2}}Sy\|_2^2\quad |y\in U,\|y\|_2=1\}\quad |\dim(U)=k+1\},\end{aligned}$$

where the last equality follows from Courant-Fischer min-max principle (Lemma 2.5). Observe that $\Sigma_{1:h}^t$ is a $h\times h$ diagonal matrix with entries $(1-\frac{\lambda_1}{2})^t,\ldots,(1-\frac{\lambda_h}{2})^t$, hence

$$\|V_{1:h}\Sigma_{1:h}^tV_{1:h}^TD^{-\frac{1}{2}}Sy\|_2^2\geq\left(1-\frac{\lambda_h}{2}\right)^{2t}\cdot\|V_{1:h}V_{1:h}^TD^{-\frac{1}{2}}Sy\|_2^2=\left(1-\frac{\lambda_h}{2}\right)^{2t}\cdot\|P^TSy\|_2^2$$

Notice that by Courant-Fischer min-max principle (Lemma 2.5) we have

$$\mu_{k+1}(S^TPP^TS)=\max_U\{\quad\min_y\{\|P^TSy\|_2^2\quad|y\in U,\|y\|_2=1\}\quad|\dim(U)=k+1\}.$$

Let U^* be the subspace with $\dim(U^*)=k+1$ which maximizes

$$\min_y\{\|P^TSy\|_2^2\quad|y\in U,\|y\|_2=1\}$$

Thus we get,

$$
\begin{aligned}
&\mu_{k+1}((D^{-\frac{1}{2}}M^t S)^T (D^{-\frac{1}{2}}M^t S)) \\
&= \max_U \{ \quad \min_y \{ \| V_{1:h}\Sigma_{1:h}^t V_{1:h}^T D^{-\frac{1}{2}} Sy\|_2^2 \quad | y \in U, \|y\|_2 = 1\} \quad | \dim(U) = k+1\} \\
&\geq \min_y \{ \| V_{1:h}\Sigma_{1:h}^t V_{1:h}^T D^{-\frac{1}{2}} Sy\|_2^2 \quad | y \in U^*, \|y\|_2 = 1\} \\
&\geq \min_y \left\{ \left(1 - \frac{\lambda_h}{2}\right)^{2t} \cdot \|P^T Sy\|_2^2 \quad | y \in U^*, \|y\|_2 = 1 \right\} \\
&= \left(1 - \frac{\lambda_h}{2}\right)^{2t} \cdot \min_y \{ \|P^T Sy\|_2^2 \quad | y \in U^*, \|y\|_2 = 1\} \\
&= \left(1 - \frac{\lambda_h}{2}\right)^{2t} \cdot \mu_{k+1}(S^T P P^T S).
\end{aligned}
$$

Therefore we have

$$
\begin{aligned}
&\mu_{k+1}\left((D^{-\frac{1}{2}}M^t S)^T (D^{-\frac{1}{2}}M^t S)\right) \\
&\geq \left(1 - \frac{\lambda_h}{2}\right)^{2t} \cdot \mu_{k+1}(S^T P P^T S) \\
&\geq \left(1 - \frac{\lambda_h}{2}\right)^{2t} \cdot \mu_{k+1}(S^T P U U^T P^T S) && \text{By Lemma 2.4} \\
&= \left(1 - \frac{\lambda_h}{2}\right)^{2t} \cdot \mu_{k+1}(U^T P^T S S^T P U) && \text{By Lemma 2.2} \\
&= \left(1 - \frac{\lambda_h}{2}\right)^{2t} \cdot \mu_{\min}(U^T P^T S S^T P U) && \text{Since } U^T P^T S S^T P U \text{ is } k+1 \times k+1 \text{ matrix} \\
&\geq \left(1 - \frac{\lambda_h}{2}\right)^{2t} \cdot \min_{z \in \mathbb{R}^{k+1},\ \|z\|_2 = 1} \|S^T P U z\|_2^2 \\
&\geq \left(1 - \frac{\theta \varphi_{\text{out}}}{2}\right)^{2t} \cdot \min_{z \in \mathbb{R}^{k+1},\ \|z\|_2 = 1} \|S^T P U z\|_2^2.
\end{aligned}
$$

□

Our goal is to prove that by selecting a random (multi)set S of vertices of a "reasonable" size, with at least a constant probability (say 2/3), for all $z \in \mathbb{R}^{k+1}$ with $\|z\|_2 = 1$, we have $\left\|S^T P U z\right\|_2^2$ is "large".

Lemma 2.12. *Let $G = (V_G, E_G)$ be a graph with $\min_{v \in V_G} deg(v) \geq 1$, and with normalized Laplacian L (Definition 2.6). Let $C_1 \ldots, C_{k+1}$ be pairwise disjoint subsets of vertices of G such that $\phi_{V_G}^G(C_i) \leq \varphi_{out}$ for all i. Let $U \in \mathbb{R}^{V_G \times [k+1]}$ be the matrix with orthonormal columns, where for $a \in V_G$, and $1 \leq i \leq k+1$, the (a,i)-th entry of U has $\sqrt{\frac{deg(a)}{vol(C_i)}}$ if $a \in C_i$, and zero otherwise. Then for every $z \in \mathbb{R}^{k+1}$ with $\|z\|_2^2 = 1$,*

$$z^T U^T L U z \leq 2\varphi_{out}$$

.

Proof. We have, $z^T U^T L U z = z^T U^T U z - z^T U^T \overline{A} U z = 1 - z^T U^T \overline{A} U z$, where $\overline{A}$ is the normalized adjacency matrix of G, since $U^T U = I$. We will prove that every eigenvalue of $U^T \overline{A} U$ lies in $[1-2\varphi_{\text{out}}, 1]$, and this implies the claim.

Let m_{ij} denote the number of edges between C_i and C_j, and m'_i denote the number of edges between C_i and $V_G \setminus \bigcup_{j=1}^{k+1} C_j$. Then observe that

$$(U^T \overline{A} U)_{ij} = u_i^T A u_j = \frac{m_{ij}}{\sqrt{\text{vol}(C_i) \cdot \text{vol}(C_j)}}.$$

Thus, $U^T \overline{A} U = W^{-1/2} H W^{-1/2}$, where $W = \text{diag}(\text{vol}(C_1), \ldots, \text{vol}(C_{k+1}))$, and H is given by $H_{ij} = m_{ij}$. By Lemma 2.2, the eigenvalues of $W^{-1/2} H W^{-1/2}$ are same as the eigenvalues of $W^{-1} H$. Therefore, it is sufficient to prove that the eigenvalues of $W^{-1} H$ lie in $[1-2\varphi_{\text{out}}, 1]$.

We know that for all i, $\phi^G_{V_G}(C_i) \leq \varphi_{\text{out}}$, and thus, for all i, $m'_i + \sum_{j \neq i} m_{ij} \leq \varphi_{\text{out}} \cdot \text{vol}(C_i)$, and $m_{ii} \geq (1-\varphi_{\text{out}})\text{vol}(C_i)$. Therefore $W^{-1} H$ is the $(k+1) \times (k+1)$ matrix such that for all i,

$$(W^{-1} H)_{ii} = \frac{m_{ii}}{\text{vol}(C_i)} \geq 1 - \varphi_{\text{out}}$$

and

$$\sum_{j \neq i} (W^{-1} H)_{ij} = \frac{1}{\text{vol}(C_i)} \sum_{j \neq i} m_{ij} \leq \varphi_{\text{out}}.$$

Thus, for every i,

$$(W^{-1} H)_{ii} + \sum_{j \neq i} (W^{-1} H)_{ij} = \frac{1}{\text{vol}(C_i)} \sum_{j=1}^{k+1} m_{ij} \leq 1, \tag{2.3}$$

and

$$(W^{-1} H)_{ii} - \sum_{j \neq i} (W^{-1} H)_{ij} \geq 1 - 2\varphi_{\text{out}}. \tag{2.4}$$

From (2.3) and (2.4), and by using the Gershgorin circle theorem (Lemma 2.6), we conclude that every eigenvalue of $W^{-1} H$ lies within $[1-2\varphi_{\text{out}}, 1]$, as required. □

Lemma 2.13. *Let $G = (V_G, E_G)$ be a graph with $\min_{v \in V_G} deg(v) \geq 1$, and with normalized Laplacian L (Definition 2.6). Let $C_1 \ldots, C_{k+1}$ be pairwise disjoint subsets of vertices of G such that $\phi^G_{V_G}(C_i) \leq \varphi_{out}$ for all i. Let $U \in \mathbb{R}^{V_G \times [k+1]}$ be the matrix with orthonormal columns, where for $a \in V_G$, and $1 \leq i \leq k+1$, the (a,i)-th entry of U has $\sqrt{\frac{deg(a)}{vol(C_i)}}$ if $a \in C_i$, and zero otherwise. Let $z \in \mathbb{R}^{k+1}$ with $\|z\|_2^2 = 1$. For a constant $\theta > 0$, let h be the largest index such that λ_h, the h-th smallest eigenvalue of L, is less than $\theta\varphi_{out}$. Then*

$$\|P_h^{\perp} U z\|_2^2 \leq \frac{2}{\theta}$$

.

Proof. Recall that $0 = \lambda_1 \le \cdots \le \lambda_n$ are the eigenvalues of L and $v_1, \ldots, v_n$ are the corresponding orthonormal eigenvectors forming a basis of $\mathbb{R}^{V_G}$. Write $Uz \in \mathbb{R}^{V_G}$ in the eignebasis as $Uz = \sum_{i=1}^n \alpha_i v_i$. Then we have,

$$(Uz)^T LUz = \left(\sum_{i=1}^n \alpha_i v_i^T\right) L \left(\sum_{i=1}^n \alpha_i v_i\right) = \sum_{i=1}^n \lambda_i \alpha_i^2 \ge \sum_{i=h+1}^n \lambda_i \alpha_i^2 \ge \theta \varphi_{\text{out}} \sum_{i=h+1}^n \alpha_i^2. \tag{2.5}$$

On the other hand, by Lemma 2.12, we have

$$z^T U^T LUz \le 2\varphi_{\text{out}}. \tag{2.6}$$

Putting (2.5) and (2.6) together, we get $\theta\varphi_{\text{out}} \sum_{i=h+1}^n \alpha_i^2 \le 2\varphi_{\text{out}}$, and thus $\sum_{i=h+1}^n \alpha_i^2 \le \frac{2}{\theta}$. Recall that $P_h^\perp = V_{h+1:n}^T V_{h+1:n}$, and therefore,

$$\left\|P_h^\perp Uz\right\|_2^2 = \left\|\sum_{i=1}^n \alpha_i P_h^\perp v_i\right\|_2^2 = \left\|\sum_{i=h+1}^n \alpha_i v_i\right\|_2^2 = \sum_{i=h+1}^n \alpha_i^2 \le \frac{2}{\theta}.$$

□

The next two lemmas concern random samples of vertices S', and prove, for any fixed z, a lower bound on $\left\|S'^T PUz\right\|_2$ as a function of the size of S'.

Lemma 2.14. *Let $C_1, \ldots, C_{k+1}$ be subsets of some universe V_G, such that for all j, $vol(C_j) \ge \frac{\beta}{k+1} vol(V_G)$ for some $\beta > 0$. Let $\eta \in (0,1)$, and*

$$s' = \frac{200(k+1)\ln(12(k+1))}{\beta \cdot (1-\eta)}.$$

Let S' be a multiset of s' independent random vertices in V_G, sampled from distribution $\mathscr{P}$ over V_G such that for all $v \in V_G$, $\left|\mathscr{P}(v) - \frac{\deg(v)}{vol(G)}\right| \le \eta \cdot \frac{\deg(v)}{vol(G)}$. Then with probability at least $\frac{11}{12}$, for every $1 \le j \le k+1$,

$$|S' \cap C_j| \ge \frac{9}{10} \cdot \frac{vol(C_j)}{vol(V_G)} s'(1-\eta).$$

Proof. For $v \in V_G$, and $1 \le r \le s'$, let X_v^r be a random variable which is 1 if the r-th sampled vertex is v, and 0 otherwise. Thus $\mathbb{E}[X_v^r] = \mathscr{P}(v) \ge (1-\eta)\frac{\deg(v)}{\text{vol}(V_G)}$. Observe that $|S' \cap C_j|$ is a random variable defined as $\sum_{r=1}^{s'} \sum_{v \in C_j} X_v^r$, where its expectation is given by

$$\mathbb{E}[|S' \cap C_j|] = \sum_{r=1}^{s'} \sum_{v \in C_j} \mathbb{E}[X_v^r] \ge s'(1-\eta)\frac{\text{vol}(C_j)}{\text{vol}(V_G)} \ge s'(1-\eta)\frac{\beta}{k+1}.$$

Notice that the random variables X_v^r are negatively associated, since for each r, $\sum_{v \in V_G} X_v^r = 1$.

Therefore, by Chernoff bound,

$$\Pr\left[|S'\cap C_j| < \frac{9s'(1-\eta)}{10}\cdot\frac{\mathrm{vol}(C_j)}{\mathrm{vol}(V_G)}\right] \le \exp\left(-\frac{s'(1-\eta)}{200}\cdot\frac{\beta}{k+1}\right).$$

By union bound,

$$\Pr\left[\exists j\colon |S'\cap C_j| < \frac{9s'(1-\eta)}{10}\cdot\frac{\mathrm{vol}(C_j)}{\mathrm{vol}(V_G)}\right] \le (k+1)\cdot\exp\left(-\frac{s'(1-\eta)}{200}\cdot\frac{\beta}{k+1}\right) \le \frac{1}{12},$$

by our choice of s'. □

Lemma 2.15. *Let $G=(V_G,E_G)$ be a graph with $\min_{v\in V_G} deg(v)\ge 1$, and with normalized Laplacian L (Definition 2.6). Let $C_1\dots,C_{k+1}$ be pairwise disjoint subsets of vertices of G of volume at least $\frac{\beta}{k+1} vol(V_G)$ each, such that $\phi^G_{V_G}(C_i)\le\varphi_{out}$ for all i. Let $U\in\mathbb{R}^{V_G\times[k+1]}$ be the matrix with orthonormal columns, where for $a\in V_G$, and $1\le i\le k+1$, the (a,i)-th entry of U has $\sqrt{\frac{deg(a)}{vol(C_i)}}$ if $a\in C_i$, and zero otherwise. Let $z\in\mathbb{R}^{k+1}$ with $\|z\|_2^2=1$. For $\theta=60$, let h be the largest index such that λ_h, the h^{th} smallest eigenvalue of L, is less than $\theta\varphi_{out}$. Let $P=D^{-\frac{1}{2}}P_h$ and $P^\perp=D^{-\frac{1}{2}}P_h^\perp$. Let $0<\eta\le\frac{1}{2}$, and*

$$s'=\frac{200(k+1)\ln(12(k+1))}{\beta\cdot(1-\eta)}.$$

Let S' be a multiset of s' independent random vertices in V_G, sampled from distribution $\mathcal{P}$ over V_G such that for all $v\in V_G$, $\left|\mathcal{P}(v)-\frac{\deg(v)}{vol(G)}\right|\le\eta\cdot\frac{\deg(v)}{vol(G)}$. Then $\left\|S'^TPUz\right\|_2\ge\frac{1}{2}\sqrt{\frac{s'}{vol(V_G)}}$, with probability at least $\frac{7}{12}$.

Proof. By triangle inequality $\|S'^TPUz\|_2\ge\left\|S'^TD^{-\frac{1}{2}}Uz\right\|_2-\|S'^TP^\perp Uz\|_2$. Observe that the vector $D^{-\frac{1}{2}}Uz$ takes a uniform value $\mu_j=\frac{z_j}{\sqrt{\mathrm{vol}(C_j)}}$ on each set C_j. By Lemma 2.14, with probability at least $\frac{11}{12}$, we have $|S'\cap C_j|\ge\frac{9s'(1-\eta)}{10}\cdot\frac{\mathrm{vol}(C_j)}{\mathrm{vol}(V_G)}$, for all j. In this event, we have,

$$\begin{aligned}\left\|S'^TD^{-\frac{1}{2}}Uz\right\|_2^2 &\ge \sum_{j=1}^{k+1}\frac{9s'(1-\eta)}{10}\cdot\frac{\mathrm{vol}(C_j)}{\mathrm{vol}(V_G)}\cdot\mu_j^2\\ &=\frac{9}{10}\cdot\frac{s'(1-\eta)}{\mathrm{vol}(V_G)}\sum_{j=1}^{k+1}\mathrm{vol}(C_j)\cdot\frac{z_j^2}{\mathrm{vol}(C_j)}\\ &=\frac{9}{10}\cdot\frac{s'(1-\eta)}{\mathrm{vol}(V_G)}\cdot\|z\|_2^2\\ &=\frac{9}{10}\cdot\frac{s'(1-\eta)}{\mathrm{vol}(V_G)},\end{aligned}\tag{2.7}$$

where the last equality follows because z is a unit vector.

Let $y=P_h^\perp Uz$. By Lemma 2.13, we have $\|y\|_2^2\le\frac{2}{\theta}$. Note that $P^\perp Uz=D^{-\frac{1}{2}}P_h^\perp Uz=D^{-\frac{1}{2}}y$. Thus we have

$$\begin{aligned}\mathbb{E}_{S'}\left[\left\|S'^T P^{\perp} U z\right\|_2^2\right] &= \mathbb{E}_{S'}\left[\left\|S'^T D^{-\frac{1}{2}} P_h^{\perp} U z\right\|_2^2\right]\\ &= s' \cdot \sum_{v \in V_G} \mathscr{P}(v) \cdot \frac{y(v)^2}{\deg(v)}\\ &\leq s' \cdot \sum_{v \in V_G} (1+\eta) \cdot \frac{\deg(v)}{\mathrm{vol}(V_G)} \cdot \frac{y(v)^2}{\deg(v)}\\ &= \frac{s'(1+\eta)}{\mathrm{vol}(V_G)} \|y\|_2^2\\ &\leq \frac{2}{\theta} \cdot \frac{s'(1+\eta)}{\mathrm{vol}(V_G)}.\end{aligned}$$

Thus, by Markov's inequality, with probability at least $\frac{2}{3}$,

$$\left\|S'^T P^{\perp} U z\right\|_2^2 \leq 3 \cdot \mathbb{E}[\left\|S'^T P^{\perp} U z\right\|_2^2] \leq \frac{6}{\theta} \cdot \frac{s'}{\mathrm{vol}(V_G)} = \frac{1}{10} \cdot \frac{s'(1+\eta)}{\mathrm{vol}(V_G)}, \tag{2.8}$$

where we get the last equality by recalling that $\theta = 60$. Putting (2.7) and (2.8) together, we get with probability at least $1 - \frac{1}{3} - \frac{1}{12} = \frac{7}{12}$,

$$\left\|S'^T P U z\right\|_2 \geq \left\|S'^T D^{-\frac{1}{2}} U z\right\|_2 - \left\|S'^T P^{\perp} U z\right\|_2 \geq \left(\sqrt{\frac{9(1-\eta)}{10}} - \sqrt{\frac{(1+\eta)}{10}}\right)\sqrt{\frac{s'}{\mathrm{vol}(V_G)}} \geq \frac{1}{4}\sqrt{\frac{s'}{\mathrm{vol}(V_G)}}.$$

□

The above lemma gives a lower bound on $\left\|S'^T P U z\right\|_2$ which holds with a constant probability. We next show how we can trade off the lower bound to get a guarantee that holds with probability arbitrarily close to one.

Lemma 2.16. *Let $G = (V_G, E_G)$ be a graph with $\min_{v \in V_G} deg(v) \geq 1$, and with normalized Laplacian L (Definition 2.6). Let $C_1 \ldots, C_{k+1}$ be pairwise disjoint subsets of vertices of G of volume at least $\frac{\beta}{k+1} vol(V_G)$ each, such that $\phi^G_{V_G}(C_i) \leq \varphi_{out}$ for all i. Let $U \in \mathbb{R}^{V_G \times [k+1]}$ be the matrix with orthonormal columns, where for $a \in V_G$, and $1 \leq i \leq k+1$, the (a,i)-th entry of U has $\sqrt{\frac{deg(a)}{vol(C_i)}}$ if $a \in C_i$, and zero otherwise. Let $z \in \mathbb{R}^{k+1}$ with $\|z\|_2^2 = 1$. For $\theta = 60$, let h be the largest index such that λ_h, the h^{th} smallest eigenvalue of L, is less than $\theta\varphi_{out}$. Let $P = D^{-\frac{1}{2}} P_h$ and $P^{\perp} = D^{-\frac{1}{2}} P_h^{\perp}$. Let $0 < \eta \leq \frac{1}{2}$, and*

$$s = \frac{1600(k+1)^2 \cdot \ln(12(k+1)) \cdot \ln(vol(V_G))}{\beta \cdot (1-\eta)}.$$

Let S' be a multiset of s' independent random vertices in V_G, sampled from distribution $\mathscr{P}$ over V_G such that for all $v \in V_G$, $\left|\mathscr{P}(v) - \frac{\deg(v)}{vol(G)}\right| \leq \eta \cdot \frac{\deg(v)}{vol(G)}$. Then for $\tau = 8(k+1)\ln vol(V_G)$, we have $\left\|S^T P U z\right\|_2 \geq \frac{1}{4}\sqrt{\frac{s}{\tau \cdot vol(V_G)}}$, with probability at least $1 - \left(\frac{5}{12}\right)^{\tau}$.

Proof. We represent S as a multi-union $S = S_1 \cup S_2 \cup \dots S_\tau$ of independently drawn sets containing $s' = \frac{s}{\tau}$ independent samples each, where

$$s' = \frac{s}{\tau} = \frac{200(k+1)\ln(12(k+1))}{\beta \cdot (1-\eta)}.$$

Using Lemma 2.15, for any i, $\Pr\left[\left\|S_i^T PUz\right\|_2 \geq \frac{1}{4}\sqrt{\frac{s}{\tau \cdot \mathrm{vol}(V_G)}}\right] \geq \frac{7}{12}$. Since the S_i's are independent sets of samples, $\Pr\left[\exists i \ \left\|S_i^T PUz\right\|_2 \geq \frac{1}{4}\sqrt{\frac{s}{\tau \cdot \mathrm{vol}(V_G)}}\right] \geq 1 - (\frac{5}{12})^\tau$. Therefore with probability at least $1 - \left(\frac{5}{12}\right)^\tau$, we have

$$\left\|S^T PUz\right\|_2^2 = \sum_i \left\|S_i^T PUz\right\|_2^2 \geq \left(\frac{1}{4}\sqrt{\frac{s}{\tau \cdot \mathrm{vol}(V_G)}}\right)^2 = \frac{s}{16\tau \mathrm{vol}(V_G)}$$

Thus, $\Pr\left[\left\|S^T PUz\right\|_2 \geq \frac{1}{4}\sqrt{\frac{s}{\tau \cdot \mathrm{vol}(V_G)}}\right] \geq 1 - (\frac{5}{12})^\tau$. □

In the next lemma, we switch the order of quantification and prove that with a constant probability, a random S achieves a large value for $\left\|S^T PUz\right\|_2$ for all z of unit norm simultaneously, with constant probability.

Lemma 2.17. *Let $G = (V_G, E_G)$ be a graph with $\min_{v \in V_G} deg(v) \geq 1$, and with normalized Laplacian L (Definition 2.6). Let $C_1 \dots, C_{k+1}$ be pairwise disjoint subsets of vertices of G of volume at least $\frac{\beta}{k+1} vol(V_G)$ each, such that $\phi^G_{V_G}(C_i) \leq \varphi_{out}$ for all i. Let $U \in \mathbb{R}^{V_G \times [k+1]}$ be the matrix with orthonormal columns, where for $a \in V_G$, and $1 \leq i \leq k+1$, the (a,i)-th entry of U has $\sqrt{\frac{deg(a)}{vol(C_i)}}$ if $a \in C_i$, and zero otherwise. Let $z \in \mathbb{R}^{k+1}$ with $\|z\|_2^2 = 1$. For $\theta = 60$, let h be the largest index such that λ_h, the h^{th} smallest eigenvalue of L, is less than $\theta\varphi_{out}$. Let $P = D^{-\frac{1}{2}} P_h$ and $P^\perp = D^{-\frac{1}{2}} P_h^\perp$. Let $0 < \eta \leq \frac{1}{2}$, and*

$$s = \frac{1600(k+1)^2 \cdot \ln(12(k+1)) \cdot \ln(vol(V_G))}{\beta \cdot (1-\eta)}.$$

Let S' be a multiset of s' independent random vertices in V_G, sampled from distribution $\mathcal{P}$ over V_G such that for all $v \in V_G$, $\left|\mathcal{P}(v) - \frac{\deg(v)}{vol(G)}\right| \leq \eta \cdot \frac{\deg(v)}{vol(G)}$. Then with probability at least $\frac{2}{3}$, for all $z \in \mathbb{R}^{k+1}$ with $\|z\|_2^2 = 1$, we have $\left\|S^T PUz\right\|_2 \geq \frac{1}{5}\sqrt{\frac{s}{\tau \cdot vol(V_G)}}$, where $\tau = 8(k+1)\ln vol(V_G)$.

Proof. Let $\mathcal{N}$ be a $k+1$ dimensional δ-net of size $(\frac{4}{\delta})^{k+1}$ on the Euclidean sphere of radius 1, for some small enough δ. Let us first explain how to construct such δ-net. Pick y_1 of unit norm in $\mathbb{R}^{k+1}$, and then for every $t \geq 2$ pick y_t of unit norm such that $\left\|y_t - y_j\right\|_2 \geq \delta$ for all $j = 1, \dots, t-1$, until no such y can be picked. Note that balls of radius $\frac{\delta}{2}$ centered at the y_t's are disjoint, and their union belongs to the ball of radius $1 + \frac{\delta}{2}$ centered at zero. Thus $|\mathcal{N}| \leq \frac{(1+\frac{\delta}{2})^{k+1}}{(\frac{\delta}{2})^{k+1}} \leq (\frac{4}{\delta})^{k+1}$.

For $z \in \mathbb{R}^{k+1}$ with $\|z\|_2 = 1$, let $\mathcal{N}(z)$, be the closest point to z from $\mathcal{N}$. Using Lemma 2.16, by union bound over all points in $\mathcal{N}$ we have that with probability at least $1-(\frac{5}{12})^{\tau}(\frac{4}{\delta})^{k+1}$, for all $y \in \mathcal{N}$, $\|S^T PUy\|_2 \geq \frac{1}{4}\sqrt{\frac{s}{\tau \cdot \mathrm{vol}(V_G)}}$. Therefore, with probability at least $1-(\frac{5}{12})^{\tau}(\frac{4}{\delta})^{k+1}$, we have for every $z \in \mathbb{R}^{k+1}$ with $\|z\|_2 = 1$, $\|S^T PU\mathcal{N}(z)\|_2 \geq \frac{1}{4}\sqrt{\frac{s}{\tau \cdot \mathrm{vol}(V_G)}}$.

Observe that

$$\begin{aligned}
\|S^T PU(z-\mathcal{N}(z))\|_2^2 &= \|S^T PU\|_2^2 \cdot \|(z-\mathcal{N}(z))\|_2^2 \\
&\leq \mu_{\max}(U^T P^T SS^T PU)\cdot\delta \\
&\leq \mu_{\max}(UU^T)\cdot\mu_{\max}(PP^T)\cdot\mu_{\max}(SS^T)\cdot\delta && \text{(By Lemma 2.2, and Lemma 2.1)} \\
&\leq \mu_{\max}(PP^T)\cdot\mu_{\max}(SS^T)\cdot\delta && \text{(Since } UU^T = I) \\
&\leq \mu_{\max}(V_h V_h^T)\cdot\mu_{\max}(D^{-1})\cdot\mu_{\max}(SS^T)\cdot\delta && \text{(Since } PP^T = D^{-\frac{1}{2}} V_h V_h^T D^{-\frac{1}{2}})
\end{aligned}$$

Next, observe that $SS^T \in N \times N$ is a diagonal matrix whose $(a,a)^{\text{th}}$ entry is the multiplicity of vertex a in S. Thus, $\mu_{\max}(SS^T)$ is the maximum multiplicity over all vertices, which is at most s. Also notice that $\mu_{\max}(D^{-1}) = \max_{v \in V_G} \frac{1}{\deg(v)} \leq 1$, and $\mu_{\max}(V_h V_h^T) = 1$, since P_h is a projection matrix. Thus we get

$$\|S^T PU(z-\mathcal{N}(z))\|_2^2 \leq s\cdot\delta$$

Therefore, with probability at least $1-(\frac{5}{12})^{\tau}(\frac{4}{\delta})^{k+1}$, for every $z \in \mathbb{R}^h$ with $\|z\|_2 = 1$, we have

$$\|S^T PUz\|_2 \geq \|S^T PU\mathcal{N}(z)\|_2 - \|S^T PU(z-\mathcal{N}(z))\|_2 \geq \frac{1}{4}\sqrt{\frac{s}{\tau\cdot\mathrm{vol}(V_G)}} - s\cdot\delta.$$

By setting $\delta = \frac{1}{20s}\sqrt{\frac{s}{\tau\cdot\mathrm{vol}(V_G)}}$, we get $\|S^T PUz\|_2 \geq \frac{1}{5}\sqrt{\frac{s}{\tau\cdot\mathrm{vol}(V_G)}}$ with probability at least

$$1-\left(\frac{5}{12}\right)^{\tau}\left(\frac{4}{\delta}\right)^{k+1} = 1-\left(\frac{5}{12}\right)^{\tau}\left(80\sqrt{s\cdot\tau\cdot\mathrm{vol}(V_G)}\right)^{k+1}.$$

Observe that $\tau = 8(k+1)\ln\mathrm{vol}(V_G)$ is large enough to ensure that the above probability is at least $\frac{2}{3}$. □

Proof of Theorem 2.9. Follows from Lemma 2.11 and Lemma 2.17. □

2.3.2 Lifting the oracle assumption

The goal of this section is to show how we can remove the oracle assumption that we made in Section 2.3.1, and get an algorithm for the **PartitionTesting** problem that fits into the query complexity model, defined in Section 2.1.1, that only allows (uniformly random) vertex, degree, and neighbor queries. This will then establish Theorem 2.7. The algorithm is presented as a main procedure PARTITIONTESTWITHOUTORACLE (Algorithm 4) that calls the subroutine ESTIMATEWITHOUTORACLE (Algorithm 3). These two procedures can be seen as a analogs of the procedures PARTITIONTEST (Algorithm 2) and ESTIMATE (Algorithm 1) respectively, from

Section 2.3.1.

Algorithm 3 ESTIMATEWITHOUTORACLE$(G, k, s, t, \sigma, R, \eta)$

```
Sample $s$ vertices from $N$ independently and with probability proportional to the degree of the vertices at random with replacement using **sampler**$(G, \eta)$. Let $S$ be the multiset of sampled vertices.
$r = 192s\sqrt{\text{vol}(V_G)}$.
for Each sample $a \in S$ do
    if $\ell_2^2$-norm tester$(G, a, \sigma, r)$ rejects then return $\infty$.    ▷ High collision probability
for Each sample $a \in S$ do
    Run $2R$ random walks of length $t$ starting from $a$. Let $\mathbf{q}_a$ and $\mathbf{q}'_a$ be the empirical distribution of running $R$ random walks started at $a$.
Let $Q$ and $Q'$ be matrices whose columns are $\{D^{-\frac{1}{2}}\mathbf{q}_a : a \in S\}$ and $\{D^{-\frac{1}{2}}\mathbf{q}'_a : a \in S\}$ respectively.
Let $\mathcal{G} := \frac{1}{2} \cdot \left(Q^T Q' + Q'^T Q\right)$
Return $\mu_{k+1}(\mathcal{G})$.
```

Algorithm 4 PARTITIONTESTWITHOUTORACLE$(G, k, \varphi_{\text{in}}, \varphi_{\text{out}}, \beta)$ ▷ Need: $\varphi_{\text{in}}^2 > 480\varphi_{\text{out}}$

```
$\eta := 0.5$
$s := 1600(k+1)^2 \cdot \ln(12(k+1)) \cdot \ln(\text{vol}(V_G)/(\beta(1-\eta))$.
$c := \frac{20}{\varphi_{\text{in}}^2}$, $t := c\ln(\text{vol}(V_G))$    ▷ Observe: $c > 0$.
$\sigma := \frac{192sk(1+\eta)}{\text{vol}(V_G)}$.
$\mu_{\text{thres}} := \frac{1}{2} \cdot \frac{8(k+1)\ln(12(k+1))}{\beta\cdot(1-\eta)} \times \text{vol}(V_G)^{-1-120c\varphi_{\text{out}}}$.
$\mu_{\text{err}} = \frac{1}{3} \cdot \frac{8(k+1)\ln(12(k+1))}{\beta\cdot(1-\eta)} \times \text{vol}(V_G)^{-1-120c\varphi_{\text{out}}}$
$R := \max\left(\frac{100s^2\sigma^{1/2}}{\mu_{\text{err}}}, \frac{200s^4\sigma^{3/2}}{\mu_{\text{err}}^2}\right)$.
if ESTIMATEWITHOUTORACLE$(G, k, s, t, \sigma, R, \eta) \le \mu_{\text{thres}}$ then
    Accept $G$.
else
    Reject $G$.
```

Recall from Definition 2.7 that with the graph G we associated a random walk, and let M be the transition matrix of that random walk. For a vertex a of G, denote by $\mathbf{p}_a^t = M^t \mathbb{1}_a$ the probability distribution of of a t step random walk starting from a. Recall that ESTIMATE assumed the existence of an oracle that takes a vertex a of G as input, and returns $D^{-\frac{1}{2}}M^t\mathbb{1}_a$. ESTIMATEWITHOUTORACLE simulates the behavior of the oracle by running several t-step random walks from a. For any vertex b, the fraction of the random walks ending in b is taken as an estimate of $\mathbf{p}_a^t(b) = \mathbb{1}_b^T M^t \mathbb{1}_a$, the probability that the t-step random walk started from a ends in b. However, for this estimate to have sufficiently small variance, the quantity $\|D^{-\frac{1}{2}}\mathbf{p}_a^t\|_2^2$ needs to be small enough. To check this, ESTIMATEWITHOUTORACLE uses the procedure **ℓ_2^2-norm tester**, whose guarantees are formally specified in the following lemma.

Lemma 2.18. *Let $G = (V_G, E_G)$. Let $a \in V_G$, $\sigma > 0$, $0 < \delta < 1$, and $R \ge \frac{16\sqrt{vol(G)}}{\delta}$. Let $t \ge 1$, and $\mathbf{p}_a^t$ be the probability distribution of the endpoints of a t-step random walk starting from a. There* *exists an algorithm, denoted by ℓ_2^2**-norm tester**(G, a, σ, R), that outputs accept if* $\|D^{-\frac{1}{2}}\mathbf{p}_a^t\|_2^2 \leq \frac{\sigma}{4}$*, and outputs reject if* $\|D^{-\frac{1}{2}}\mathbf{p}_a^t\|_2^2 > \sigma$*, with probability at least* $1-\delta$*. The running time of the tester is* $O(R \cdot t)$*.*

Our ℓ_2^2**-norm tester** is a modification of ℓ_2^2**-norm tester** in [CPS15]. We defer the proof of this lemma to Appendix A.2. The running time of ℓ_2^2**-norm tester** is independent of σ, since $\|D^{-\frac{1}{2}}\mathbf{p}_a^t\|_2^2 \geq \frac{1}{\mathrm{vol}(G)}$ for all $a \in V_G$. We will use the following definition of a (σ, t)-good vertex for the rest of the section.

Definition 2.8. We say that a vertex $a \in V_G$, is (σ, t)*-good* if $\|D^{-\frac{1}{2}}\mathbf{p}_a^t\|_2^2 \leq \sigma$.

We first claim that for all multisets S containing only (σ, t)-good vertices, with a good probability over the R random walks, the quantity $\mathcal{G}$ that Algorithm 3 returns is a good approximation to $(D^{-\frac{1}{2}}M^tS)^T(D^{-\frac{1}{2}}M^tS)$ in Frobenius norm.

Lemma 2.19. *Let* $G = (V_G, E_G)$ *be a graph. Let* $0 < \sigma \leq 1$ $t > 0$*,* $\mu_{err} > 0$*, k be an integer, and let S be a multiset of s vertices, all whose elements are* (σ, t)*-good. Let*

$$R = \max\left(\frac{100s^2\sigma^{1/2}}{\mu_{err}}, \frac{200s^4\sigma^{3/2}}{\mu_{err}^2}\right).$$

For each $a \in S$ *and each* $b \in V_G$*, let* $\mathbf{q}_a(b)$ *and* $\mathbf{q}'_a(b)$ *be random variables which denote the fraction out of the R random walks starting from a, which end in b. Let Q and Q' be matrices whose columns are* $(D^{-\frac{1}{2}}\mathbf{q}_a)_{a\in S}$ *and* $(D^{-\frac{1}{2}}\mathbf{q}'_a)_{a\in S}$ *respectively. Let* $\mathcal{G} = \frac{1}{2}\left(Q^TQ' + Q'^TQ\right)$*. Then with probability at least* $49/50$*,* $|\mu_{k+1}(\mathcal{G}) - \mu_{k+1}((D^{-\frac{1}{2}}M^tS)^T(D^{-\frac{1}{2}}M^tS))| \leq \mu_{err}$*.*

The proof of the lemma is given in Appendix A.2. We now prove that Algorithm 4 indeed outputs a YES with good probability on a YES instance. For this, we need the following lemma which is a modification of Lemma 4.3 of [CPS15], and the proof of this lemma is deferred to Appendix A.2.

Lemma 2.20. *For all* $0 < \alpha < 1$*, and all* $G = (V_G, E_G)$ *which is* (k, φ_{in})*-clusterable, there exists* $V'_G \subseteq V_G$ *with* $vol(V'_G) \geq (1-\alpha)vol(V_G)$ *such that for any* $t \geq \frac{2\ln(vol(V_G))}{\varphi_{in}^2}$*, every* $u \in V'_G$ *is* $\left(\frac{2k}{\alpha\cdot vol(V_G)}, t\right)$*-good.*

Theorem 2.10. *Let* $\varphi_{in} > 0$*, and integer* $k \geq 1$*. Then for every* (k, φ_{in})*-clusterable graph* $G = (V_G, E_G)$ *(see definition 2.2), with* $\min_{v\in V_G} deg(v) \geq 1$*, Algorithm 4 accepts G with probability at least* $\frac{5}{6}$*.*

Proof. If Algorithm 4 outputs a NO one of the following events must happen.

- E_1: Some vertex in S is not $(\frac{\sigma}{4}, t)$-good.
- E_2: All vertices in S are $(\frac{\sigma}{4}, t)$-good, but ℓ_2^2**-norm tester** fails on some vertex.

- E_3: All vertices in S are $(\frac{\sigma}{4}, t)$-good, and ℓ_2^2**-norm tester** succeeds on all vertices, but $|\mu_{k+1}(Q^T Q) - \mu_{k+1}((D^{-\frac{1}{2}} M^t S)^T (D^{-\frac{1}{2}} M^t S))| > \mu_{\text{err}}$.

If none of the above happen then Algorithm 3 returns

$$\mu_{k+1}(\mathscr{G}) \le \mu_{k+1}((D^{-\frac{1}{2}} M^t S)^T (D^{-\frac{1}{2}} M^t S)) + \mu_{\text{err}} \le \mu_{\text{yes}} + \mu_{\text{err}} < \mu_{\text{thres}},$$

and Algorithm 4 accepts.

Recall that we use **sampler**(G,η) to sample vertices, where $\eta = \frac{1}{2}$. Apply Lemma 2.20 with $\alpha = \frac{1}{24s(1+\eta)}$. Then by the union bound, with probability at least $1 - \alpha \cdot (1+\eta) = 1 - \frac{1}{24}$ all the vertices in S are $\left(\frac{48sk(1+\eta)}{\text{vol}(V_G)}, t\right)$-good, that is, $(\frac{\sigma}{4}, t)$-good, where $\sigma = \frac{192sk}{\text{vol}(V_G)}$, as chosen in Algorithm 3. Thus, $\Pr[E_1] \le \frac{1}{24}$. Given that E_1 doesn't happen, by Lemma 2.8, on any sample, ℓ_2^2**-norm tester** fails with probability at most $\frac{16\sqrt{\text{vol}(V_G)}}{r} < \frac{1}{12s}$ for $r = 192s\sqrt{\text{vol}(V_G)}$, as chosen in Algorithm 3. Thus, with probability at least $1 - \frac{1}{12}$, ℓ_2^2**-norm tester** succeeds on all the sampled vertices, which implies $\Pr[E_2] \le \frac{1}{12}$. Given that both E_1 and E_2 don't happen, by Lemma 2.19, with probability at least $\frac{49}{50}$, Algorithm 3 returns a value that is at most μ_{err} away from $\mu_{k+1}((D^{-\frac{1}{2}} M^t S)^T (D^{-\frac{1}{2}} M^t S))$. Thus, $\Pr[E_3] \le \frac{1}{50}$. By the union bound, the probability that Algorithm 4 rejects is at most $\frac{1}{24} + \frac{1}{12} + \frac{1}{50} < \frac{1}{6}$. □

Next, we prove that Algorithm 4 indeed returns a NO with good probability on a NO instance.

Theorem 2.11. *Let $\varphi_{out} > 0$, $\beta \in (0,1)$, and integer $k \ge 1$. Then for every $(k, \varphi_{out}, \beta)$-unclusterable graph $G = (V_G, E_G)$ (see definition 2.2), with $\min_{v \in V_G} deg(v) \ge 1$, Algorithm 4 rejects G with probability at least $\frac{4}{7}$.*

Proof. If the algorithm outputs a YES, then one of the following events must happen.

- E_1: Some vertices in S are not (σ, t)-good, but ℓ_2^2**-norm tester** misses these and passes all vertices.
- E_2: All vertices in S are (σ, t)-good, but $|\mu_{k+1}(\mathscr{G}) - \mu_{k+1}((D^{-\frac{1}{2}} M^t S)^T (D^{-\frac{1}{2}} M^t S))| > \mu_{\text{err}}$.
- E_3: $\mu_{k+1}((D^{-\frac{1}{2}} M^t S)^T (D^{-\frac{1}{2}} M^t S)) < \mu_{\text{no}} = \frac{8(k+1)\ln(12(k+1))}{\beta \cdot (1-\eta)} \times \text{vol}(V_G)^{-1-120c\varphi_{\text{out}}}$.

If none of the above happen then Algorithm 3 returns

$$\mu_{k+1}(\mathscr{G}) \ge \mu_{k+1}((D^{-\frac{1}{2}} M^t S)^T (D^{-\frac{1}{2}} M^t S)) - \mu_{\text{err}} \ge \mu_{\text{no}} - \mu_{\text{err}} > \mu_{\text{thres}},$$

and Algorithm 4 rejects.

By Lemma 2.8, the probability that ℓ_2^2**-norm tester** passes a bad vertex is at most $\frac{16\sqrt{\text{vol}(V_G)}}{r} < \frac{1}{12s}$ for $r = 192s\sqrt{\text{vol}(V_G)}$, as chosen in Algorithm 3. Thus, $\Pr[E_1] \le \frac{1}{12}$. If all vertices in S are

(σ, t)-good, by Lemma 2.19, with probability at least $\frac{49}{50}$, Algorithm 3 returns a value that is at most μ_{err} away from $\mu_{k+1}((D^{-\frac{1}{2}}M^t S)^T(D^{-\frac{1}{2}}M^t S))$. Thus, $\Pr[E_2] \le \frac{1}{50}$. Finally, by Theorem 2.9, $\Pr[E_3] \le \frac{1}{3}$. By the union bound, the probability that Algorithm 4 accepts is at most $\frac{1}{12} + \frac{1}{50} + \frac{1}{3} < \frac{3}{7}$ □

Now we are set to prove Theorem 2.1.

Proof of Theorem 2.1. The correctness of the algorithm is guaranteed by Theorems 2.10 and 2.11. Since these theorems give correctness probability that is a constant larger than 1/2, it can be boosted up to 2/3 using standard techniques (majority of the answers of a sufficiently large constant number of independent runs). It remains to analyze the query complexity. The running time of the **sampler** algorithm to sample each vertex is $\tilde{O}(\frac{|V_G|}{\mathrm{vol}(G)})$. Hence in total the query complexity of sampling is $\tilde{O}(s \cdot \sqrt{\mathrm{vol}(G)})$. For each of the s sampled vertices, we run **ℓ_2^2-norm tester** once, followed by R random walks of t steps each. Each call to the **ℓ_2^2-norm tester** takes $O(rt) = O(st\sqrt{\mathrm{vol}(V_G)}) = O(st\sqrt{m})$ queries, as guaranteed by Lemma 2.18. The random walks from each vertex take $O(Rt)$ time. Thus, the overall query complexity is $O(srt + sRt + s\sqrt{m})$. Substituting the values of s, r, R, and t as defined in Algorithm 4, and noting that $m = \mathrm{vol}(V_G)/2$, we get the required bound. □

2.4 Lower Bound for the noisy parities problem with applications

Recall the definition of **NoisyParities** from Section 2.1, and consider a deterministic algorithm querying T out of n vertices in an instance. We wish to prove the following lower bound on the number of queries T needed to get a nontrivial advantage over a random guess.

Theorem 2.4 (restated). Consider a deterministic algorithm ALG for the **NoisyParities** problem with parameters d and ε. Let $b = 1/(8\ln d)$. Suppose ALG makes at most $n^{1/2+\delta}$ queries on n vertex graphs, where $\delta < \min(1/16, b\varepsilon)$. Then ALG gives the correct answer with probability at most $1/2 + o(1)$.

We present the proof of this theorem in Section 2.4.3, but first we setup some preliminaries here, and then use this theorem to establish query complexity lower bounds in Section 2.4.2.

2.4.1 Preliminaries and notation

Random d-regular Graphs and the Configuration Model

Recall that the **NoisyParities** problem (Definition 2.5) has a random d-regular graph generated according to the configuration model as its underlying graph. The configuration model of Bollobás generates a random d regular graph $G = (V, E)$ over a set V of n vertices (provided dn is even) as follows. It first generates d half-edges on each vertex and identifies the set of half-edges with $V \times [d]$. Then in each round, an arbitrary unpaired half-edge (u, i) of some arbitrary vertex u is picked, and it is paired up with a uniformly random unpaired half-edge (v, j). This results in the addition of an edge (u, v) to E. This continues until all the half-edges are paired up. (This might result in self-loops and parallel edges, so G is not necessarily simple.) The following is known about the expansion of random d-regular graphs generated by the configuration model [Bol88].

Fact 2.1. *For $d \geq 3$ let $\eta(d) \in (0,1)$ be such that $(1-\eta(d))\log_2(1-\eta(d)) + (1+\eta(d))\log_2(1+\eta(d)) > 4/d$. Then with probability $1 - o(1)$, a random d-regular graph on n vertices generated from the configuration model has expansion at least $(1-\eta(d))/2$.*

Definition 2.9. Given a graph $G = (V, E)$ and a vertex $v \in V$, $B_G(v, r)$ denotes the ball centered at v with radius r, that is, the set of vertices which are at a distance at most r from v in G.

The following bound on the size of a ball follows from a simple calculation.

Proposition 2.1. *If G is a (subgraph of a) d-regular graph for $d \geq 2$, then for any vertex v, $|B_G(v,0)| = 1$, $|B_G(v,1)| \leq d+1$, $|B_G(v,2)| \leq d^2+1$, and for $r > 2$, $|B_G(v,r)| \leq d^r$.*

Fourier Transform of Boolean Functions

Fix a finite set E with $|E| = m$, and identify each subset S of E with a boolean vector in $\{0,1\}^E$ in the natural way. The set of functions $f : \{0,1\}^E \longrightarrow \mathbb{R}$ form a 2^m dimensional vector space. Define an inner product $\langle \cdot, \cdot \rangle$ on this vector space as $\langle f, g \rangle = 2^{-m} \sum_{x \in \{0,1\}^E} f(x) g(x)$. For each $\alpha \in \{0,1\}^n$, define its *characteristic function* $\chi_\alpha : \{0,1\}^E \longrightarrow \mathbb{R}$ as $\chi_\alpha(x) = (-1)^{\alpha \cdot x}$. Then the set of functions $\{\chi_\alpha : \alpha \in \{0,1\}^E\}$ form an orthonormal basis with respect to the inner product $\langle \cdot, \cdot \rangle$. Thus, any function $f : \{0,1\}^E \longrightarrow \mathbb{R}$ can be resolved in this basis as $f = \sum_{\alpha \in \{0,1\}^E} \widehat{f}(\alpha) \chi_\alpha$, where

$$\widehat{f}(\alpha) = \langle f, \chi_\alpha \rangle = 2^{-m} \sum_{x \in \{0,1\}^E} f(x) \chi_\alpha(x).$$

We will need the following properties of the Fourier transform, whose proofs can be found in [O'D14].

Proposition 2.2. *Let $f : \{0,1\}^E \to \mathbb{R}$ be given by $f(z) = \varepsilon^{|z|}(1-\varepsilon)^{|E|-|z|}$, where $|z|$ denotes the number of ones in the the vector z. Then $\widehat{f}$ is given by $\widehat{f}(\alpha) = 2^{-|E|}(1-2\varepsilon)^{|\alpha|}$ for all $\alpha \in \{0,1\}^E$.*

Proposition 2.3 (Fourier transforms of affine subspaces). *Let S be a subspace of $\{0,1\}^E$ of dimension r, and $b \in \{0,1\}^E$. Let $f : \{0,1\}^E \to \mathbb{R}$ be given by $f(z) = 1$ if $\gamma \cdot z = \gamma \cdot b$ for all $\gamma \in S$, and $f(z) = 0$ otherwise. Then $\widehat{f}$ is given by $\widehat{f}(\alpha) = 2^{-(m-r)}(-1)^{\alpha \cdot b}$ if $\alpha \in S$, and $\widehat{f}(\alpha) = 0$ otherwise.*

Proposition 2.4 (Convolution Theorem). *Let $f, g : \{0,1\}^E \to \mathbb{R}$. Then $\widehat{fg}$ is given by*

$$\widehat{fg}(\alpha) = \sum_{\beta \in \{0,1\}^n} \widehat{f}(\beta) \widehat{g}(\alpha + \beta).$$

Incidence Matrices, Eulerian Subgraphs, Spanning Forests

Given a graph $G = (V, E)$, its *incidence matrix* is the binary $V \times E$ matrix whose (v, e)-entry is 1 if v is an endpoint of e, and 0 otherwise. A graph is *Eulerian* if and only if each of its vertices has even degree, or equivalently, the mod-2 nullspace of its incidence matrix contains the all ones vector $\mathbb{1}_E$. The set of subgraphs of G is in the natural one-to-one correspondence with $\{0,1\}^E$, where the set of Eulerian subgraphs of G corresponds to the nullspace of the incidence matrix of G.

We define the *rank* of a graph to be the cardinality of its spanning forest, which is also equal to the rank of its incidence matrix. Fix a spanning forest F of a graph $G = (V, E)$. We use this forest to construct a basis for $\{0,1\}^E$, the vector space of subgraphs of G, as follows. For each $e \in F$, let $v_e(e) = 1$ and $v_e(e') = 0$ for $e' \neq e$. For each $e \in E \setminus F$, define the vector $v_e \in \{0,1\}^E$ as $v_e(e') = 1$ if e' belongs to the unique cycle in $F \cup \{e\}$, and $v_e(e') = 0$ otherwise. Then the collection of vectors $\{v_e : e \in F\} \cup \{v_e : e \in E \setminus F\}$ forms a basis of $\{0,1\}^E$. Here, the set $\{v_e : e \in E \setminus F\}$ spans the subspace of Eulerian subgraphs of G, whereas $\{v_e : e \in F\}$ spans a complementary subspace: the space of sub-forests of F. As a consequence, we have that every subgraph of G can be written uniquely as a symmetric difference of an Eulerian subgraph of G and a sub-forest of F.

Lemma 2.21. *Let F be a spanning forest of a graph $G = (V, E)$. Then the Eulerian subgraphs of G are in one-to-one correspondence with subsets of $E \setminus F$, where the bijection is given by $E^* \leftrightarrow E^* \setminus F$. In other words, for every $S \subseteq E \setminus F$, there exists a unique Eulerian subgraph E^* of G such that $E^* \setminus F = S$.*

Proof. For each $e \in E \setminus F$, let $C(e)$ denote the unique cycle in $F \cup \{e\}$. First we prove that $E^* \to E^* \setminus F$ is surjective. Let $S \subseteq E \setminus F$. Then $E^* = \bigoplus_{e \in S} C(e)$ is Eulerian and $E^* \setminus F = S$. Next, we prove that $E^* \to E^* \setminus F$ is injective. Let $S \subseteq E \setminus F$ and let E^* and E' be Eulerian subgraphs of G such that $E^* \setminus F = E' \setminus F = S$. Then $(E^* \oplus E') \setminus F = \emptyset$, which means $E^* \oplus E' \subseteq F$. But F is a forest and $E^* \oplus E'$ is Eulerian. Therefore, $E^* \oplus E' = \emptyset$, which means $E^* = E'$. □

Lemma 2.22. *Let F be a spanning forest of a graph $G = (V, E)$ such that the endpoints of the edges in $E \setminus F$ are pairwise distance Δ apart in F. Let $G^* = (V, E^*)$ be an Eulerian subgraph of G. Then $|E^*| \geq \Delta \cdot |E^* \setminus F|$.*

Proof. Partition the edges of G^* into cycles, and consider each cycle one by one. Between any two occurrences of non-forest edges in the cycle, we have a path consisting of edges from F. By the separation condition on the endpoints of edges not in F, each such path must have length at least Δ. Thus, we have at least Δ forest edges per non-forest edge in G^*. □

Total Variation Distance

Definition 2.10. Let Ω be a finite set. The *Total Variation Distance (TVD)* between two probability distributions p and p' over Ω, denoted by $\mathrm{TVD}(p, p')$ (resp. two random variables X and

X' taking values from Ω, denoted by $\mathrm{TVD}(X,X')$) is defined as $(1/2)\cdot\sum_{s\in\Omega}|p(s)-p'(s)|$ (resp. $(1/2)\cdot\sum_{s\in\Omega}|\Pr[X=s]-\Pr[X'=s]|$).

Lemma 2.23. *Let X_1, X_1' be random variables taking values in Ω_1, and let X_2, X_2' be random variables taking values in Ω_2. Then*

$$TVD((X_1,X_2),(X_1',X_2')) \le TVD(X_1,X_1') + \sum_{s\in\Omega_1} \Pr[X_1=s]\cdot TVD((X_2|X_1=s),(X_2'|X_1'=s)).$$

Proof. For $s_1\in\Omega_1$, let $p(s_1)=\Pr[X_1=s_1]$, and $p'(s_1)=\Pr[X_1'=s_1]$. For $s_1\in\Omega_1$ and $s_2\in\Omega_2$, let $q_{s_1}(s_2)=\Pr[X_2=s_2|X_1=s_1]$, and $q'_{s_1}(s_2)=\Pr[X_2'=s_2|X_1'=s_1]$. Then we have, $\Pr[(X_1,X_2)=(s_1,s_2)]=p(s_1)\cdot q_{s_1}(s_2)$ and $\Pr[(X_1',X_2')=(s_1,s_2)]=p'(s_1)\cdot q'_{s_1}(s_2)$.

$$\begin{aligned}\mathrm{TVD}((X_1,X_2),(X_1',X_2')) &= \frac{1}{2}\sum_{(s_1,s_2)\in\Omega_1\times\Omega_2}\left|p(s_1)\cdot q_{s_1}(s_2)-p'(s_1)\cdot q'_{s_1}(s_2)\right| \\ &\le \frac{1}{2}\sum_{(s_1,s_2)\in\Omega_1\times\Omega_2}\left[\left|p(s_1)\cdot q_{s_1}(s_2)-p(s_1)\cdot q'_{s_1}(s_2)\right|+\left|p(s_1)\cdot q'_{s_1}(s_2)-p'(s_1)\cdot q'_{s_1}(s_2)\right|\right].\end{aligned}$$

The first term above is

$$\begin{aligned}\frac{1}{2}\sum_{(s_1,s_2)\in\Omega_1\times\Omega_2}\left|p(s_1)\cdot q_{s_1}(s_2)-p(s_1)\cdot q'_{s_1}(s_2)\right| &= \sum_{s_1\in\Omega_1}p(s_1)\cdot\frac{1}{2}\sum_{s_2\in\Omega_2}\left|q_{s_1}(s_2)-q'_{s_1}(s_2)\right| \\ &= \sum_{s\in\Omega_1}\Pr[X_1=s]\cdot\mathrm{TVD}((X_2|X_1=s),(X_2'|X_1'=x)),\end{aligned}$$

while the second term is

$$\frac{1}{2}\sum_{(s_1,s_2)\in\Omega_1\times\Omega_2}\left|p(s_1)\cdot q'_{s_1}(s_2)-p'(s_1)\cdot q'_{s_1}(s_2)\right| = \frac{1}{2}\sum_{s_1\in\Omega_1}\left|p(s_1)-p'(s_1)\right|\cdot\sum_{s_2\in\Omega_2}q'_{s_1}(s_2)=\mathrm{TVD}(X_1,X_1'),$$

since $\sum_{s_2\in\Omega_2}q'_{s_1}(s_2)=\sum_{s_2\in\Omega_2}\Pr[X_2'=s_2|X_1'=s_1]=1$ for all $s_1\in\Omega_1$. □

Corollary 2.1. *Let X_1, X_1' be random variables taking values in Ω_1, and let X_2, X_2' be random variables taking values in Ω_2. Let $\mathcal{E}_1\subseteq\Omega_1$. Then*

$$TVD((X_1,X_2),(X_1',X_2')) \le TVD(X_1,X_1')+\Pr[X_1\notin\mathcal{E}]+\sum_{s\in\mathcal{E}}\Pr[X_1=s]\cdot TVD((X_2|X_1=s),(X_2'|X_1'=s)).$$

Proof. Follows since for all $s\in\Omega_1$ (and in particular, for $s\notin\mathcal{E}$), $\mathrm{TVD}((X_2|X_1=s),(X_2'|X_1'=s))\le 1$ by definition. □

Corollary 2.2. *For $i=1$ to T, let X_i and X_i' be random variables taking values in Ω_i. For $i=1$ to $T-1$, let $\mathcal{E}_i\subseteq\Omega_1\times\cdots\times\Omega_i$, such that $(s_1,\ldots,s_i)\in\mathcal{E}_i$ implies $(s_1,\ldots,s_{i-1})\in\mathcal{E}_{i-1}$. Then*

$$TVD((X_1,\ldots,X_T),(X_1',\ldots,X_T'))\le\Pr[(X_1,\ldots,X_T)\notin\mathcal{E}_T]+$$

$$\sum_{i=1}^{T}\sum_{(s_1,\ldots,s_{i-1})\in\mathcal{E}_{i-1}}\Pr\left[\bigwedge_{j=1}^{i-1}X_j=s_j\right]\cdot TVD\left(\left(X_i\mid\bigwedge_{j=1}^{i-1}X_j=s_j\right),\left(X_i'\mid\bigwedge_{j=1}^{i-1}X_j'=s_j\right)\right).$$

Proof. Follows by repeated application of Corollary 2.1. □

2.4.2 Reductions to partition testing and MAX-CUT

Reduction to PartitionTesting

In this section, we show how the problem **NoisyParities** reduces to testing **PartitionTesting**$(k, \varphi_{\text{in}}, \varphi_{\text{out}}, \beta)$, even for $k = 1$ and any $\beta \le 1$. By this reduction, we establish a lower bound of $n^{1/2+\Omega(\varphi_{\text{out}})}$ on the number of queries required to test whether a graph is $(1, \varphi_{\text{in}})$-clusterable for some constant φ_{in} (the YES case), or it is $(2, \varphi_{\text{out}}, \beta)$-unclusterable for any constant $\beta \le 1$ (the NO case).

Theorem 2.5 (restated). There exist positive constants φ_{in} and b such that for all $\varphi_{\text{out}} \le 1$, any algorithm that distinguishes between a $(1, \varphi_{\text{in}})$-clusterable graph (that is, a φ_{in}-expander) and a $(2, \varphi_{\text{out}}, 1)$-unclusterable graph on n vertices with success probability at least $2/3$ must make at least $(n/2)^{1/2+b\varphi_{\text{out}}/2}$ queries, even when the input is restricted to d-regular graphs for a large enough constant d.

The reduction is given by Algorithm 5.

```
Algorithm 5 ReductionToPartitionTesting ($G = (V, E)$, $y : E \longrightarrow \{0, 1\}$)
Input: $G = (V, E)$, labeling $y : E \longrightarrow \{0, 1\}$
$V' := V \times \{0, 1\}$.        ▷ We denote the vertex $(v, b) \in V \times \{0, 1\}$ by $v^b$ for readability.
$E'_0 := \bigcup_{e=(u,v)\in E:\ y(e)=0} \{(u^0, v^0), (u^1, v^1)\}$.
$E'_1 := \bigcup_{e=(u,v)\in E:\ y(e)=1} \{(u^0, v^1), (u^1, v^0)\}$.
$E' = E'_0 \cup E'_1$.
return $G' = (V', E')$.
```

Observe that the reduction is "query complexity preserving" in the sense that any query from a **PartitionTesting** algorithm asking the neighbors of a vertex $v^b \in V'$ can be answered by making (at most) one query, asking the yet undisclosed edges incident on v in G and their labels. To establish the correctness of the reduction, it is sufficient to prove:

1. The YES case: If the edges of G are labeled independently and uniformly at random, then G' is an expander with high probability.
2. The NO case: If each edge $e = (u, v)$ of G is labeled $X(u) + X(v) + Z(u, v)$, where $Z(u, v)$ is 1 with probability ε, then with high probability G' contains a cut with n vertices on each side whose expansion is $O(\varepsilon)$.

Lemma 2.24. *Let $G = (V, E)$ be a d-regular φ-expander with $|V| = n$. Suppose each edge $(u, v) \in E$ independently and uniformly given label $Y(u, v) \in \{0, 1\}$. Suppose ReductionToPartitionTesting on input (G, Y) returns the graph $G' = (V', E')$. Then G' is a* $\min(\varphi/4, 1/32)$-*expander with probability at least* $1 - 2^{2n} \cdot \exp(-dn/256)$.

Proof. We need to prove that every $C \subseteq V'$ with $|C| \le |V'|/2 = n$ expands well. Let $C_0 = \{v \in V : v^0 \in C\}$ and $C_1 = \{v \in V : v^0 \in C\}$ be the "projections" of C on the two halves of $V' = V \times \{0,1\}$, each half identified with V, so that $|C| = |C_0| + |C_1|$. Then at least one of the following must hold.

1. $|C_0 \cup C_1| \le n/2$.
2. $|C_0 \cap C_1| \ge n/4$.
3. $|C_0 \oplus C_1| \ge n/4$.

In the first case, consider the set of edges which cross the set $C_0 \cup C_1$ in G. Since G is a φ-expander and $|C_0 \cup C_1| \le n/2$, the number of such edges is at least $\varphi d|C_0 \cup C_1|$. For each such edge, one of its two copies in G' must cross the set C. Therefore, the expansion of C is at least

$$\frac{\varphi d|C_0 \cup C_1|}{d|C|} \ge \frac{\varphi}{2}.$$

In the second case, we cannot have $|C_0 \cap C_1| > n/2$, otherwise we contradict the assumption that $|C| \le n$. Therefore, $|C_0 \cap C_1| \le n/2$. Consider the set of edges which cross $|C_0 \cap C_1|$ in G. Again, since G is a φ-expander and $|C_0 \cap C_1| \le n/2$, the number of such edges is at least $\varphi d|C_0 \cap C_1|$. As before, for each such edge e, at least one of its two copies in E' must cross the set C. Therefore, the expansion of C is at least

$$\frac{\varphi d|C_0 \cap C_1|}{d|C|} \ge \frac{\varphi n/4}{n} = \frac{\varphi}{4}.$$

Finally, consider the third case. Let m be the number of edges in G, both of whose endpoints are in $C_0 \oplus C_1$. Therefore, the number of edges in G with exactly one endpoint in $C_0 \oplus C_1$ is $d|C_0 \oplus C_1| - 2m$. We split into two sub-cases depending on whether $m \le d|C_0 \oplus C_1|/4$, or $m > d|C_0 \oplus C_1|/4$.

In the first sub-case, consider the $d|C_0 \oplus C_1| - 2m$ edges of G with exactly one endpoint in $C_0 \oplus C_1$. For each of such edge, (exactly) one of its copies in G' crosses the set C. Therefore, the expansion of C is at least

$$\frac{d|C_0 \oplus C_1| - 2m}{d|C|} \ge \frac{d|C_0 \oplus C_1| - d|C_0 \oplus C_1|/2}{d|C|} = \frac{|C_0 \oplus C_1|/2}{|C|} \ge \frac{n/8}{n} = \frac{1}{8}.$$

In the second sub-case, consider each edge $(u,v) \in E$ with $u, v \in C_0 \oplus C_1$. Suppose both u and v belong to the same C_i. If $Y(u,v) = 0$, none of the copies of the edge (u,v) in E' crosses the set C, and if $Y(u,v) = 1$, then both copies cross. Similarly, suppose one of u and v belongs to C_0 and the other belongs to C_1. If $Y(u,v) = 1$, none of the copies of the edge (u,v) in E' crosses the set C, and if $Y(u,v) = 0$, then both copies cross. Thus for each of the m edges

$(u, v) \in E$ with $u, v \in C_0 \oplus C_1$, both of its copies cross C with probability $1/2$, and none crosses with probability $1/2$. This happens independently for all the m edges. Therefore, by Chernoff bound, the number of edges both of whose copies cross the set C is at least $m/4 \geq d|C_0 \oplus C_1|/16$ with probability at least

$$1 - \exp(-m/16) \geq 1 - \exp(-d|C_0 \oplus C_1|/64) \geq 1 - \exp(-dn/256).$$

Assuming this happens, the expansion of C is at least

$$\frac{2 \times d|C_0 \oplus C_1|/16}{d|C|} = \frac{|C_0 \oplus C_1|/8}{|C|} \geq \frac{n/32}{n} = \frac{1}{32}.$$

This holds for each set C falling into this sub-case. Applying the union bound over all the at most 2^{2n} sets falling into this sub-case, we have that with probability at least $1 - 2^{2n} \cdot \exp(-dn/256)$, all sets falling into this sub-case have expansion at least $1/32$. □

Lemma 2.25. *Let $G = (V, E)$ be a d-regular graph with $|V| = n$. For each $v \in V$, let $X(v)$ be an independent uniformly random bit. For each edge $(u, v) \in E$, let $Z(u, v)$ be an independent random bit which is 1 with probability ε and 0 otherwise. Suppose each edge $(u, v) \in E$ is labeled $Y(u, v) = X(u) + X(v) + Z(u, v)$. Suppose ReductionToPartitionTesting on input (G, Y) returns the graph $G' = (V', E')$. Then with probability at least $1 - \exp(-\varepsilon nd/6)$, there exists a set $V^* \subseteq V'$ with $|V^*| = n = |V'|/2$ whose expansion is at most 2ε.*

Proof. Define V^* as

$$V^* = \{v^0 \,:\, v \in V, X(v) = 0\} \cup \{v^1 \,:\, v \in V, X(v) = 1\},$$

so that $|V^*| = n$. By a case-by-case consideration, it is easy to verify that if for $e = (u, v) \in E$ we have $Z(e) = 1$, then the two edges between u^0, v^0, u^1, v^1 cross the cut $(V^*, V' \setminus V^*)$. Conversely, if $Z(e) = 0$, then one of the edges between u^0, v^0, u^1, v^1 lies within V^* and the other lies outside V^*. Thus, the number of edges crossing the cut is twice the size of the set $\{e \in E \,:\, Z(e) = 1\}$. The expectation of the size of this set is $\varepsilon \cdot |E| = \varepsilon nd/2$. Since $\{Z(e)\}_{e \in E}$ are independent and identically distributed, application of the Chernoff bound gives us that the size of the set $\{e \in E \,:\, Z(e) = 1\}$ is at most εnd with probability at least $1 - \exp(-\varepsilon nd/6)$. Thus, the number of edges crossing the cut $(V^*, V' \setminus V^*)$ is at most $2\varepsilon nd$ with high probability. Dividing by nd, the volume of V^*, we have that the expansion of V^* is at most 2ε with probability at least $1 - \exp(-\varepsilon nd/6)$. □

Proof of Theorem 2.5. Let $d = 512$, $\varphi = (1 - \eta(d))/2 \approx 0.45$, and $\varphi_{\text{in}} = \min(\varphi/4, 1/32)$. As before, let $b = 1/(8 \ln d)$ (with $d = 512$ now). Given $\varphi_{\text{out}} \leq 1$, let $\varepsilon = \varphi_{\text{out}}/2 \leq 1/2$. Suppose there is an algorithm for **PartitionTesting** which makes $(n/2)^{1/2+\delta}$ queries on n vertex graphs and outputs the correct answer with probability at least $2/3$, for some $\delta < b\varphi_{\text{out}}/2 = b\varepsilon = \min(1/16, b\varepsilon)$ (note that $b\varepsilon \leq 1/(16 \ln 512) \leq 1/16$). Then for any probability distribution $\mathscr{D}$ over n-vertex **PartitionTesting** instances, there exists a deterministic algorithm ALG($\mathscr{D}$) making $O(n^{1/2+\delta})$

queries which outputs the correct answer with probability at least $2/3$, on a random instance of **PartitionTesting** drawn from the distribution.

Let (G, y) be a random instance of the **NoisyParities** problem with parameters d and ε, where $G = (V, E)$ is a graph and $y : E \longrightarrow \{0, 1\}$ is an edge lebeling. Apply ReductionToPartitionTesting to (G, y), and thus, get a random instance G' of **PartitionTesting** from the appropriate probability distribution $\mathscr{D}$. Run ALG($\mathscr{D}$) on this instance and return the answer. Note that to answer one query of ALG($\mathscr{D}$), we make at most one query into G. Thus, this reduction gives an algorithm ALG$'$ for **NoisyParities** making at most $n^{1/2+\delta}$ queries.

The underlying graph G is a random d-regular graph on n vertices. By Fact 2.1, with high probability, G is a φ-expander. Hence, by Lemma 2.24, if (G, y) is a YES instance, then with high probability the reduced graph G' is a φ_{in}-expander for $\varphi_{\text{in}} = \min(\varphi/4, 1/32)$ (we chose d large enough so that the failure probability $2^{2n} \cdot \exp(-dn/256)$ in Lemma 2.24 becomes $o(1)$). On the other hand, if (G, y) is a NO instance, then by Lemma 2.25, the reduced graph G' is a graph on $2n$ vertices containing with high probability a subset of n vertices whose expansion is at most $2\varepsilon = \varphi_{\text{out}}$. Thus, G' is $(2, \varphi_{\text{out}}, 1)$-unclusterable. Hence, the reduction succeeds with probability $1 - o(1)$. Since ALG answers correctly with probability at least $2/3$, ALG$'$ answers correctly with probability at least $2/3 - o(1)$.

However, by Theorem 2.4, since ALG$'$ makes at most $n^{1/2+\delta}$ queries, ALG$'$ can be correct with probability at most $1/2 + o(1)$. This is a contradiction. □

Reduction to approximating MAX-CUT value

In this section, we show how the problem **NoisyParities** reduces to estimating maxcut. By this reduction, we establish the following theorem.

Theorem 2.6 (restated). There exists a constant β such that for any $\varepsilon' > 0$ and any $d \geq 3$, any algorithm that distinguishes correctly with probability $2/3$ between the following two types of n-vertex d-degree bounded graphs must make at least $n^{1/2+\min(1/16, \varepsilon'/(24 \ln d))}$ queries.

- The YES instances: Graphs which have a cut of size at least $(nd/4) \cdot (1 - \varepsilon')$.
- The NO instances: Graphs which do not have a cut of size more than $(1 + \beta d^{-1/2}) \cdot (nd/8)$.

The reduction from **NoisyParities** to MAX-CUT works as follows.

```
procedure ReductionToMAXCUT(Graph $G = (V, E)$, Edge labeling $y : E \longrightarrow \{0, 1\}$)
    $E' = \{e \in E : y(e) = 1\}$.
    return $G' = (V, E')$.
```

We claim that a YES instance of **NoisyParities** is reduced with high probability to a NO instance of MAX-CUT, and vice versa. To prove that the reduction correctly converts a YES instance of

NoisyParities to a NO instance of MAX-CUT, we need the following fact which is implied by Theorem 1.6 of [DMS15].

Fact 2.2. *There exists an absolute constant α such that the following holds for all all d large enough. Let G be a random d-regular n-vertex graph generated from the configuration model. Then with probability $1-o(1)$, the maximum cut in G cuts at most $1/2+\alpha d^{-1/2}$ fraction of the edges.*

Lemma 2.26. *There exists a constant β such that for all d the following holds. Let $G=(V,E)$ be a d-regular φ-expander with $|V|=n$. Suppose each edge $(u,v)\in E$ independently and uniformly given label $Y(u,v)\in\{0,1\}$. Suppose ReductionToMAXCUT on input (G,Y) returns the graph $G'=(V,E')$. Then with probability $1-o(1)$, the maxcut in G' is at most $(1+\beta d^{-1/2})\cdot nd/8$.*

Proof. By Fact 2.2, with probability $1-o(1)$ every cut in G has size at most $(1/2+\alpha d^{-1/2})\cdot nd/2$. Given that every cut in G has size at most $(1/2+\alpha d^{-1/2})\cdot nd/2$, we have the following. Consider an arbitrary cut in G'. The expected number of edges in this cut with label 1 is at most $(1/2+\alpha d^{-1/2})\cdot nd/4=(1+2\alpha d^{-1/2})\cdot nd/8$. By Chernoff bound, the probability that more than $(1+\varepsilon')\cdot(1+2\alpha d^{-1/2})\cdot nd/8$ edges in G' lie in this cut is at most

$$\exp\left(-\frac{\varepsilon'^2\cdot(1+2\alpha d^{-1/2})\cdot nd}{24}\right)\le\exp\left(-\frac{\varepsilon'^2\cdot nd}{24}\right)\le\exp(-n),$$

for $\varepsilon'=24d^{-1/2}$. By union bound over all the 2^n cuts in G', we have that the probability that some cut value exceeds $(1+24d^{-1/2})\cdot(1+2\alpha d^{-1/2})\cdot nd/8\le(1+(24+50\alpha)d^{-1/2})\cdot nd/8$ is at most $(2/e)^n=o(1)$. Adding to this the $o(1)$ probability that G itself has a large cut, and setting $\beta=24+50\alpha$, we get the claim. □

Next, we prove that the reduction correctly converts a NO instance of **NoisyParities** to a YES instance of MAX-CUT, we need the following claim.

Lemma 2.27. *Let $G=(V,E)$ be an arbitrary d-regular graph, and let $\{X(v)\ :\ v\in V\}$ be a set of independent binary random variables, each of which is 0 and 1 with probability $1/2$. Let $V_0=\{v\in V\ :\ X(v)=0\}$ $V_1=\{v\in V\ :\ X(v)=1\}$. Let C be the random variable whose value is the number of edges in the (V_0,V_1) cut. Then Var$[C]\le 2d\cdot\mathbb{E}[C]$.*

Proof. For each $e\in E$, let $C(e)$ be the indicator random variable that is 1 if e lies in the (V_0,V_1) cut, and 0 otherwise. Since $X(v)$'s are independent, for any two edges e and e' which do not share an endpoint, $C(e)$ and $C(e')$ are independent. $C=\sum_{e\in E}C(e)$, and therefore,

$$(\mathbb{E}[C])^2=\sum_{e,e'\in E}\mathbb{E}[C(e)]\mathbb{E}[C(e')]\ge\sum_{e,e'\in E:e\cap e'=\emptyset}\mathbb{E}[C(e)]\mathbb{E}[C(e')]=\sum_{e,e'\in E:e\cap e'=\emptyset}\mathbb{E}[C(e)C(e')].$$

Now, we have

$$\mathbb{E}[C^2]=\sum_{e,e'\in E}\mathbb{E}[C(e)C(e')]=\sum_{e,e'\in E:e\cap e'=\emptyset}\mathbb{E}[C(e)C(e')]+\sum_{e,e'\in E:e\cap e'\neq\emptyset}\mathbb{E}[C(e)C(e')].$$

Using the lower bound on $(\mathbb{E}[C])^2$, we have

$$\mathbb{E}[C^2] \le (\mathbb{E}[C])^2 + \sum_{e,e' \in E : e \cap e' \neq \emptyset} \mathbb{E}[C(e)C(e')],$$

which implies,

$$\mathrm{Var}[C] = \mathbb{E}[C^2] - (\mathbb{E}[C])^2 \le \sum_{e,e' \in E : e \cap e' \neq \emptyset} \mathbb{E}[C(e)C(e')] \le \sum_{e \in E} \mathbb{E}[C(e)] \cdot |\{e' \in E \,:\, e \cap e' \neq \emptyset\}|$$

where we used in the last inequality that $C(e') \le 1$ for any e'. Using the fact that the graph is d-regular, we have that $|\{e' \in E \,:\, e \cap e' \neq \emptyset\}| \le 2d$ for any e. Therefore,

$$\mathrm{Var}[C] \le 2d \cdot \sum_{e \in E} \mathbb{E}[C(e)] = 2d \cdot \mathbb{E}[C],$$

as required. □

Lemma 2.28. *Let $G = (V, E)$ be a d-regular graph with $|V| = n$. For each $v \in V$, let $X(v)$ be an independent uniformly random bit. For each edge $(u, v) \in E$, let $Z(u, v)$ be an independent random bit which is* 1 *with probability ε and* 0 *otherwise. Suppose each edge $(u, v) \in E$ is labeled $Y(u, v) = X(u) + X(v) + Z(u, v)$. Suppose ReductionToMAXCUT on input (G, Y) returns the graph $G' = (V, E')$. Then with probability at least $1 - o(1)$, there exists a cut (V_0, V_1) in G' of value at least $(nd/4) \cdot (1 - 2\varepsilon) \cdot (1 - o(1))$.*

Proof. Let $V_0 = \{v \in V \,:\, X(v) = 0\}$ and $V_1 = \{v \in V \,:\, X(v) = 1\}$, as in the statement of Lemma 2.27, and let C be the random variable whose value is the number of edges of G in the (V_0, V_1) cut. Then $\mathbb{E}[C] = nd/4$, since $|E| = nd/2$ and each edge is cut with probability $1/2$. By Chebyshev's inequality, we have,

$$\Pr\left[|C - \mathbb{E}[C]| > \frac{\mathbb{E}[C]}{n^{1/4}}\right] \le \frac{\mathrm{Var}[C] \cdot n^{1/2}}{(\mathbb{E}[C])^2} \le \frac{2d \cdot \mathbb{E}[C] \cdot n^{1/2}}{(\mathbb{E}[C])^2} = \frac{8dn^{1/2}}{nd} = \frac{8}{n^{1/2}},$$

where the second inequality follows from Lemma 2.27. Therefore, with probability at least $1 - 8/n^{1/2}$, we have $C \ge \mathbb{E}[C](1 - n^{-1/4}) = (nd/4) \cdot (1 - n^{-1/4})$.

Now, let us condition on the values of $X(v)$'s which ensure $C \ge \mathbb{E}[C](1 - n^{-1/4})$. Then the cut (V_0, V_1) is fixed, and the labels on the edges become independent. Each edge (u, v) of G in the (V_0, V_1) cut has label 0 with probability ε and 1 with probability $1 - \varepsilon$. By the Chernoff bound, with probability $1 - \exp(-\varepsilon C/3) \ge 1 - \exp(-(\varepsilon nd/4) \cdot (1 - n^{-1/2}))$, at most $2\varepsilon C$ out of the C edges of G in the (V_0, V_1) cut have label 0, and therefore, at least $(1 - 2\varepsilon)C$ edges have label 1. All these edges with label 1 appear in the (V_0, V_1) cut of G'. Thus, with probability at least $1 - 8/n^{1/2} - \exp(-(\varepsilon nd/4) \cdot (1 - n^{-1/2})) = 1 - o(1)$, G' contains a cut of size at least $(nd/4) \cdot (1 - n^{-1/4}) \cdot (1 - 2\varepsilon) = (nd/4) \cdot (1 - 2\varepsilon) \cdot (1 - o(1))$. □

Proof of Theorem 2.6. Consider an algorithm that approximates maxcut within a factor $2 - \varepsilon'$

with probability at least 2/3, and assume it makes $n^{1/2+\delta}$ queries. Then for any distribution over the instances, there exists a deterministic algorithm ALG making $n^{1/2+\delta}$ queries and having the same approximation guarantee on a random instance drawn from the distribution.

Let (G, y) be a random instance of the **NoisyParities** problem with parameters $\varepsilon = \varepsilon'/24$ and $d = (4\beta/\varepsilon')^2$, where β is the constant from Lemma 2.26. Here, $G = (V, E)$ is a graph and $y : E \longrightarrow \{0, 1\}$ is an edge lebeling. Apply ReductionToMAXCUT to (G, y), and thus, get a random instance G' of MAX-CUT from the appropriate probability distribution. Run ALG on this instance and obtain an estimate z of the maxcut. Return YES if $z > (1+\varepsilon'/4)nd/8$, otherwise return NO. Note that to answer one query of ALG, we make one query into G. Thus, this reduction gives an algorithm ALG$'$ for **NoisyParities** making at most $n^{1/2+\delta}$ queries.

If (G, y) is a YES instance of the **NoisyParities** problem, then by Lemma 2.26, with probability $1 - o(1)$, G' has maxcut at most $(1+\beta d^{-1/2}) \cdot (nd/8) = (1+\varepsilon/4) \cdot (nd/8)$. Thus, the estimate of maxcut given by ALG is at most $(1+\varepsilon/4) \cdot (nd/8)$ with probability at least 2/3, and we return YES. On the other hand, if (G, y) is a NO instance of the **NoisyParities** problem, then by Lemma 2.28, with probability $1 - o(1)$, G' has maxcut at least

$$\frac{nd}{4} \cdot (1-2\varepsilon) \cdot (1-o(1)) \geq (1-3\varepsilon) \cdot \frac{nd}{4} = \left(1 - \frac{\varepsilon'}{8}\right) \cdot \frac{nd}{4}.$$

Therefore, the estimate of maxcut given by ALG, with probability at least 2/3, is at least

$$\frac{1-\varepsilon'/8}{2-\varepsilon'}\frac{nd}{4} = \frac{1-\varepsilon'/8}{1-\varepsilon'/2}\frac{nd}{8} > \left(1-\frac{\varepsilon'}{8}\right)\left(1+\frac{\varepsilon'}{2}\right) \cdot \frac{nd}{8} > \left(1+\frac{\varepsilon'}{4}\right) \cdot \frac{nd}{8},$$

and we return NO. Thus, ALG' is correct with probability at least $2/3 - o(1)$. Therefore, by Theorem 2.4, $\delta \geq \min(1/16, b\varepsilon)$, where $b = 1/(8 \ln d)$. Thus, $\delta = \Omega(\varepsilon'/\log(1/\varepsilon'))$. □

2.4.3 Query lower bound for the noisy parities problem

Recall the **NoisyParities** problem (Definition 2.5). In this section, we prove Theorem 2.4, which gives a lower bound on the query complexity of the **NoisyParities** problem. We start out by formalizing the the execution of the algorithm's query as a process which generates the instance incrementally.

Interaction and Closure

Formally, the interaction that takes place between the algorithm and the adversary is given by the procedure Interaction. Here NextQuery is the function which simulates the behavior of the algorithm: it takes as input the uncovered edge-labeled graph and the set of vertices already queried, and returns an unqueried vertex. It is helpful to make the following observations.

1. The random graph is generated according to the configuration model. As soon as a

vertex q is queried, the unpaired half-edges on q are paired up one-by-one to random unpaired half-edges incident on the yet unqueried vertices.

2. In the YES case, the label generated for any edge is uniformly random, as per the problem definition in Section 2.1. In the NO case, although the label is generated without referring to the parities X of the vertices in the problem specification, the parities are built-in in an implicit manner, and the distribution of the labels is still consistent with the problem specification. Also, this is the only place where the behavior of Interaction differs depending on whether it is executing the YES case or the NO case.

```
procedure INTERACTION(ANSWER, $V$, $\varepsilon$)
    $Q_0 := \emptyset$, $H_0 := \emptyset$, $F_0 := 0$, $T = n^{1/2+\delta}$.
    for $t = 1$ to $T$ do
                        ▷ Invariant: All half-edges incident on vertices in $Q_{t-1}$ are paired.
                        ▷ Invariant: $F_{t-1}$ is a spanning forest of $H_{t-1}$.
                        ▷ Invariant: If Answer = NO then for any cycle $C \subseteq H_{t-1}$, $\sum_{e \in C}(Y(e) + Z(e)) = 0$.
        $q_t :=$ NextQuery($Q_{t-1}, H_{t-1}$).        ▷ Assumption: $q_t \notin Q_{t-1}$.
        $Q_t := Q_{t-1} \cup \{q_t\}$.
        $H_t := H_{t-1}$, $F_t := F_{t-1}$.
        while $q$ has an unpaired half-edge $(q, i)$ do
            Pair up $(q, i)$ with a random unpaired half-edge, say $(v, j)$. Call the resulting edge
    between $q$ and $v$ as $e$.        ▷ $v \notin Q_{t-1}$ unless $v = q$.
            if Answer = YES then
                if $F_t \cup \{e\}$ is acyclic then
                    $F_t := F_t \cup \{e\}$.
                Generate label $Y(e) := 0$ or $1$ with probability $1/2$ each.
            else        ▷ Answer = NO.
                Generate noise $Z(e) := 1$ with probability $\varepsilon$ and $0$ with probability $1 - \varepsilon$.
                if $F_t \cup \{e\}$ is acyclic then
                    $F_t := F_t \cup \{e\}$.
                    Generate label $Y(e) := 0$ or $1$ with probability $1/2$ each.
                else        ▷ $F_t \cup \{e\}$ contains a single cycle.
                    Let $P$ be the unique path from $q$ to $v$ in $F_t$.
                    Generate label $Y(e) := Z(e) + \sum_{e' \in P}(Y(e') + Z(e'))$.
            $H_t := H_t \cup \{(e, Y(e))\}$.
```

Recall that our goal is to prove that the algorithm does not gain sufficient information with $n^{1/2+\delta}$ queries to distinguish between the YES case and the NO case. In order to facilitate our analysis, we give the following additional information to the algorithm for free, and refer to this modified version of Interaction as InteractionWithClosure, and then argue that the algorithm fails nonetheless.

1. InteractionWithClosure ensures that the pairwise distance in F between vertices on which a non-forest edge is incident is at least $b \ln n$, where $b = 1/(8 \ln d)$, as defined in

Theorem 2.4. As soon as the algorithm manages to uncover a new edge resulting in violation of this invariant, InteractionWithClosure throws an error and conservatively assumes that the algorithm found the correct answer already. (In particular, this includes the scenarios where self-loops and parallel edges are discovered.) We say that event Err_t happened if InteractionWithClosure throws an error in round t.

2. As soon as a new edge incident on the queried vertex q_t that cannot be added to F_{t-1} is discovered, InteractionWithClosure generates the whole ball of radius $b \ln n$ around q_t. This might already result in the event Err_t as defined above. If not, InteractionWithClosure adds the BFS tree around q to F_t, labels its edges uniformly at random, samples and records the noise for its edges, adds these labeled edges to H_t, and gives H_t back to the algorithm.

Analysis of InteractionWithClosure

Definition 2.11. After any round t of InteractionWithClosure, we call a vertex $v \in V$ *discovered* if it was queried (that is, $v \in Q_t$) or if one of its neighbors in G was queried (that is, it had degree at least one in H_t). We denote by D_t the set of vertices discovered after round t.

We now define certain "good" events $\mathscr{E}_t$ which are sufficient to ensure that our analysis works and gets us the query lower bound. Moreover, we will also show that these events are extremely likely to happen.

Definition 2.12. Define $R_t = H_t \setminus F_t$ to be the set of non-forest edges seen by the end of round t. Then $R_t \supseteq R_{t-1}$ for all t. For any round t, we say that event $\mathscr{E}_t$ happened if the following conditions hold.

1. InteractionWithClosure completes the t^{th} round without throwing an error, that is, none of the events Err_j for $j \le t$ happen.
2. $|D_j| \le d n^{1/2+\delta} \ln n$ for all $j \le t$.

First, let us prove a bound on the probability that a non-forest edge is found in round t.

Lemma 2.29. *If event $\mathscr{E}_{t-1}$ happens then the probability that InteractionWithClosure encounters an edge in round t which forms a cycle with edges in F_{t-1} is at most $(2d^2 \ln n)/n^{1/2-\delta}$.*

Proof. The number of undiscovered vertices is at least $n - |D_{t-1}| \ge n - d n^{1/2+\delta} \ln n \ge n/2 + 2$, and therefore, there are at least $d(n/2+2)$ free half-edges incident on undiscovered vertices. Therefore, the probability that at least one of the discovered (unqueried) vertices becomes a neighbor of q_t, when we pair up the at most d free half edges incident on q_t, is at most

$$d \cdot \frac{|D_{t-1} \setminus Q_{t-1}| \cdot d}{d(n/2+2) - 2d} \le \frac{d \cdot |D_{t-1}|}{n/2} \le \frac{2d^2 n^{1/2+\delta} \ln n}{n} = \frac{2d^2 \ln n}{n^{1/2-\delta}}.$$

□

Our next lemma and proves a bound on the probability that the algorithm will throw an error in round t.

Lemma 2.30. *If event $\mathscr{E}_{t-1}$ happens then the probability that InteractionWithClosure throws an error in round t, that is, event Err_t happens, is at most* $(16d^4 \ln^2 n)/n^{7/8-2\delta}$.

Proof. Event Err_t happens only in the following two cases.

1. InteractionWithClosure encounters an edge (q_t, v) such that the distance between q_t and v in F_{t-1} is at most $b \ln n$.
2. InteractionWithClosure encounters an edge (q_t, v) which forms a cycle with edges in F_{t-1}, and while generating the ball of radius $b \ln n$, it encounters another edge which cannot be added to the forest.

Let us bound the probabilities of the above two events separately. The number of undiscovered vertices is at least

$$n - |D_{t-1}| \geq n - dn^{1/2+\delta} \ln n \geq \frac{n}{2} + 2n^{1/8} + 2 \geq \frac{n}{2} + 2.$$

First, observe that $|B_{H_{t-1}}(q_t, b \ln n)| \leq d^{b \ln n} = n^{b \ln d} = n^{1/8}$, by Proposition 2.1. q_t has at most d free half-edges, the vertices $B_{H_{t-1}}(q_t, b \ln n) \setminus Q_{t-1}$ have at most $dn^{1/8}$ free half-edges, and we have at least $dn/2$ free half-edges every time we pair a half-edge incident on q_t. Thus, the probability of finding a new non-forest edge closing a short cycle, which is same as the probability that at least one of the vertices $B_{H_{t-1}}(q_t, b \ln n) \setminus Q_{t-1}$ gets an edge incident on q_t, is at most

$$\frac{2d^2 n^{1/8}}{dn} = \frac{2d}{n^{7/8}}.$$

Suppose that in round t we find a neighbor v of q_t such that the edge (q_t, v) cannot be added to the forest. Let us construct the breadth-first search tree around q_t of radius $b \ln n$ by taking each vertex already added to the tree at a time, and pairing up its free half-edges. Consider the processing of some such vertex u, and let W be the vertices already added to the BFS tree at the time u is processed. Since $W \subseteq B_G(q, b \ln n)$, $|W| \leq d^{b \ln n} = n^{b \ln d} = n^{1/8}$, and therefore, the number of edges in the BFS tree is at most $n^{1/8}$. Now, the probability that u gets a new edge to some vertex in $W \cup D_{t-1}$ is at most

$$d \cdot \frac{|W \cup D_{t-1}| \cdot d}{d(n/2 + 2n^{1/8} + 2) - 2n^{1/8} - 2d} \leq \frac{2(n^{1/8} + dn^{1/2+\delta} \ln n) \cdot d}{n} . \leq \frac{4d^2 n^{1/2+\delta} \ln n}{n} = \frac{4d^2 \ln n}{n^{1/2-\delta}}.$$

Note that this must happen for some u for InteractionWithClosure to find a non-forest edge close to the edge (q, v) and throw the error. Since the number of such u's is at most

$|B_G(q_t, b\ln n)| \le n^{b\ln d} = n^{1/8}$, the probability that InteractionWithClosure fails is bounded from above by

$$n^{1/8}\cdot\frac{4d^2\ln n}{n^{1/2-\delta}} = \frac{4d^2\ln n}{n^{3/8-\delta}}.$$

The above holds when conditioned on at least one of the vertices in $D_{t-1}\setminus Q_{t-1}$ being a neighbor of q_t. Unconditioning and using Lemma 2.29, we get that the probability that InteractionWithClosure fails due to the second reason above is bounded by

$$\frac{2d^2\ln n}{n^{1/2-\delta}}\cdot\frac{4d^2\ln n}{n^{3/8-\delta}} = \frac{8d^4\ln^2 n}{n^{7/8-2\delta}}.$$

Adding to this the probability of failure due to the first reason specified above, we have that the probability that event Err_t happens is at most

$$\frac{2d}{n^{7/8}} + \frac{8d^4\ln^2 n}{n^{7/8-2\delta}} \le \frac{16d^4\ln^2 n}{n^{7/8-2\delta}}.$$

□

The next two lemmas essentially prove that if the event $\mathscr{E}_{t-1}$ happens, then it is very likely that $\mathscr{E}_t$ happens too. We then put together these claims and prove that the event $\mathscr{E}_T$ happens with high probability, where $T = n^{1/2+\delta}$ is the number of queries.

Lemma 2.31. *For every t the following holds: if $\mathscr{E}_{t-1}$ happens, then* $\Pr\left[|R_t| > (4d^2\ln n)\cdot(1+t/n^{1/2-\delta})\right] \le n^{-2d^2/3}$.

Proof. For every $j \le t$, conditioned on $\mathscr{E}_{j-1}$, we have that $|R_j\setminus R_{j-1}|$ is one with probability at most $(2d^2\ln n)/n^{1/2-\delta}$, and zero otherwise, by Lemma 2.29. Let $r_1\ldots,r_t$ be independent Bernoulli random variables, each taking value one with probability $(2d^2\ln n)/n^{1/2-\delta}$, and zero otherwise. Then for each j, $|R_j\setminus R_{j-1}| = |R_j| - |R_{j-1}|$ is stochastically dominated by r_j. Let us use the Chernoff bound to upper bound $\Pr[\sum_{j=1}^t r_j > (4d^2\ln n)\cdot(1+t/n^{1/2-\delta})]$, which will also give an upper bound on $\Pr[|R_j| > (4d^2\ln n)\cdot(1+t/n^{1/2-\delta})]$. For this, observe that $\mathbb{E}[\sum_{j=1}^t r_j] = (2d^2 t\ln n)/n^{1/2-\delta}$.

First, consider the case where $t < n^{1/2-\delta}$. Using Chernoff bound, we have,

$$\begin{aligned}\Pr\left[\sum_{j=1}^t r_j > 4d^2\ln n\right] &\le \Pr\left[\sum_{j=1}^t r_j > \left(1+\frac{n^{1/2-\delta}}{t}\right)\cdot\frac{2d^2 t\ln n}{n^{1/2-\delta}}\right]\\ &\le \exp\left(-\frac{n^{1-2\delta}}{3t^2}\cdot\frac{2d^2 t\ln n}{n^{1/2-\delta}}\right) = \exp\left(-\frac{(2d^2\ln n)\cdot n^{1/2-\delta}}{3t}\right).\end{aligned}$$

Using the upper bound on t, we have

$$\Pr\left[\sum_{j=1}^t r_j > (4d^2\ln n)\cdot\left(1+\frac{t}{n^{1/2-\delta}}\right)\right] \le \Pr\left[\sum_{j=1}^t r_j > 4d^2\ln n\right] \le n^{-2d^2/3}. \qquad (2.9)$$

Next, consider the case where $t \geq n^{1/2-\delta}$. Using the Chernoff bound again, we get,

$$\Pr\left[\sum_{j=1}^{t} r_j > \frac{4d^2 t \ln n}{n^{1/2-\delta}}\right] \leq \exp\left(-\frac{2d^2 t \ln n}{3n^{1/2-\delta}}\right) \leq n^{-2d^2/3}.$$

Thus,

$$\Pr\left[\sum_{j=1}^{t} r_j > (4d^2 \ln n)\cdot\left(1+\frac{t}{n^{1/2-\delta}}\right)\right] \leq \Pr\left[\sum_{j=1}^{t} r_j > \frac{4d^2 t \ln n}{n^{1/2-\delta}}\right] \leq n^{-2d^2/3}. \tag{2.10}$$

Equations (2.9) and (2.10) together imply that $\Pr\left[|R_t| > (4d^2 \ln n)\cdot(1+t/n^{1/2-\delta})\right] \leq n^{-2d^2/3}$.
□

Lemma 2.32. *For every t, if $\mathscr{E}_{t-1}$ happens and moreover, if $|R_t| \leq (4d^2 \ln n)\cdot(1+t/n^{1/2-\delta})$, then $|D_t| \leq dn^{1/2+\delta}\ln n$.*

Proof. If $\mathscr{E}_{t-1}$ happens then every round $j \leq t$ which did not discover a non-forest edge (that is, $|R_j| = |R_{j-1}|$) discovered at most d new vertices. On the other hand, every round $j \leq t$ which discovered a new non-forest edge (that is, $|R_j| = |R_{j-1}|+1$) discovered at most $n^{b\ln d} = n^{1/8}$ new vertices, as it discovered $B_G(q_j, b\ln n)$, whose size is at most $n^{b\ln d}$, by Proposition 2.1. Therefore,

$$|D_t| \leq dt + n^{1/8}\cdot|R_t| \leq dt + n^{1/8}\cdot(4d^2\ln n)\cdot(1+t/n^{1/2-\delta}).$$

Since $t \leq n^{1/2+\delta}$ and $\delta < 1/16$, we have,

$$|D_t| \leq dn^{1/2+\delta} + 4d^2 n^{1/8}\ln^2 n\cdot(1+n^{2\delta}) \leq dn^{1/2+\delta}\ln n.$$

□

Lemma 2.33. *The event $\mathscr{E}_T$ happens with probability $1-o(1)$.*

Proof. Let p_t^{err} be the probability that event Err_t happens. Then we prove by induction that there is an absolute constant c such that for each t, event $\mathscr{E}_t$ happens with probability at least

$$1 - \frac{32d^4\cdot\ln^2 n}{n^{7/8-2\delta}}\cdot t.$$

The claim is obvious for $t=0$. For $t>0$, let us upper bound the probability that $\mathscr{E}_t$ does not happen, given $\mathscr{E}_{t-1}$ happens. The reasons for $\mathscr{E}_t$ not happening are the following.

1. Event Err_t happens. This happens with probability p_t^{err}.

2. $|R_t| > (4d^2\ln n)\cdot(1+t/n^{1/2-\delta})$. (If $|R_t| \leq (4d^2\ln n)\cdot(1+t/n^{1/2-\delta})$ then $dn^{1/2+\delta}\ln n$ is guaranteed by Lemma 2.32.)

By Lemma 2.30, the probability that InteractionWithClosure throws an error in round t is at most

$$p_t^{\text{err}} \le \frac{16d^4 \cdot \ln^2 n}{n^{7/8-2\delta}}.$$

By Lemma 2.31, the event $|R_t| > (4d^2 \ln n) \cdot (1 + t/n^{1/2-\delta})$ happens with probability at most $n^{-2d^2/3} < n^{-6}$, because we assumed $d \ge 3$ in the definition of the **NoisyParities** problem. By induction hypothesis, $\mathcal{E}_{t-1}$ itself happens with probability at least

$$1 - \frac{32d^4 \cdot \ln^2 n}{n^{7/8-2\delta}} \cdot (t-1).$$

Thus, $\mathcal{E}_t$ happens with probability at least

$$1 - \frac{32d^4 \cdot \ln^2 n}{n^{7/8-2\delta}} \cdot (t-1) - \frac{16d^4 \cdot \ln^2 n}{n^{7/8-2\delta}} - n^{-6} \ge 1 - \frac{32d^4 \cdot \ln^2 n}{n^{7/8-2\delta}} \cdot t,$$

as required.

As a consequence, the event $\mathcal{E}_T$ happens with probability at least

$$1 - \frac{32d^4 \cdot \ln^2 n}{n^{7/8-2\delta}} \cdot T \ge 1 - \frac{32d^4 \cdot \ln^2 n}{n^{7/8-2\delta}} \cdot n^{1/2+\delta} \ge 1 - \frac{32d^4 \cdot \ln^2 n}{n^{3/8-3\delta}}.$$

Using the fact $\delta < 1/16$, we conclude that $\mathcal{E}_T$ happens with probability $1 - o(1)$. □

Bounding TVD in each Round

Recall that our goal is to prove Theorem 2.4, which states that an algorithm which makes at most $n^{1/2+\delta}$ queries is unable to determine whether InteractionWithClosure is executing the YES or the NO case, assuming δ is less than some constant times ε. For this, we crucially use Corollary 2.2 as follows. The random variable X_t consists of the t^{th} query of the algorithm and its result in a YES instance, whereas the random variable X'_t consists of the t^{th} query of the algorithm and its result in a NO instance. Thus, the realization of the random variable $(X_1, \dots, X_t)$ (resp. $(X'_1, \dots, X'_t)$) captures the snapshot of the run of InteractionWithClosure until the t^{th} query in the YES (resp. NO) case. The events $\mathcal{E}_t$ are as defined in Definition 2.12, and they satisfy the requirements of Corollary 2.2.

Our goal is to prove that if $T \le n^{1/2+\delta}$, then

$$\text{TVD}((X_1, \dots, X_T), (X'_1, \dots, X'_T)) = o(1). \tag{2.11}$$

Since the answer of the algorithm is a function of the realization of $(X_1, \dots, X_T)$ (resp. $(X'_1, \dots, X'_T)$) in the YES (resp. NO) case, the above statement implies that the total variation distance between the algorithm's answer in the YES case and the algorithm's answer in the NO case is only $o(1)$. Therefore, the algorithm's answer is correct with probability $1/2 + o(1)$.

In order to establish (2.11), by Corollary 2.2, it is sufficient to prove that

$$\sum_{t=1}^{T} \sum_{(s_1,\ldots,s_{t-1})\in\mathscr{E}_{t-1}} \Pr\left[\bigwedge_{j=1}^{t-1} X_j = s_j\right] \cdot \mathrm{TVD}\left(\left(X_t \mid \bigwedge_{j=1}^{t-1} X_j = s_j\right), \left(X'_t \mid \bigwedge_{j=1}^{t-1} X'_j = s_j\right)\right) + \Pr[(s_1,\ldots,s_T)\notin\mathscr{E}_T] = o(1) \tag{2.12}$$

Here, we already proved in Lemma 2.33 that $\Pr[(s_1,\ldots,s_T)\notin\mathscr{E}_T] = o(1)$. Therefore, it is sufficient to prove that

$$\sum_{t=1}^{T} \sum_{(s_1,\ldots,s_{t-1})\in\mathscr{E}_{t-1}} \Pr\left[\bigwedge_{j=1}^{t-1} X_j = s_j\right] \cdot \mathrm{TVD}\left(\left(X_t \mid \bigwedge_{j=1}^{t-1} X_j = s_j\right), \left(X'_t \mid \bigwedge_{j=1}^{t-1} X'_j = s_j\right)\right) = o(1). \tag{2.13}$$

Informally, the above claim states the following. Suppose InteractionWithClosure executes on a YES instance and a NO instance in parallel, and for the first $t-1$ rounds of these executions, the queries and the responses to the queries match. Then the probability distributions of the responses to the query in the t^{th} round are $o(1)$-close in total variation distance.

Recall that the executions of InteractionWithClosure on YES and NO instances differ only in the following situation: the current edge whose label is to be generated forms a cycle with edges in F. Therefore, in such a situation, if the forced label in the NO case does not match the uniformly random label in the YES case, then this is responsible for some TVD between X_t and X'_t conditioned on the snapshot of the run of InteractionWithClosure until round $t-1$. Apart from this step, the executions of InteractionWithClosure in the YES case and the NO case are identical. Moreover, the event $\mathscr{E}_T$ ensures that the number of rounds in which an edge closing a cycle is encountered is at most $O(d^2 n^{2\delta}\ln n)$. Therefore, it is sufficient to prove that for all $t \le T$ and for all $(s_1,\ldots,s_{t-1}) \in \mathscr{E}_{t-1}$, we have

$$\mathrm{TVD}\left(\left(X_t \mid \bigwedge_{j=1}^{t-1} X_j = s_j\right), \left(X'_t \mid \bigwedge_{j=1}^{t-1} X'_j = s_j\right)\right) = o(n^{-2b\varepsilon}), \tag{2.14}$$

where $b = 1/(8\ln d)$, as defined earlier. From this, as long as $\delta < b\varepsilon$, (2.13) follows. We devote the rest of this subsection to prove claim (2.14).

Let e be an edge such that when e arrives, the forest F maintained by InteractionWithClosure already contains a path P between the endpoints of e. In the YES case, the label $Y(e)$ of e is 0 or 1 uniformly at random, whereas in the NO case, the label is $Z(e) + \sum_{e'\in P}(Y(e') + Z(e'))$. We are, therefore, interested in bounding the TVD between the distribution of $Z(e) + \sum_{e'\in P}(Y(e') + Z(e'))$ conditioned on the labels of the previous edges, and the uniform distribution on $\{0,1\}$. Since we are conditioning on the labels of all the previous edges, inclusive of edges $e' \in P$, this distance is same as the TVD between the distribution of $Z(e) + \sum_{e'\in P} Z(e')$ conditioned on the labels, and the uniform distribution. Furthermore, observe that Z_e itself is independent of the labels of the previous edges, and is 1 with probability $\varepsilon < 1/2$ and 0 otherwise. Therefore, the TVD between $Z(e) + \sum_{e'\in P} Z(e')$ conditioned on the labels and the uniform distribution is at most the TVD between $\sum_{e'\in P} Z(e')$ conditioned on the labels and the uniform distribution.

In order to bound the TVD between $\sum_{e'\in P} Z(e')$ conditioned on the labels and the uniform distribution, we need to determine the distribution of $\sum_{e'\in P} Z(e')$ conditioned on the labels in the first place. We use the Fourier transform to achieve this. We use Bayes' rule and write the posterior distribution of Z, conditioned on the labels $Y = y$, as being proportional to the product of the prior distribution of Z and the probability of labels Y being y conditioned on Z. Then we use the convolution theorem to get the Fourier transform of the posterior distribution of Z. An appropriate Fourier coefficient then gives us the *bias* of $\sum_{e'\in P} Z(e')$ conditioned on the labels.

Definition 2.13. The *bias* of a binary random variable X is the TVD between its distribution and the uniform distribution over $\{0,1\}$. Equivalently, the bias of X is equal to $|\Pr[X=0]-1/2| = |\Pr[X=1]-1/2|$.

Let $H=(V,E_H)$ be the graph formed by the edges which arrived before e. Let F_H be the forest maintained by $\mathsf{InteractionWithClosure}$ when e arrives (so that F_H is a spanning forest of H). Let the random variable Y and Z, both taking values in $\{0,1\}^{E_H}$, denote the random labels and the random noise of the edges in E_H respectively. Fix $y \in \{0,1\}^{E_H}$. We are interested in the distribution of $Y(e)$, the label on edge e, conditioned on $Y=y$. We want to prove that if the graph H and the spanning forest F_H satisfy certain properties, then the distribution of $Y(e)$ conditioned on $Y=y$ is close to uniform.

Theorem 2.12. *Suppose the graph H and the spanning forest F_H are such that the endpoints of the edges in $(E_H \cup \{e\}) \setminus F$ are pairwise at least a distance Δ apart in F_H. Then the bias of the distribution of the NO-case label of the new edge e conditioned on the labels of the previous edges is at most*

$$\frac{(1-2\varepsilon)^{\Delta-1}(1+(1-2\varepsilon)^{\Delta})^{|E_H\setminus F_H|}}{2-(1+(1-2\varepsilon)^{\Delta})^{|E_H\setminus F_H|}}.$$

Proof. Let P be the path in F_H between the endpoints of e, and let $C = P\cup\{e\}$ be the cycle in $F\cup\{e\}$. Since $Y(e) = Z(e) + \sum_{e'\in P}(Y(e')+Z(e'))$, the distribution of $Y(e)$ conditioned on $Y=y$, has the same bias as the distribution of $Z(e)+\sum_{e'\in P} Z(e') = \sum_{e'\in C} Z(e)$ conditioned on $Y=y$. Here $Z(e)$ is independent of the previous labels Y, and hence, the bias of $\sum_{e'\in C} Z(e')$ conditioned on $Y=y$ is at most the bias of $\sum_{e'\in P} Z(e')$ conditioned on $Y=y$. It is, therefore, sufficient to bound from above the bias of $\sum_{e'\in P} Z(e')$ conditioned on $Y=y$.

The posterior distribution of the random noise Z given the labels $Y=y$ is given by

$$\Pr[Z=z \mid Y=y] = \frac{\Pr[Y=y \mid Z=z]\cdot\Pr[Z=z]}{\Pr[Y=y]} = \frac{f(z)\cdot g_y(z)}{\sum_{z'\in\{0,1\}^{E_{t-1}}} f(z')\cdot g_y(z')} = \frac{h_y(z)}{\sum_{z'\in\{0,1\}^{E_{t-1}}} h_y(z')},$$

where the functions f, g_y and h_y are defined as $f(z)=\Pr[Z=z]$, $g_y(z)=\Pr[Y=y \mid Z=z]$, and $h_y = f\cdot g_y$. The Fourier transforms of these functions are as follows. Since $f(z)=\varepsilon^{|z|}(1-\varepsilon)^{|E_H|-|z|}$, by Proposition 2.2, we have for all $\alpha\in\{0,1\}^{E_H}$,

$$\widehat{f}(\alpha) = 2^{-|E_H|}(1-2\varepsilon)^{|\alpha|}.$$

Next let us consider the function g_y. Let E^* denote the nullspace of the incidence matrix of H_{t-1}, that is, the set of indicator vectors of Eulerian subgraphs of H_{t-1}. We say that y and z are compatible if for every cycle C in H_{t-1} we have $\sum_{e\in C} y(e) = \sum_{e\in C} z(e)$, that is, $\gamma\cdot y = \gamma\cdot z$ for all $\gamma\in E^*$. Since the dimension of E^* is $|E_H| - |F_H|$, there are exactly $2^{|F_H|}$ compatible y's for every z, one for each of the $2^{|F_H|}$ labelings of edges in F. Moreover, each of the $2^{|F_H|}$ labelings are realized with equal probability, because the edges of F are labeled independently with a 0 or a 1 with probability 1/2 each. Therefore we have,

$$g_y(z) = \begin{cases} 2^{-|F_H|} & \text{if } \gamma\cdot y = \gamma\cdot z \text{ for all } \gamma\in E^* \\ 0 & \text{otherwise.}\end{cases}$$

Then by Proposition 2.3, the Fourier transform of g_y is given by

$$\widehat{g}_y(\alpha) \quad = \quad \begin{cases} 2^{-|E_H|}(-1)^{\alpha\cdot y} & \text{if } \alpha\in E^* \\ 0 & \text{otherwise.}\end{cases}$$

Using convolution theorem (Proposition 2.4), we have,

$$\widehat{h}_y(\alpha) = \sum_{\beta\in\{0,1\}^{E_H}} \widehat{g}_y(\beta)\widehat{f}(\alpha+\beta) = 2^{-2|E_H|}\sum_{\beta\in E^*}(-1)^{\beta\cdot y}(1-2\varepsilon)^{|\alpha+\beta|}.$$

Recall that our goal was to bound the bias of $\sum_{e'\in P} Z(e')$ conditioned on the labels y, where e is the edge whose label is being generated, and P is the unique path in F between the endpoints of e. Let $\pi\in\{0,1\}^{E_H}$ be the indicator vector of P. Then the bias of $\sum_{e'\in P} Z(e') = \pi\cdot Z$ conditioned on y is expressed as follows.

$$\begin{aligned} \widehat{h}_y(\pi) &= 2^{-|E_H|}\sum_{z\in\{0,1\}^{E_{t-1}}} h_y(z)(-1)^{\pi\cdot z}, \\ \widehat{h}_y(0) &= 2^{-|E_H|}\sum_{z\in\{0,1\}^{E_{t-1}}} h_y(z). \end{aligned}$$

Therefore,

$$\begin{aligned} \frac{|\widehat{h}_y(\pi)|}{\widehat{h}_y(0)} &= \frac{1}{\sum_{z'\in\{0,1\}^{E_{t-1}}} h_y(z')}\cdot\left|\sum_{z\in\{0,1\}^{E_{t-1}},\,\pi\cdot z=0} h_y(z) - \sum_{z\in\{0,1\}^{E_{t-1}},\,\pi\cdot z=0} h_y(z)\right| \\ &= |\Pr[\pi\cdot Z = 0 \mid Y = y] - \Pr[\pi\cdot Z = 1 \mid Y = y]| \\ &= \mathrm{bias}(\pi\cdot Z \mid Y = y). \end{aligned}$$

It is thus sufficient to upper bound $|\widehat{h}_y(\pi)|/\widehat{h}_y(0)$. We now bound $|\widehat{h}_y(\pi)|$ and $\widehat{h}_y(0)$ separately. We have

$$\widehat{h}_y(\pi) = 2^{-2|E_H|}\sum_{\beta\in E^*}(-1)^{\beta\cdot y}(1-2\varepsilon)^{\beta+\pi} \le 2^{-2|E_H|}\sum_{\beta\in E^*}(1-2\varepsilon)^{\beta+\pi}.$$

For $\beta\in E^*$, the indicator vector of an Eulerian subgraph of H, consider the set P' of edges whose indicator vector is $\beta+\pi$. Then $P'\cup\{e\}$ is Eulerian. Conversely, if $P'\cup\{e\}$ is Eulerian for some $P'\subseteq E_H$, then its indicator vector is $\beta+\pi$ for some $\beta\in E^*$. Since we assumed that the endpoints of the edges in $(E_H\cup\{e\})\setminus F_H$ are pairwise at least a distance Δ apart in F_H, by Lemma 2.22, we have

$$|P'\cup\{e\}|\geq\Delta|(P'\cup\{e\})\setminus F_H|=\Delta(|P'\setminus F_H|+1).$$

For any $\gamma\in\{0,1\}^{E_H}$, let $\overline{\gamma}$ denote the projection of β onto the span of the indicator vectors of the edges not in F. Then the above statement can be rewritten as,

$$|\beta+\pi|+1\geq\Delta(|\overline{\beta+\pi}|+1)=\Delta(|\overline{\beta}|+1),$$

where the last equality holds because π, being the indicator vector of a path in F_H, has zero projection onto the span of the indicator vectors of the edges not in F. Therefore,

$$\widehat{h}_y(\pi)\leq 2^{-2|E_H|}\sum_{\beta\in E^*}(1-2\varepsilon)^{\Delta(|\overline{\beta}|+1)-1}=2^{-2|E_H|}(1-2\varepsilon)^{\Delta-1}\sum_{\beta\in E^*}(1-2\varepsilon)^{\Delta|\overline{\beta}|}.$$

By Lemma 2.21, as β varies over the indicator vectors of Eulerian subgraphs of H, its projection $\overline{\beta}$ varies over $\{0,1\}^{E_H\setminus F_H}$. Therefore,

$$\widehat{h}_y(\pi)\leq 2^{-2|E_H|}(1-2\varepsilon)^{\Delta-1}\sum_{\overline{\beta}\in\{0,1\}^{E_H\setminus F_H}}(1-2\varepsilon)^{\Delta|\overline{\beta}|}=2^{-2|E_H|}(1-2\varepsilon)^{\Delta-1}(1+(1-2\varepsilon)^{\Delta})^{|E_H\setminus F_H|}.$$

We also have,

$$\widehat{h}_y(0)=2^{-2|E_H|}\sum_{\beta\in E^*}(-1)^{\beta\cdot y}(1-2\varepsilon)^{|\beta|}\geq 2^{-2|E_H|}\left(2-\sum_{\beta\in E^*}(1-2\varepsilon)^{|\beta|}\right).$$

Again, by Lemma 2.22 we have $|\beta|\geq\Delta\cdot|\overline{\beta}|$. Therefore,

$$\widehat{h}_y(0)\geq 2^{-2|E_H|}\left(2-\sum_{\beta\in E^*}(1-2\varepsilon)^{\Delta\cdot|\overline{\beta}|}\right).$$

As before, as β varies over the indicator vectors of Eulerian subgraphs of H, its projection $\overline{\beta}$ varies over $\{0,1\}^{E_H\setminus F_H}$. Therefore,

$$\widehat{h}_y(0)\geq 2^{-2|E_H|}\left(2-\sum_{\overline{\beta}\in\{0,1\}^{E_H\setminus F_H}}(1-2\varepsilon)^{\Delta\cdot|\overline{\beta}|}\right)=2^{-2|E_H|}\left(2-(1+(1-2\varepsilon)^{\Delta})^{|E_H\setminus F_H|}\right).$$

The upper bound on $|\widehat{h}_y(P)|$ and the lower bound on $\widehat{h}_y(\emptyset)$ together imply

$$\operatorname{bias}\left(\sum_{e'\in P}Z(e')\ \middle|\ Y=y\right)=\frac{|\widehat{h}_y(P)|}{\widehat{h}_y(\emptyset)}\leq\frac{(1-2\varepsilon)^{\Delta-1}(1+(1-2\varepsilon)^{\Delta})^{|E_H\setminus F_H|}}{2-(1+(1-2\varepsilon)^{\Delta})^{|E_H\setminus F_H|}}.$$

□

Wrapping Up

Proof of Theorem 2.4. As a consequence of Lemma 2.33 and Lemma 2.31, at any point of time during the execution of InteractionWithClosure, with probability $1-o(1)$ we have that the endpoints of the edges in $E_H \setminus F_H$ are pairwise separated in F_H by a distance at least $b \ln n$, and moreover,

$$|E_H \setminus F_H| = |R| \leq (4d^2 \ln n) \cdot (1 + t/n^{1/2-\delta}) = O(d^2 n^{2\delta} \ln n),$$

because $t \leq n^{1/2+2\delta}$. At the time of labeling a new edge e which forms a cycle with edges in F, let us apply Theorem 2.12, with $\Delta = b \ln n$. This gives that the bias in the label of e in the NO case is at most

$$\frac{(1-2\varepsilon)^{\Delta-1}(1+(1-2\varepsilon)^{\Delta})^{|E_H \setminus F_H|}}{2-(1+(1-2\varepsilon)^{\Delta})^{|E_H \setminus F_H|}} = \frac{(1-2\varepsilon)^{b\ln n-1}(1+(1-2\varepsilon)^{b\ln n})^{|E_H \setminus F_H|}}{2-(1+(1-2\varepsilon)^{b\ln n})^{|E_H \setminus F_H|}}.$$

Since $|E_H \setminus F_H| = O(d^2 n^{2\delta} \ln n)$, we have

$$(1+(1-2\varepsilon)^{b\ln n})^{|E_H \setminus F_H|} \leq (1+n^{-2b\varepsilon})^{|E_H \setminus F_H|} \leq (1+n^{-2b\varepsilon})^{O(d^2 n^{2\delta} \ln n)} = 1+o(1),$$

because $\delta < b\varepsilon$. Therefore, the bias in the label of e in the NO case is at most

$$\frac{n^{-2b\varepsilon}}{1-2\varepsilon} \cdot (1+o(1)) = O(n^{-2b\varepsilon}).$$

This is the required bound on the TVD between the snapshots of the executions of InteractionWithClosure in the YES and the NO case, in a generic round, given that the snapshots until the end of the previous round were the same. This proves claim (2.14), and hence, Theorem 2.4. □

2.5 Clusterability in bounded degree graphs

In this section we solve the **Clusterability** problem for bounded degree graphs using our **PartitionTesting** algorithm. We first start by stating the definitions. Notice that we change our notion of conductance and ε-closeness to be same as [CPS15] to ensure that we can apply lemmas from that paper. In particular, these definitions are different from the ones given in Section 2.1.1, which our **PartitionTesting** primitive uses. However, given a graph G with vertex degrees bounded by d, one can easily convert G implicitly into a graph G' such that volumes and conductances in G' under our definition from Section 2.1.1 are identical to volumes and conductances under the definition of [CPS15]. This transformation is simply the operation of adding an appropriate number of self-loops to every node, and can hence be done implicitly, allowing us to use our algorithm for **PartitionTesting** on G' to test clusterability in G. We now give the definitions.

We are given a degree d-bounded graph $G = (V_G, E_G)$ on n vertices with m edges. For any vertex $v \in V_G$, we denote its degree in G by $\deg(v)$. For any vertex set $V' \subseteq V_G$, we denote by $G[V']$ the subgraph of G induced by V'. Given a pair of disjoint sets $A, B \subseteq V_G$, we define $E_G(A,B) = E_G \cap (A \times B)$. The internal and external conductance parameters of (subsets of vertices of) G are defined as follows.

Definition 2.14. For a set $S \subseteq C \subseteq V_G$, the *conductance of S within C*, denoted by $\Phi_C^G(S)$, is $\frac{E_G[S, C \setminus S]}{d \cdot |S|}$.

Definition 2.15. The *internal conductance* of $C \subseteq V_G$, denoted by $\Phi^G(C)$, is defined to be $\min_{S \subseteq C, 0 < |S| \le \frac{|C|}{2}} \Phi_C^G(S)$ if $|C| > 1$ and one otherwise. The *conductance* of G is $\Phi(G) = \Phi^G(V_G)$. We say that C has conductance at least φ, or equivalently that it is a *φ-expander* if $\Phi^G(C) \ge \varphi$. The *external conductance* of C is defined to be $\Phi_{V_G}^G(C)$.

Based on the conductance parameters, clusterability and far from clusterability is defined as follows.

Definition 2.16. [**Bounded degree graph clusterability**] For a degree d-bounded graph $G = (V_G, E_G)$ with n vertices, we say that G is *(k,φ)-bounded-degree-clusterable* if there exists a partition of V_G into $1 \le h \le k$ sets $C_1, \cdots, C_h$ such that for each $i = 1, \ldots, h$, $\Phi^G(C_i) \ge \varphi$.

Definition 2.17. A degree d-bounded graph $G = (V_G, E_G)$ with n vertices is *ε-far from (k,φ')-bounded-degree-clusterable* if we need to add or delete more than εdn edges to obtain any *(k,φ')-bounded-degree-clusterable* graph of maximum degree at most d. We say that G is ε-close to *(k,φ')-bounded-degree-clusterable,* if G is not ε-far from *(k,φ')-bounded-degree-clusterable.* We say that G is ε-far from φ'-expander if G is ε-far from *$(1,\varphi')$-bounded-degree-clusterable.*

The goal of this section is to establish Theorem 2.3, and Theorem 2.2, restated here for convenience of the reader. Theorem 2.3 follows as a consequence of our Theorem 2.1, Lemma 5.9, and Lemma 5.10 of [CPS15].

Theorem 2.3 (restated) *Let $0 \le \varepsilon \le \frac{1}{2}$. Suppose $\varphi' \le \alpha$, (for $\alpha = \min\{\frac{c_{exp}}{150d}, \frac{c_{exp} \cdot \varepsilon}{1400 \log(\frac{16k}{\varepsilon})}\}$, where d denotes the maximum degree), and $\varphi' \le c \cdot \varepsilon^2 \varphi^2 / \log(\frac{32k}{\varepsilon})$ for some small constant c. Then there exists a randomized algorithm for* **Clusterability**$(k, \varphi, 2k, \varphi', \varepsilon)$ *problem on degree d-bounded graphs that gives the correct answer with probability at least $2/3$, and which makes $poly(1/\varphi) \cdot poly(k) \cdot poly(1/\varepsilon) \cdot poly(d) \cdot poly\log(n) \cdot n^{1/2 + O(\varepsilon^{-2} \log(\frac{32k}{\varepsilon}) \cdot \varphi'/\varphi^2)}$ queries on graphs with n vertices.*

Theorem 2.3 follows as a consequence of our Theorem 2.1, and Lemma 4.5 of [CPS15].

Theorem 2.2 (restated) *Suppose $\varphi' \le \alpha_{4.5}\varepsilon$, (for the constant $\alpha_{4.5} = \Theta(\min(d^{-1}, k^{-1}))$ from Lemma 4.5 of [CPS15], where d denotes the maximum degree), and $\varphi' \le c' \varepsilon^2 \varphi^2 / k^2$ for some* *small constant c'. Then there exists a randomized algorithm for* **Clusterability**$(k,\varphi,k,\varphi',\varepsilon)$ *problem on degree d-bounded graphs that gives the correct answer with probability at least* $2/3$, *and which makes $poly(1/\varphi)\cdot poly(k)\cdot poly(1/\varepsilon)\cdot poly(d)\cdot poly\log(n)\cdot n^{1/2+O(\varepsilon^{-2}k^2\cdot\varphi'/\varphi^2)}$ queries on graphs with n vertices.*

We will need the following results from [CPS15] to show that the property of being far from being clusterable implies a decomposition into many large sets with small outer conductance.

Lemma 2.34 (Lemma 5.9 of [CPS15])**.** *Let $0<\varphi\le\frac{c_{exp}}{150d}$, and $0<\varepsilon\le\frac{1}{2}$ for some constant c_{exp}. If $G=(V,E)$ is ε-far from any graph H with $\Phi(H)\ge\varphi$, then there is a subset of vertices $A\subseteq V$ with $\frac{\varepsilon}{18}|V|\le|A|\le\frac{1}{2}|V|$ such that $\Phi^G(A)\le\frac{700}{c_{exp}}\cdot\varphi$. In particular, $E_G(A,V\setminus A)\le\frac{700}{c_{exp}}\cdot\varphi\cdot d\cdot|A|$.*

Lemma 2.35 (Lemma 5.10 of [CPS15])**.** *Let $G=(V,E)$ be ε-far from (k,φ)-bounded-degree-clusterable, and $\varphi\le\frac{c_{exp}}{d}$ for some constant c_{exp}. If there is a partition of V into h sets $C_1,\dots,C_h$ with $1\le h\le k$, such that $E[C_1,\dots,C_h]=0$, then there is an index i, $1\le i\le h$, with $|C_i|\ge\frac{\varepsilon}{8}\cdot\frac{|V|}{k}$ such that $G[C_i]$ is $\frac{\varepsilon}{2}$ -far from any H on vertex set C_i with maximum degree d and $\Phi(H)\ge\varphi$.*

We first prove the following lemma and then use it in the proof of Theorem 2.3.

Lemma 2.36. *Let $0\le\varepsilon\le\frac{1}{2}$, $\alpha=\min\{\frac{c_{exp}}{150d},\frac{c_{exp}\cdot\varepsilon}{1400\log(\frac{16k}{\varepsilon})}\}$, and $\varphi\le\alpha$. If $G=(V,E)$ is ε-far from (k,φ)-bounded-degree-clusterable, then there exist a partition of V into $k+1$ subsets $C_1,\dots,C_{k+1}$ such that $E[C_1,\cdots,C_{k+1}]\le\frac{700}{c_{exp}}\varphi\cdot d\cdot|V|\log(\frac{16k}{\varepsilon})$, and for each $1\le i\le k+1$, $|C_i|\ge\frac{\varepsilon^2}{1152}\cdot\frac{|V|}{k}$.*

Proof. Let $n=|V|$. By induction we construct a sequence of partitions $\{C_1^1\}$, $\{C_1^2,C_2^2\}$, $\cdots$, $\{C_1^{k+1},\cdots C_{k+1}^{k+1}\}$ of V such that each partition $\{C_1^h,\cdots,C_h^h\}$ satisfies the following properties:

1. $|C_i^h|\ge\frac{\varepsilon^2}{1152}\cdot\frac{|V|}{k}$ for every i, $1\le i\le h$,
2. $E[C_1^h,\cdots,C_h^h]\le\frac{700}{c_{\text{exp}}}\varphi\cdot d\cdot n\cdot\log(\frac{16k}{\varepsilon})$

The first partition is $\{C_1^1\}=\{V\}$, which satisfies properties (1) and (2). Given a partition $\{C_1^h,\cdots,C_h^h\}$ which satisfies the properties, we construct the partition $\{C_1^{h+1},\cdots,C_{h+1}^{h+1}\}$ as follows.

Let G' be the graph obtained by removing all edges between different subsets C_i^h and C_j^h , $1\le i<j\le h$, from G. Observe that $\varphi\le\frac{1}{2}\frac{\varepsilon}{\frac{700}{c_{\text{exp}}}\log(\frac{16k}{\varepsilon})}$, hence,

$$E[C_1^h,\cdots,C_h^h]\le\frac{700}{c_{\text{exp}}}\varphi\cdot d\cdot n\cdot\log\left(\frac{16k}{\varepsilon}\right)\le\frac{1}{2}\varepsilon\cdot d\cdot n.$$

Therefore G' is $\frac{\varepsilon}{2}$-far from (k,φ)-bounded-degree-clusterable, and thus we can apply Lemma 2.35. Therefore, there is an index i_h, $1\le i_h\le h$, such that $|C_{i_h}^h|\ge\frac{\frac{\varepsilon}{2}}{8}\cdot\frac{|V|}{k}$ and $G'[C_{i_h}^h]$ is $\frac{\varepsilon}{4}$-far from any H on vertex set C_{i_h} with maximum degree d and $\Phi(H)\ge\varphi$. Thus, by Lemma 2.34 there is a set $A_{h+1} \subseteq C^h_{i_h}$ with $\frac{\frac{\varepsilon}{4}}{18}|C^h_{i_h}| \le |A_{h+1}| \le \frac{1}{2}|C^h_{i_h}|$ such that $E[A_{h+1}, C^h_{i_h} \setminus A_{h+1}] \le \frac{700}{c_{\exp}} \cdot \varphi \cdot d \cdot |A_{h+1}|$. Our new partition is $\{C^h_1, \cdots, A_{h+1}, C^h_{i_h} \setminus A_{h+1}, \cdots, C^h_h\}$. Now we prove that the new partition satisfies properties (1) and (2).

Recall that $|C^h_{i_h}| \ge \frac{\varepsilon}{16} \cdot \frac{|V|}{k}$. Thus, we have $|A_{h+1}| \ge \frac{\varepsilon}{4\times 18} \cdot |C^h_{i_h}| \ge \frac{\varepsilon^2}{1152} \cdot \frac{|V|}{k}$ and $|C^h_{i_h} \setminus A_{h+1}| \ge \frac{1}{2}|C^h_{i_h}| \ge \frac{\varepsilon}{32} \cdot \frac{|V|}{k}$. Therefore our new partition satisfies property (1).

In order to prove (2), imagine constructing a rooted decomposition tree T whose vertices corresponds to subsets of vertices in V as follows. The root is the set of all vertices. Whenever a set of vertices C is split into A and $C \setminus A$, we add A and $C \setminus A$ as the left and right child of C respectively. The construction ensures the following.

- (P1) The non-leaf nodes of the tree correspond to sets of vertices of size at least $\frac{\varepsilon n}{16k}$.
- (P2) The size of the set corresponding to the left child is $|A| = \delta|C|$ and the size of the right child is $|C \setminus A| = (1-\delta)|C|$ for some $\frac{\varepsilon}{72} \le \delta \le \frac{1}{2}$.

Whenever a set of vertices C is split into A and $C \setminus A$, we will charge the edges cut in this decomposition step to vertices in A by placing a charge of $\frac{700}{c_{\exp}}\varphi \cdot d$ at each vertex $v \in A$. Clearly, the total charge placed in the vertices in A is an upper bound on the number of edges cut at this step. We now observe that the total number of times any vertex v gets charged in the decomposition process is bounded by $\log(\frac{16k}{\varepsilon})$. This follows from the fact that each time a vertex gets charged, the size of its set decreases by at least a factor 2, and by (P1), the non-leaf nodes has size at least $\frac{\varepsilon n}{16k}$. Thus the total number of edges cut in the decomposition process is bounded by $\frac{700}{c_{\exp}}\varphi \cdot d \cdot n \cdot \log(\frac{16k}{\varepsilon})$. □

Now we are able to prove Theorem 2.3:

Proof of Theorem 2.3. Let $G = (V, E)$ be a degree d-bounded graph with n vertices. We prove that there exists a randomized algorithm for **Clusterability**$(k, \varphi, 2k, \varphi', \varepsilon)$ problem on degree d-bounded graphs that gives the correct answer with probability at least $\frac{2}{3}$.

Let G' be a graph obtained from $G = (V, E)$ by increasing the degree of every $v \in V_G$ by $d - \deg(v)$, by adding self-loops. Observe that for any set $S \subseteq C \subseteq V$, we have $\mathrm{vol}_{G'}(S) = d \cdot |S|$. Hence, $\mathrm{vol}_{G'}(S) \le \frac{\mathrm{vol}_{G'}(V)}{2}$ if and only if $|S| \le \frac{n}{2}$. Moreover note that,

$$\phi^{G'}_C(S) = \frac{E_{G'}[S, C \setminus S]}{\mathrm{vol}_{G'}(S)} = \frac{E_G[S, C \setminus S]}{d \cdot |S|} = \Phi^G_C(S). \tag{2.15}$$

Thus for any $C \subseteq V$ we have,

$$\Phi^G(C) = \min_{S \subseteq C, 0 < |S| \le \frac{|C|}{2}} \Phi^G_C(S) = \min_{S \subseteq C, 0 < \mathrm{vol}(S) \le \frac{\mathrm{vol}(C)}{2}} \phi^{G'}_C(S) = \phi^{G'}(C). \tag{2.16}$$

Now we apply our **PartitionTesting** algorithm to G', and prove that it can distinguish between graphs that are (k,φ)-bounded-degree-clusterable, and those are ε-far from $(2k,\varphi')$-bounded-degree-clusterable with high probability.

Let G be (k,φ)-bounded-degree-clusterable. then there exists a partition of V into sets $C_1,\cdots,C_h$, for $h \le k$ such that for each $i = 1,\dots,h$, $\Phi^G(C_i) \ge \varphi$. Thus by equation (2.16), we have $\phi^{G'}(C_i) \ge \varphi$ for all $i = 1,\dots,h$. Therefore G' is (k,φ)-clusterable. Hence, by Theorem 2.1, **PartitionTesting** algorithm accepts G' with probability at least $\frac{2}{3}$.

Now suppose that G is ε-far from $(2k,\varphi')$-bounded-degree-clusterable. Since $\varphi' < \alpha$, by Lemma 2.36, there exists a partition of V into $2k+1$ subsets $C_1,\dots,C_{2k+1}$ such that $E[C_1,\cdots,C_{2k+1}] \le \frac{700}{c_{\exp}}\varphi\cdot d\cdot|V|\log(\frac{32k}{\varepsilon})$, and for each $1 \le i \le 2k+1$, $|C_i| \ge \frac{\varepsilon^2}{1152}\cdot\frac{|V|}{2k}$.

We say that cluster C_i is bad if $E[C_i, V\setminus C_i] \ge \frac{700}{c_{\exp}}\varphi'\cdot d\cdot\log(\frac{32k}{\varepsilon})\cdot|C_i|$. Define set B as the set of bad clusters i.e., $B = \{C_i : E[C_i, V\setminus C_i] \ge \frac{4\times1152}{\varepsilon^2}\cdot\frac{700}{c_{\exp}}\varphi'\cdot d\cdot\log(\frac{32k}{\varepsilon})\cdot|C_i|\}$. Thus we have,

$$\begin{aligned}
\frac{700}{c_{\exp}}\varphi\cdot d\cdot\log\left(\frac{32k}{\varepsilon}\right)\cdot|V| &\ge E[C_1,\cdots,C_{2k+1}]\\
&\ge \sum_{i=1}^{2k+1} E[C_i, V\setminus C_i]\\
&\ge \sum_{C_i\in B} E[C_i, V\setminus C_i]\\
&\ge \sum_{C_i\in B}\frac{4\times1152}{\varepsilon^2}\cdot\frac{700}{c_{\exp}}\varphi'\cdot d\cdot\log\left(\frac{32k}{\varepsilon}\right)\cdot|C_i|\\
&\ge |B|\cdot\left(\frac{4\times1152}{\varepsilon^2}\cdot\frac{700}{c_{\exp}}\varphi'\cdot d\cdot\log\left(\frac{32k}{\varepsilon}\right)\right)\cdot\left(\frac{\varepsilon^2}{1152}\cdot\frac{|V|}{2k}\right)
\end{aligned}$$

Thus $|B| \le \frac{k}{2}$. Hence there exist at least $k+1$ disjoint sets of vertices $C_1, C_2,\dots,C_{k+1}$, in G such that for $i\in[1..(k+1)]$, $|C_i| \ge \frac{\varepsilon^2}{1152}\cdot\frac{|V|}{2k}$, and $|E(C_i, V\setminus C_i)| \le \frac{4\times1152}{\varepsilon^2}\cdot\frac{700}{c_{\exp}}\varphi'\cdot d\cdot\log\left(\frac{32k}{\varepsilon}\right)\cdot|C_i|$. Thus by equation (2.15), for each i, $1\le i\le k+1$ we have $\mathrm{vol}_{G'}(C_i) \ge \frac{\varepsilon^2}{1152\cdot k}\mathrm{vol}_{G'}(V)$, and $\phi_V^{G'}(C) \le \frac{4\times1152}{\varepsilon^2}\cdot\frac{700}{c_{\exp}}\varphi'\cdot\log(\frac{32k}{\varepsilon})$. Hence, by Definition 2.2, G' is $(k,\varphi_{\text{out}},\beta)$-unclusterable for $\beta = \frac{\varepsilon^2}{1152}$, and $\varphi_{\text{out}} = \frac{4\times1152}{\varepsilon^2}\cdot\frac{700}{c_{\exp}}\varphi'\cdot\log(\frac{32k}{\varepsilon})$. We set $c = \frac{c_{\exp}}{480\times700\times4\times1152}$. Since $\varphi' \le c\cdot\frac{\varepsilon^2\varphi^2}{\log(\frac{32k}{\varepsilon})}$, we have $\varphi_{\text{out}} < \frac{1}{480}\varphi^2$, and hence, we can apply Theorem 2.1. Therefore, **PartitionTesting** algorithm rejects G' with probability at least $\frac{2}{3}$. The running time follows easily from the fact that $m \le d\cdot n$. □

For the proof of Theorem 2.2 we will need the following result from [CPS15] which establish connection between the properties of far from being clusterable, and being unclusterable.

Lemma 2.37. *(Lemma 4.5 of [CPS15]) Let $\alpha_{4.5} = \Theta(\min(d^{-1}, k^{-1}))$ be a certain constant that depends on d and k. If $G = (V,E)$ is ε-far from (k,φ')-degree-bounded-clusterable with $\varphi' \le$*

$\alpha_{4.5}\varepsilon$, then there exist a partition of V into subsets $C_1,\dots,C_{k+1}$ such that for each i, $1 \le i \le k+1$, $|C_i| \ge \frac{\varepsilon^2}{1152\cdot k}|V|$, and $\Phi^G_{V_G}(C) \le \frac{ck^2\varphi'}{\varepsilon^2}$, for some constant c.

Now we are able to prove Theorem 2.2:

Proof of Theorem 2.2. We wish to prove that there exists a randomized algorithm for **Clusterability**$(k,\varphi,k,\varphi',\varepsilon)$ problem on degree d-bounded graphs that gives the correct answer with probability at least $\frac{2}{3}$. Let $G=(V,E)$ be a degree d-bounded graph with n vertices. Let G' be a graph obtained from $G=(V,E)$ by increasing the degree of every $v \in V_G$ by $d-\deg(v)$, by adding self-loops. Now we apply our **PartitionTesting** algorithm to G', and prove that it can distinguish between graphs that are (k,φ)-bounded-degree-clusterable and those are ε-far from (k,φ')-bounded-degree-clusterable, with high probability.

Let G be (k,φ)-bounded-degree-clusterable. Then there exists a partition of V into sets $C_1,\cdots,C_h$, for $h \le k$ such that for each $i=1,\dots,h$, $\Phi^G(C_i) \ge \varphi$. Thus by equation (2.16), we have $\phi^{G'}(C_i) \ge \varphi$ for all $i=1,\dots,h$. Therefore G' is (k,φ)-clusterable. Hence, by Theorem 2.1, **PartitionTesting** algorithm accepts G' with probability at least $\frac{2}{3}$.

Now suppose that G is ε-far from (k,φ')-bounded-degree-clusterable. Since $\varphi' \le \alpha_{4.5}\varepsilon$, by Lemma 2.37, there exist a partition of V into subsets $C_1,\dots,C_{k+1}$ such that for each i, $1 \le i \le k+1$, $|C_i| \ge \frac{\varepsilon^2}{1152\cdot k}|V|$, and $\Phi^G_V(C) \le \frac{ck^2\varphi'}{\varepsilon^2}$ for some constant c. Thus by equation (2.15), for each i, $1 \le i \le k+1$ we have $\mathrm{vol}_{G'}(C_i) \ge \frac{\varepsilon^2}{1152\cdot k}\mathrm{vol}_{G'}(V)$, and $\phi^{G'}_V(C) \le \frac{ck^2\varphi'}{\varepsilon^2}$. Thus by Definition 2.2, G' is $(k,\varphi_{\mathrm{out}},\beta)$-unclusterable for $\beta = \frac{\varepsilon^2}{1152}$ and $\varphi_{\mathrm{out}} = \frac{ck^2\varphi'}{\varepsilon^2}$. We set $c' = \frac{1}{480\cdot c}$. Since $\varphi' \le c'\varepsilon^2\varphi^2/k^2$, we have $\varphi_{\mathrm{out}} < \frac{1}{480}\varphi^2$, hence, we can apply Theorem 2.1. Therefore, **PartitionTesting** algorithm rejects G' with probability at least $\frac{2}{3}$. The running time follows easily from the fact that $m \le d\cdot n$. □

3 Spectral Clustering Oracles in Sublinear Time

This chapter is based on a joint work with Grzegorz Gluch, Michael Kapralov, Silvio Lattanzi and Christian Sohler. It has been accepted to the ACM-SIAM Symposium on Discrete Algorithms (SODA'21) [GKL+21].

3.1 Introduction

As a central problem in unsupervised learning, graph clustering has been extensively studied in the past decades. Several formalizations of the problem have been considered in the literature. In this paper, we focus on the following (informal) variant of graph clustering: Given a graph G and an integer k, we are interested in finding k nonoverlapping sets $C_1, C_2, \ldots, C_k$ that are internally well-connected and that have a sparse cut to the outside. A popular approach to this problem is spectral clustering [KVV04b, NJW02, SM00, VL07]: One embeds vertices of the graph into k dimensional Euclidean space using the bottom k eigenvectors of the Laplacian, and clusters the points in Euclidean space using the k-means algorithm (in practice), or using a more careful space partitioning approach (in theory). Spectral clustering has been applied in the context of a wide variety of problems, for example, image segmentation [SM00], speech separation [BJ06], clustering of protein sequences [PCS06], and predicting landslides in geophysics [BMD+15]. Spectral clustering usually requires to process the graph in two steps. First one computes the spectral embedding and then one clusters the resulting point set. This two stage approach seems to be highly non-local and it seems to be hard to obtain faster methods, if one only has to determine the cluster membership for a small subset of the vertices. However, such a sublinear time access is desirable in some applications. As a basic step towards such a sublinear time clustering algorithm, we need a way to quickly access the spectral embedding in some way. Therefore, we ask the following question, where we use $f_x \in \mathbb{R}^k$ to denote the spectral embedding of vertex x:

Is it possible to obtain dot product access to the spectral embedding of a graph in sublinear time? In other words, given a pair of vertices $x, y \in V$, can we quickly approximate the dot product $\langle f_x, f_y \rangle$ in $o(n)$ time?

If such access is possible, it appears plausible that one can design a *sublinear spectral clustering oracle*, a small space data structure that provides fast query access to a good clustering of the graph. Our main result in this paper is **(a)** a small space data structure that provides query access to dot products in the spectral embedding, as above, and **(b)** a sublinear time spectral clustering oracle that uses this data structure.

We study a popular version of the spectral clustering problem where one assumes the existence of a planted solution, namely that the input graph can be partitioned into clusters $C_1, \ldots, C_k$ whose internal connectivity is nontrivially higher than the external connectivity. The goal is to recover the clusters approximately. An average case version of this problem, where the clusters induce Erdős-Rényi graphs (or random regular graphs), and the edges across clusters are similarly random, has been studied extensively in the literature on the stochastic block model (SBM) [Abb18] for its close relationship to the community detection problem. In this work we study a worst-case version of this problem:

Given a graph $G = (V, E)$ that admits a partitioning into a disjoint union of k induced expanders $C_1, \ldots, C_k$ with outer conductance bounded by $\epsilon \ll 1$, output an approximation to $C_1, \ldots, C_k$ that is correct up to a $O(\epsilon)$ error **on every cluster**.

We define a *spectral clustering oracle with per cluster error* $\delta \in (0, 1)$ as a small space data structure that implicitly defines disjoint subsets $\widehat{C}_1, \ldots, \widehat{C}_k$ of V such that for some permutation π on k elements one has $|C_i \Delta \widehat{C}_{\pi(i)}| \le \delta |C_i|$ for every $i = 1, \ldots, k$. The oracle must provide fast query access to such a clustering. The focus of this paper is:

Design a sublinear time spectral clustering oracle with per cluster error $\approx O(\epsilon)$.

Our main result is a spectral clustering oracle as above, with a slight loss in error parameter. Specifically, our spectral clustering oracle is correct up to $O(\epsilon \log k)$ error on every cluster:

Theorem 3.1 (Informal)**.** *There exists a spectral clustering oracle that for every graph $G = (V, E)$ that admits a partitioning into a disjoint union of k induced expanders $C_1, \ldots, C_k$ with outer conductance bounded by $\epsilon \ll \frac{1}{\log k}$ achieves error $O(\epsilon \log k)$ per cluster, query time $\approx n^{1/2+O(\epsilon)}$, preprocessing time $\approx 2^{O(\frac{1}{\epsilon} k^4 \log^2(k))} n^{1/2+O(\epsilon)}$ and space $\approx n^{1/2+O(\epsilon)}$.*

Query times can be made faster at the expense of increased space and prepropcessing time, as long as the product of query time and preprocessing time is $\approx n^{1+O(\epsilon)}$, *leading in particular to a nearly linear time algorithm for spectral clustering.*

As byproduct of our main result we also obtain new efficient clustering algorithms in the Local Computation Algorithms (LCA) model (see [RTVX11] for introduction of the model and [ARVX12] for LCA with limited randomness).

A very important feature of the problem above is the fact that our algorithms recovers a $1 - O(\epsilon \log k)$ fraction of **every cluster** as opposed to just classifying a $1 - O(\epsilon \log k)$ fraction of vertices of the graph correctly (this latter question allows one to output fewer than k clusters, and is much easier to solve). To put this in perspective, it is instructive to apply multiway Cheeger inequalities (e.g., [LGT14], [CKCLL$^+$13]) to our setting, noting that the k-th eigenvalue λ_k of the normalized Laplacian of a graph that can be partitioned into k clusters as above is bounded by $O(\epsilon)$. This means that multiway Cheeger inequalities can be used to recover k clusters with outer conductance $k^2\sqrt{\epsilon}$ (see [LGT14]), which becomes trivial unless $\epsilon < 1/k^4$ (we note that our problem admits a much simpler solution when $\epsilon \ll 1/k$). One may note that multiway Cheeger inequalities can also recover $0.9k$ clusters with outer conductance $\log^{O(1)} k\sqrt{\epsilon}$ in our setting (e.q. [LRTV12]), but, as mentioned above, recovering most clusters is much easier that recovering each cluster to $1 \pm O(\epsilon)$ multiplicative error, and does not solve our problem. The most relevant prior result is due to Sinop [Sin16], where the author achieves error $O(\sqrt{\epsilon})$ per cluster using spectral techniques. Sinop's result improves up on previous work of [AS12], which achieved per cluster error of $O(\epsilon k)$ (or, rather, is somewhat incomparable to [AS12] due to the worst dependence on ϵ, but a lack of dependence on k). As we argue below, Sinop's techniques are hard to extend to the sublinear time regime. At the same time, one should note that our result improves on [AS12] under the assumption that cluster sizes are comparable while using only sublinear time in the size of the input graph.

Main challenges and comparison to results on testing cluster structure. This problem is related the well-studied expansion testing problem [KS08, NS10, GR11b, CS10b, KPS13], which corresponds to the setting of one or two clusters, as well as to the problem of testing cluster structure of graphs, where one essentially wants to determine k, the number of clusters in G. The problem of *testing* cluster structure has recently been considered in the literature [CPS15]: given access to a graph G as above, compute the value of k (in fact, both results [CPS15] and [CKK$^+$18] apply to the harder property testing problem of distinguishing between graphs that are k-clusterable according to the definition above and graphs that are ϵ-far from k-clusterable, but a procedure for computing k is the centerpiece of both results). It is interesting to note that the work of [CPS15] also yields an algorithm for our problem, but only under very strong assumptions on the outer conductance of the clusters (one needs $\epsilon \ll \frac{1}{\text{poly}(k)\log n}$). The recent work of Peng [Pen20] considers a robust version of testing cluster structure, but requires $\epsilon \ll \frac{1}{\text{poly}(k)\log n}$, just like the work of [CPS15].

The recent work of [CKK$^+$18] on testing cluster structure yields an optimal tester, which works for any ϵ smaller than a constant and achieves essentially optimal runtime, but unfortunately their techniques do no extend to the 'learning' version of the problem. The reason is very simple: the algorithm of [CKK$^+$18] needs to distinguish between the graph G being a union of k clusters and $k+1$ clusters, and their approach amounts to verifying whether a graph can be partitioned into k clusters. To do so it suffices to check whether the spectral embedding is effectively k-dimensional, i.e. whether it spans a nontrivial $(k+1)$-dimensional volume. In order to certify this, however, it suffices to exhibit $k+1$ vertices that span a nontrivial $(k+1)$-dimensional volume. For that, one essentially only needs to locate at least one 'typical' point in every cluster, which is much easier than our task of correctly recovering almost all, i.e. a $1-O(\epsilon)$ fraction of vertices in every cluster. In other words, testing graph cluster structure requires only a rather basic access to and control of the spectral embedding. The main technical contribution of our paper is a set of tools for getting precise dot product access to this embedding, together with several new structural claims about it that enable our clustering algorithm.

Comparison to the work of Sinop [Sin16]. The work of Sinop [Sin16] gives a nearly linear time algorithm for recovering every cluster up to error of $1 \pm O(\sqrt{\epsilon})$ using spectral techniques[1], for sufficiently small ϵ. The algorithm would be very hard to implement in sublinear time, since one of its central tools (the ROUND procedure, which controls propagation of error i.e., Lemma 5.4 of [Sin16]) heavily relies on the ability to have explicit access to the eigendecomposition of the Laplacian. Specifically, Sinop's algorithm first finds a crude approximation S to a cluster to be recovered, and then improves the approximation by explicitly constructing the corresponding submatrix of the spectral embedding and performing an SVD. One could plausibly envision implementing this using random walks, but that would be challenging, since one would need to consider a random walk induced on a rather unstructured subset of vertices of the graph.

Our contributions: sublinear time access to the spectral embedding. Let $G=(V,E)$ be a d-regular graph with $n=|V|$. Without loss of generality we assume that $V=\{1,\ldots,n\}$. We assume that n and d are given to the algorithm and that we have oracle access to G: We can specify a vertex $x \in V$ and a number $i, 1 \le i \le d$, and we will be given in constant time the i-th neighbor of x. This is also called the bounded degree graph model.

In this paper we will consider d-regular graphs that have a certain cluster structure. We parameterize this cluster structure using the internal and external conductance parameters.

Definition 3.1 (**Internal and external conductance**)**.** Let $G=(V,E)$ be a graph. For a set $S \subseteq C \subseteq V$, let $E(S, C \setminus S)$ be the set of edges with one endpoint in S and the other in $C \setminus S$. The *conductance of a set S within C* is $\phi^G_C(S) = \frac{|E(S,C\setminus S)|}{d|S|}$. The *external-conductance* of set C is

[1] One must note that the work of [Sin16] does not require the bounded degree assumption, and can handle clusters of significantly different size.

defined to be $\phi_V^G(C) = \frac{|E(C,V\setminus C)|}{d|C|}$. The *internal-conductance* of set $C \subseteq V$, denoted by $\phi^G(C)$, is

$$\min_{S\subseteq C, 0<|S|\leq \frac{|C|}{2}} \phi_C^G(S)$$

if $|C| > 1$ and one otherwise.

Remark 3.1. *For simplicity we present all the proofs for d-regular graphs, even though all the proofs also work for d-bounded graphs, with the same definition of conductance as in Definition 3.1 (i.e., with normalization by $d|S|$ as opposed to the volume of S; the two notions of conductance can in the worst case differ by a factor of d). Note that this is equivalent to converting a d-bounded degree graph G to a d-regular graph G^{reg} by adding $d - deg(v)$ self-loops to each vertex v with degree $deg(v)$. Let L^{reg} be the normalized Laplacian of G^{reg}. Then the random walk on graph G is exactly same as a lazy random walk on graph G^{reg} and the definition of conductance is consistent.*

Based on the conductance, clusterability of graphs is defined as follows.

Definition 3.2 (**(k,φ,ϵ)-clustering**)**.** Let $G = (V,E)$ be a d-regular graph. A (k,φ,ϵ)-clustering of G is a partition of vertices V into disjoint subsets $C_1 \cup \ldots \cup C_k$ such that for all $i \in [k]$, $\phi^G(C_i) \geq \varphi$, $\phi_V^G(C_i) \leq \epsilon$ and for all $i, j \in [k]$ one has $\frac{|C_i|}{|C_j|} \in O(1)$. G is called (k,φ,ϵ)-clusterable if there exists a (k,φ,ϵ)-clustering for G.

We also need for formally define spectral embedding.

Definition 3.3 (Spectral embedding)**.** For a d-regular graph $G = (V,E)$ and integer $2 \leq k \leq n$ we define the spectral embedding of G as follows. Let $U \in \mathbb{R}^{k\times n}$ denote the matrix of the bottom k eigenvectors of the normalized Laplacian of G (this choice is not unique; fix any such matrix U). Then for every $x \in V$ the spectral embedding $f_x \in \mathbb{R}^k$ of x is the x-th column of the matrix U, which we write as $U = (f_y)_{y\in V}$.

Remark 3.2. *We note that the spectral embedding $f_x, x \in V$ is not uniquely defined. However, in this paper we are only interested in obtaining dot product access to this embedding, i.e. in fast algorithms for computing $\langle f_x, f_y\rangle$ for $x, y \in V$. Such dot products are in fact uniquely defined for any G that is (k,φ,ϵ)-clusterable with ϵ/φ^2 smaller than an absolute constant – see Remark 3.4 below.*

Our first algorithmic result is a sublinear time spectral dot product oracle:

Theorem 3.2. *[Spectral Dot Product Oracle] Let $\epsilon, \varphi \in (0,1)$ with $\epsilon \leq \frac{\varphi^2}{10^5}$. Let $G = (V,E)$ be a d-regular graph that admits a (k,φ,ϵ)-clustering $C_1,\ldots,C_k$. Let $\frac{1}{n^5} < \xi < 1$. Then* InitializeOracle$(G, 1/2, \xi)$ *(Algorithm 4) computes in time $(\frac{k}{\xi})^{O(1)} \cdot n^{1/2+O(\epsilon/\varphi^2)} \cdot (\log n)^3 \cdot \frac{1}{\varphi^2}$ a sublinear space data structure $\mathcal{D}$ of size $(\frac{k}{\xi})^{O(1)} \cdot n^{1/2+O(\epsilon/\varphi^2)} \cdot (\log n)^3$ such that with probability at least $1 - n^{-100}$ the following property is satisfied:*

For every pair of vertices $x, y \in V$, SpectralDotProduct$(G, x, y, 1/2, \xi, \mathscr{D})$ *(Algorithm 5) computes an output value $\langle f_x, f_y \rangle_{apx}$ such that with probability at least* $1 - n^{-100}$

$$\left| \langle f_x, f_y \rangle_{apx} - \langle f_x, f_y \rangle \right| \le \frac{\xi}{n}.$$

The running time of SpectralDotProduct$(G, x, y, 1/2, \xi, \mathscr{D})$ *is* $(\frac{k}{\xi})^{O(1)} \cdot n^{1/2+O(\epsilon/\varphi^2)} \cdot (\log n)^2 \cdot \frac{1}{\varphi^2}$.

Furthermore, for any $0 \le \delta \le 1/2$*, one can obtain the following trade-offs between preprocessing time and query time: Algorithm* SpectralDotProduct$(G, x, y, \delta, \xi, \mathscr{D})$ *requires* $(\frac{k}{\xi})^{O(1)} \cdot n^{\delta+O(\epsilon/\varphi^2)} \cdot (\log n)^2 \cdot \frac{1}{\varphi^2}$ *per query when the prepressing time of Algorithm* InitializeOracle(G, δ, ξ) *is increased to* $(\frac{k}{\xi})^{O(1)} \cdot n^{1-\delta+O(\epsilon/\varphi^2)} \cdot (\log n)^3 \cdot \frac{1}{\varphi^2}$.

Our results: a spectral clustering oracle. Our goal is to compute a data structure that provides sublinear time access to a (k, φ, ϵ)-clustering of G. Such a data structure is called a (k, φ, ϵ)-clustering oracle. We now formally define a spectral clustering oracle in the Local Computation (LCA) model:

Definition 3.4 (Spectral clustering oracle)**.** A randomized algorithm $\mathcal{O}$ is a (k, φ, ϵ)-clustering oracle if, when given query access to a d-regular graph $G = (V, E)$ that admits a (k, φ, ϵ)-clustering $C_1, \dots, C_k$, the algorithm $\mathcal{O}$ provides consistent query access to a partition $\widehat{P} = (\widehat{C}_1, \dots, \widehat{C}_k)$ of V. The partition $\widehat{P}$ is determined solely by G and the algorithm's random seed. Moreover, with probability at least $9/10$ over the random bits of $\mathcal{O}$ the partition $\widehat{P}$ has the following property: for some permutation π on k elements one has for every $i \in [k]$:

$$|C_i \triangle \widehat{C}_{\pi(i)}| \le O\left(\frac{\epsilon \cdot \log(k)}{\varphi^3}\right) |C_i|.$$

Remark 3.3. *Note that it is crucial that $\mathcal{O}$ provides consistent answers, i.e. classifies a given $x \in V$ in the same way every time it is queried (for a fixing of its random seed).*

We are interested in clustering oracles that perform few probes per query. Our main contribution is:

Theorem 3.3. *For every integer $k \ge 2$, every $\varphi \in (0,1)$, every $\epsilon \ll \frac{\varphi^3}{\log k}$, every $\delta \in (0, 1/2]$ there exists a (k, φ, ϵ)-clustering oracle that:*

- *has $\widetilde{O}_\varphi\left(2^{O\left(\frac{\varphi^2}{\epsilon} k^4 \log^2(k)\right)} \cdot n^{1-\delta+O(\epsilon/\varphi^2)}\right)$ preprocessing time,*
- *has $\widetilde{O}_\varphi\left(\left(\frac{k}{\epsilon}\right)^{O(1)} \cdot n^{\delta+O(\epsilon/\varphi^2)}\right)$ query time,*
- *uses $\widetilde{O}_\varphi\left(\left(\frac{k}{\epsilon}\right)^{O(1)} \cdot n^{1-\delta+O(\epsilon/\varphi^2)}\right)$ space,*

- *uses* $\widetilde{O}_\varphi\left(\left(\frac{k}{\epsilon}\right)^{O(1)} \cdot n^{O(\epsilon/\varphi^2)}\right)$ *random bits,*

where O_φ suppresses dependence on φ and $\widetilde{O}$ hides all polylog(n) *factors.*

To the best of our knowledge, our algorithm is the first sublinear spectral clustering algorithm in literature. We hope that our main technique for providing sublinear time access to the spectral embedding will have further applications in sublinear time spectral graph theory. Our simple algorithm for recovering clusters using hyperplane partitioning in a carefully defined sequence of subspaces may also be of independent interest in spectral partitioning problems. We provide a detailed overview of the analysis and the main ideas are involved in Section 3.3.

Other related work. Besides the work on property testing and the work on clustering with labelled, data another closely related area is local clustering. In local clustering one is interested of finding the entire cluster around a node v in time proportional to the size of the cluster. Several algorithms are known for this problem [ACL08, AGPT16, OA14, ST14, ALM13b] but unfortunately they cannot be applied to solve our problem because when the clusters have linear size they take linear time (in addition, the output clusters may overlap). In this paper instead we focus on solving the problem using strictly sublinear time.

3.2 Preliminaries

In this paper we mostly use the matrix notation to represent graphs. For a vertex $x \in V$, we say that $\mathbb{1}_x \in \mathbb{R}^n$ is the indicator of x, that is, the vector which is 1 at index x and 0 elsewhere. For a (multi) set $I_S = \{x_1, \ldots, x_s\}$ of vertices from V we abuse notation and also denote by S the $n \times s$ matrix whose i^{th} column is $\mathbb{1}_{x_i}$. For $i \in \mathbb{N}$ we use $[i]$ to denote the set $\{1, 2, \ldots, i\}$.

For a symmetric matrix A, we write $\nu_i(A)$ (resp. $\nu_{\max}(A), \nu_{\min}(A)$) to denote the i^{th} largest (resp. maximum, minimum) eigenvalue of A.

Let $m \leq n$ be integers. For any matrix $A \in \mathbb{R}^{n\times m}$ with singular value decomposition (SVD) $A = Y\Gamma Z^T$ we assume $Y \in \mathbb{R}^{n\times n}$, $\Gamma \in \mathbb{R}^{n\times n}$ is a diagonal matrix of singular values and $Z \in \mathbb{R}^{m\times n}$ (this is a slightly non-standard definition of the SVD, but having Γ be a square matrix will be convenient). Y has orthonormal columns, the first m columns of Z are orthonormal, and the rest of the columns of Z are zero. For any integer $q \in [m]$ we denote $Y_{[q]} \in \mathbb{R}^{n\times q}$ as the first q columns of Y and $Y_{-[q]}$ to denote the matrix of the remaining columns of Y. We also denote by $Z_{[q]} \in \mathbb{R}^{m\times q}$ as the first q columns of Z and $Z_{-[q]}$ to denote the matrix of the remaining $n-q$ columns of Z. Finally we denote by $\Gamma_{[q]} \in \mathbb{R}^{q\times q}$ the submatrix of Γ corresponding to the first q rows and columns of Γ and we use $\Gamma_{-[q]}$ to denote the submatrix corresponding to the last $n-q$ rows and $n-q$ columns of Γ. So for any $q \in [m]$ the span of $Y_{-[q]}$ is the orthogonal complement of the span of $Y_{[q]}$ in $\mathbb{R}^n$, also the span of the columns of $Z_{-[q]}$ is the orthogonal complement of the span of $Z_{[q]}$ in $\mathbb{R}^m$. Thus we can write $A = Y_{[q]}\Gamma_{[q]}Z_{[q]}^T + Y_{-[q]}\Gamma_{-[q]}Z_{-[q]}^T$.

We also denote with A_G the adjacency matrix of G and with L the *normalized Laplacian* of G where $L = I - \frac{A_G}{d}$. For L we denote its eigenvalues with $0 \le \lambda_1 \le \ldots \le \lambda_n \le 2$ and we write Λ to refer to the diagonal matrix of these eigenvalues in ascending order. We also denote with $(u_1, \ldots, u_n)$ an orthonormal basis of eigenvectors of L and with $U \in \mathbb{R}^{n\times n}$ the matrix whose columns are the orthonormal eigenvectors of L arranged in increasing order of eigenvalues. Therefore the eigendecomposition of L is $L = U\Lambda U^T$. We write $U_{[k]} \in \mathbb{R}^{n\times k}$ for the matrix whose columns are the first k columns of U and also define $F = U_{[k]}^T$. For every vertex x we denote the spectral embedding of vertex x on the bottom k eigenvectors of L with $f_x \in \mathbb{R}^k$, i.e. $f_x = F\mathbb{1}_x$. For pairs of vertices $x, y \in V$ we use the notation

$$\langle f_x, f_y\rangle := f_x^T f_y$$

to denote the dot product in the embedded domain.

Remark 3.4. *We note that if G is a (k, φ, ϵ)-clusterable graph with ϵ/φ^2 smaller than a constant, the space spanned by the bottom k eigenvectors of the normalized Laplacian of G is uniquely defined, i.e. the choice of $U_{[k]}$ is unique up to multiplication by an orthonormal matrix $R \in \mathbb{R}^{k\times k}$ on the right. Indeed, by Lemma 3.3 below one has $\lambda_k \le 2\epsilon$ and by Lemma 3.1 below one has $\lambda_{k+1} \ge \varphi^2/2$. Thus, since we assume that ϵ/φ^2 is smaller than an absolute constant, we have $2\epsilon < \varphi^2/2$, and therefore the subspace spanned by the bottom k eigenvectors of the Laplacian, i.e. the space of $U_{[k]}$, is uniquely defined, as required. We note that while the choice of f_x for $x \in V$ is not unique, but the dot product between the spectral embedding of $x \in V$ and $y \in V$ is well defined, since for every orthonormal $R \in \mathbb{R}^{k\times k}$ one has $\langle Rf_x, Rf_y\rangle = (Rf_x)^T(Rf_y) = f_x^T(R^TR)f_y = f_x^T f_y$.*

In this paper we also consider the transition matrix of the *random walk associated with G* $M = \frac{1}{2}\cdot\left(I + \frac{A}{d}\right)$. From any vertex v, this random walk takes every edge incident to v with probability $\frac{1}{2d}$, and stays on v with the remaining probability which is at least $\frac{1}{2}$. Note that this random walk is exactly same as a lazy random walk on G and that $M = I - \frac{L}{2}$. Observe that $\forall i$ u_i is also an eigenvector of M, with eigenvalue $1 - \frac{\lambda_i}{2}$. We denote with Σ the diagonal matrix of the eigenvalues of M in descending order. Therefore the eigendecomposition of M is $M = U\Sigma U^T$. We write $\Sigma_{[k]} \in \mathbb{R}^{k\times k}$ for the matrix whose columns are the first k rows and columns of Σ. Furthermore, for any t, M^t is a transition matrix of random walks of length t. For any vertex x, we denote the probability distribution of a t-step random walk starting from x by $m_x = M^t\mathbb{1}_x$. For a (multi) set $I_S = \{x_1, \ldots, x_s\}$ of vertices from V, let matrix $M^tS \in \mathbb{R}^{n\times s}$ is a matrix whose columns are probability distributions of t-step random walks starting from vertices in I_S. More formally the ith column of M^tS is m_{x_i}. For any vertex $x \in V$ let $\mathcal{N}(x) : \{y \in V : \{x, y\} \in E\}$ denote the set of vertices that are adjacent to the vertex x.

Definition 3.5 (**Cluster Centers**)**.** Let $G = (V, E)$ be a d-regular graph. Let $C_1, \ldots, C_k$ be a (k, φ, ϵ)-clustering of G. We define the *spectral center* of cluster C_i as

$$\mu_i := \frac{1}{|C_i|}\sum_{x\in C_i} f_x.$$

For vertex $x \in V$, we define μ_x as the cluster center of the cluster which x belongs to.

In our analysis we use the following standard results on eigenvalues and matrix norms. Recall that for any $m \times n$ matrix A, the multi-sets of nonzero eigenvalues of AA^T and $A^T A$ are equal.

Lemma 3.1 ([CKK⁺18]). *Let G be any graph which is composed of k components $C_1, \dots C_k$ such that $\phi^G(C_i) \geq \varphi$ for any $i \in [k]$. Let L be the normalized Laplacian matrix of G, and λ_{k+1} be the $(k+1)$st smallest eigenvalue of L. Then $\lambda_{k+1} \geq \frac{\varphi^2}{2}$.*

For a d-regular graph G, let $\rho_G(k)$ denote the minimum value of the maximum conductance over any possible k disjoint nonempty subsets. That is

$$\rho_G(k) \leq \min_{\text{disjoint } S_1,\dots,S_k} \max_i \phi_G(S_i)$$

Lemma 3.2 ([LGT14]). *For any d-regular graph G and any $k \geq 2$, it holds that*

$$\lambda_k \leq 2\rho_G(k).$$

Lemma 3.3. *Let $G = (V, E)$ be a d regular graph that admits a (k, φ, ϵ)-clustering $C_1, \dots, C_k$. Let L be the normalized Laplacian matrix of G. Let $\lambda_1 \leq \dots \leq \lambda_n$ be eigenvalues of L, then we have $\lambda_{k+1} \geq \frac{\varphi^2}{2}$ and $\lambda_k \leq 2\epsilon$.*

Proof. Note that G is composed of k components $C_1, \dots C_k$ such that for all $1 \leq i \leq k$ we have $\phi^G(C_i) \geq \varphi$. Hence, by Lemma 3.1 we get $\lambda_{k+1} \geq \frac{\varphi^2}{2}$. Moreover for all $1 \leq i \leq k$, we have $\phi_V^G(C_i) \leq \epsilon$. Thus by Lemma 3.2 we have $\lambda_k \leq 2\epsilon$. □

Since we assume that the maximum ratio of cluster sizes is bounded by a constant, we have

Proposition 3.1. *Let $G = (V, E)$ be a d regular graph that admits a (k, φ, ϵ)-clustering $C_1, \dots, C_k$. Then we have $\min_{i \in \{1,\dots,k\}} |C_i| = \Omega\left(\frac{n}{k}\right)$ and $\max_{i \in \{1,\dots,k\}} |C_i| = O\left(\frac{n}{k}\right)$.*

A symmetric $n \times n$ matrix is positive semi-definite, if and only if all its eigenvalues are non-negative. The spectral norm of matrix $A \in \mathbb{R}^{n \times n}$ is defined as $\max_{x \in \mathbb{R}^n, x \neq 0} \frac{\|Ax\|_2}{\|x\|_2}$ that equals the square root of the largest eigenvalue of the matrix $A^T A$. The Frobenius norm of a matrix A is defined as $\sqrt{\sum_{i,j}(A_{i,j})^2}$. For matrices $A, \widetilde{A} \in \mathbb{R}^{n \times n}$, we write $A \preccurlyeq \widetilde{A}$, if $\forall x \in \mathbb{R}^n$ we have $x^T A x \leq x^T \widetilde{A} x$.

3.3 Technical overview

In this section we give an overview of the analysis and the main technical contributions of the paper. Recall that we denote the matrix of bottom k eigenvectors of the normalized Laplacian

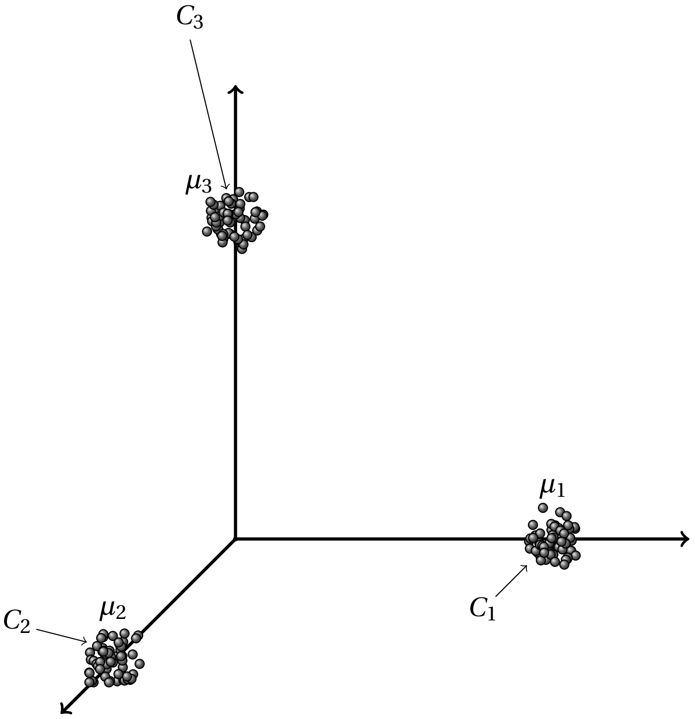

Figure 3.1 – Example of a spectral embedding where points are concentrated around means.

of G by $U_{[k]}$. The spectral embedding of a vertex $x \in V$, denoted by $f_x \in \mathbb{R}^k$, is simply the x-th column of $U_{[k]}^T$. The main intuition behind spectral clustering is that the points $f_x \in \mathbb{R}^k$ are well-concentrated around cluster means $\mu_i \in \mathbb{R}^k$, defined for every $i = 1, \ldots, k$ by

$$\mu_i = \frac{1}{|C_i|} \sum_{x \in C_i} f_x. \tag{3.1}$$

See Fig. 3.1 for an illustration.

The contributions of our paper are twofold. Our first contribution is a primitive that provides dot product access to the spectral embedding of a graph in sublinear time: we show in Theorem 3.2 how, given any pair of vertices $x, y \in V$ one can compute

$$\langle f_x, f_y \rangle_{apx} \approx \langle f_x, f_y \rangle, \tag{3.2}$$

in time $\approx n^{1/2+O(\epsilon)}$ per evaluation (see Algorithm 10 in Section 3.5 for the formal definition of $\langle \cdot, \cdot \rangle_{apx}$ and its analysis).

Our second contribution is to show how dot product access as in (3.2) above allows one to solve the cluster recovery problem. Both of these contributions are based on a new property of the spectral embedding that we establish. This property allows us to quantify the intuitive statement that vertices in the embedding concentrate around cluster means defined in (3.2) above in a very strong formal sense.

In the rest of this section we first present our sublinear time dot product oracle (in Section 3.3.1) and then outline how access to such an oracle can be used to design a simple spectral clustering algorithm (in Section 3.3.2). We assume that the inner conductance of the clusters φ is constant for the purposes of this overview to simplify notation.

3.3.1 Sublinear time dot product access to the spectral embedding

We start with a description of the main underlying ideas underlying the proof of Theorem 3.2. Our starting point from earlier work is the observation that collision statistics of random walks can be used to exhibit the structure of a (k,φ,ϵ)-clusterable graph. In particular, in (k,φ,ϵ)-clusterable graphs, there is a gap between λ_k and λ_{k+1}, and the behavior of random walks is essentially determined by the bottom k eigenvectors of the Laplacian and the corresponding eigenvalues. This suggests that we can potentially use random walks to determine the spectral embedding. The spectral embedding is of course not necessarily unique (for example, if not all of the bottom k eigenvalues are unique). However, the dot product of the embedded vertices is still well-defined as a function of the subspace spanned by the bottom k eigenvectors of the Laplacian, as the subspace itself is uniquely defined because of the aforementioned gap between λ_k and λ_{k+1}. See Remark 3.4 for more details. We now give an overview of our approach.

Fix two vertices $x, y \in V$. We would like to compute

$$\langle f_x, f_y \rangle = (F\mathbb{1}_x)^T (F\mathbb{1}_y) = \mathbb{1}_x^T U_{[k]} U_{[k]}^T \mathbb{1}_y.$$

The direct approach to this would amount to computing an eigendecomposition of M to obtain $U_{[k]}$, but that would take at least $\Omega(n)$ time and is too expensive for our purposes. On the other hand, it is well-known that we are able to estimate, in about $n^{1/2}$ time, the dot product

$$(M^t\mathbb{1}_x)^T (M^t\mathbb{1}_y) = \mathbb{1}_x^T M^{2t} \mathbb{1}_y.$$

Note that $\mathbb{1}_x^T M^{2t}\mathbb{1}_y = \mathbb{1}_x^T U\Sigma^{2t}U^T\mathbb{1}_y$. Thus to get $U_{[k]}U_{[k]}^T$ from $\mathbb{1}_x^T M^{2t}\mathbb{1}_y$ we need to remove the matrix Σ^{2t} from the middle. Specifically, we can estimate the quantity above as follows. For some precision parameter $\xi \in (0,1)$ we first run $\approx n^{1/2+O(\epsilon/\varphi^2)}/\xi^2$ random walks from x, letting $\widehat{m}_x \in \mathbb{R}^n$ denote a vector whose a'th component is the fraction of random walks from x that end up at a. Similarly, we run $\approx n^{1/2+O(\epsilon/\varphi^2)}/\xi^2$ random walks from y, letting $\widehat{m}_y \in \mathbb{R}^n$ denote a vector whose a'th component is the fraction of random walks from y that end up at a. One can show[2] that with high (constant) probability we have

$$\left|\widehat{m}_x^T \widehat{m}_y - \mathbb{1}_x^T M^{2t}\mathbb{1}_y\right| \leq \xi \cdot \frac{1}{n}. \tag{3.3}$$

[2] This calculation is mostly amounts to a rather standard collision counting calculation that relies on the birthday paradox if one wants to establish the claim **for most vertices** $x, y \in V$ (this was done in [CPS15] and [CKK+18] for example). Our new moment bounds for the spectral embedding (see Lemmas 3.4 and 3.5 in Section 3.4) allow us to establish such a claim **for all vertices** $x, y \in V$ – see Lemma 3.22.

While (3.3) is not directly useful, a primitive for constructing empirical distributions $\widehat{m}_x$ and $\widehat{m}_y$ as above is a central part of our approach. We formalize it as Algorithm 6 (RUNRANDOMWALKS) below:

Algorithm 6 RUNRANDOMWALKS(G, R, t, x)

```
Run $R$ random walks of length $t$ starting from $x$
Let $\widehat{m}_x(y)$ be the fraction of random walks that ends at $y$ ▷ vector $\widehat{m}_x$ has support at most $R$
return $\widehat{m}_x$
```

Even if we cannot apply (3.3) directly, it lets us compute a seemingly related to quantity $\mathbb{1}_x^T M^{2t} \mathbb{1}_y$ quickly by invoking Algorithm 6 and computing one dot product. In order to get from $\mathbb{1}_x^T M^{2t} \mathbb{1}_y$ to $\mathbb{1}_x^T U_{[k]} U_{[k]}^T \mathbb{1}_y$, we need to somehow apply a linear transformation on the random walk distributions **before** computing the dot product between them, i.e. we need a different dot product operation. It is easy to see that the correct linear transformation is given by the matrix $U_{[k]}\Sigma_{[k]}^{-2t}U_{[k]}^T$, where $M^t = U\Sigma^t U^T$ is the eigendecomposition of M and $U_{[k]}$ stands for the matrix of bottom k eigenvectors of the Laplacian[3]. Specifically, we have

$$(M^t\mathbb{1}_x)^T(U_{[k]}\Sigma_{[k]}^{-2t}U_{[k]}^T)(M^t\mathbb{1}_y) = \mathbb{1}_x^T U_{[k]} U_{[k]}^T \mathbb{1}_y = \langle f_x, f_y \rangle,$$

which is exactly the quantity we are interested in. Of course, there is a major problem with this approach, since $U_{[k]}\Sigma_{[k]}^{-2t}U_{[k]}^T$ is an $n \times n$ matrix! To get around this issue, we approximate $U_{[k]}\Sigma_{[k]}^{-2t}U_{[k]}^T$ by a sparse low rank matrix, as we describe below. Specifically, we let I_S be a multiset of $s \ll n$ vertices selected uniformly at random. Let S be the $n \times s$ matrix whose j-th column equals $\mathbb{1}_{i_j}$ and let $\widetilde{W}\widetilde{\Sigma}^{2t}\widetilde{W}^T$ denote the eigendecomposition of $\frac{n}{s} \cdot (M^t S)^T (M^t S)$[4]. We show that with an appropriate choice of the sampling parameter $s \ll n$ one has

$$U_{[k]}\Sigma_{[k]}^{-2t}U_{[k]}^T \approx M^t S \cdot \widetilde{\Psi} \cdot S^T M^t, \tag{3.4}$$

where

$$\widetilde{\Psi} = \frac{n}{s} \cdot \widetilde{W}_{[k]}\widetilde{\Sigma}_{[k]}^{-4t}\widetilde{W}_{[k]}^T \tag{3.5}$$

is an $s \times s$ matrix that can be computed explicitly. The corresponding primitive to compute $(M^t S)^T (M^t S)$ is presented as Algorithm 7 (ESTIMATECOLLISIONPROBABILITIES) below. It basically estimates the Gram matrix of random walk distributions out of I_S (denoted by $\mathcal{G}$) by counting collisions, and taking medians of estimates to reduce failure probability appropriately. After computing the approximate Gram matrix, we derive from it the matrix

[3] Note that this matrix is not well defined in the presence of repeated eigenvectors, but any fixed choice of this matrix suffices for our purposes. It is also interesting to note that while we use a canonical choice of the eigendecomposition of M throughout the paper, all our bounds are oblivious to the choice of this basis, and hold for the *subspace* of bottom k eigenvectors, which is well defined since there is a gap between the k-th and $(k+1)$-th eigenvalues in k-clusterable graphs.

[4] We abuse notation somewhat by writing S to denote the $n \times s$ matrix whose (a, j)-th entry equals 1 if the j-th sampled vertex equals a and 0 otherwise.

$\Psi = \frac{n}{s}\cdot \widehat{W}_{[k]}\widehat{\Sigma}_{[k]}^{-2}\widehat{W}_{[k]}^T$, where $\mathcal{G} = \widehat{W}\widehat{\Sigma}\widehat{W}^T$ is the eigendecomposition of $\mathcal{G}$ (see line (8) and line (10) of Algorithm 9; note that G is a symmetric matrix, and hence an eigendecomposition exists).

Algorithm 7 ESTIMATECOLLISIONPROBABILITIES(G, I_S, R, t)

```
for $i = 1$ to $O(\log n)$ do
    $\widehat{Q}_i :=$ ESTIMATETRANSITIONMATRIX$(G, I_S, R, t)$
    $\widehat{P}_i :=$ ESTIMATETRANSITIONMATRIX$(G, I_S, R, t)$
    $\mathcal{G}_i := \frac{1}{2}\left(\widehat{P}_i^T\widehat{Q}_i + \widehat{Q}_i^T\widehat{P}_i\right)$        ▷ $\mathcal{G}_i$ is symmetric
Let $\mathcal{G}$ be a matrix obtained by taking the entrywise median of $\mathcal{G}_i$'s        ▷ $\mathcal{G}$ is symmetric
return $\mathcal{G}$        ▷ $\mathcal{G} \in \mathbb{R}^{s\times s}$
```

Algorithm 7 uses an auxiliary primitive presented as

Algorithm 8 ESTIMATETRANSITIONMATRIX(G, I_S, R, t)

```
for each sample $x \in I_S$ do
    $\widehat{m}_x :=$ RUNRANDOMWALKS$(G, R, t, x)$
Let $\widehat{Q}$ be the matrix whose columns are $\widehat{m}_x$ for $x \in I_S$
return $\widehat{Q}$        ▷ $\widehat{Q}$ has at most $Rs$ non-zeros
```

The proof of (3.4) relies on matrix perturbation bounds (the Davis-Kahan $\sin\theta$ theorem) as well as spectral concentration inequalities, crucially coupled with our tail bounds on the spectral embedding (see Lemma 3.4 and Lemma 3.5). In particular Lemma 3.4 and it's consequence - Lemma 3.5 can be used to bound the leverage scores of $U_{[k]}$ (i.e. $||f_x||_2^2$ for $x \in V$). This part of the analysis is presented in Section 3.5.2.

Lemma 3.4. *[Tail-bound] Let $\varphi \in (0,1)$ and $\epsilon \le \frac{\varphi^2}{100}$, and let $G = (V, E)$ be a d-regular graph that admits (k, φ, ϵ)-clustering $C_1, \ldots, C_k$. Let L be the normalized Laplacian of G. Let u be a normalized eigenvector of L with $||u||_2 = 1$ and with eigenvalue at most 2ϵ. Then for any $\beta > 1$ we have*

$$\frac{1}{n}\cdot\left|\left\{x \in V : |u(x)| \ge \beta\cdot\sqrt{\frac{10}{\min_{i\in[k]}|C_i|}}\right\}\right| \le \left(\frac{\beta}{2}\right)^{-\varphi^2/20\cdot\epsilon}.$$

Lemma 3.5. *Let $\varphi \in (0,1)$ and $\epsilon \le \frac{\varphi^2}{100}$, and let $G = (V, E)$ be a d-regular graph that admits (k, φ, ϵ)-clustering $C_1, \ldots, C_k$. Let u be a normalized eigenvector of L with $||u||_2 = 1$ and with eigenvalue at most 2ϵ. Then we have*

$$||u||_\infty \le n^{20\cdot\epsilon/\varphi^2}\cdot\sqrt{\frac{160}{\min_{i\in k}|C_i|}}.$$

We note that the number of samples s is chosen as $s \approx k^{O(1)}n^{O(\epsilon/\varphi^2)}$ (see Algorithm 9) , where the second factor is due to our upper bound on the ℓ_∞ norm of the bottom k eigenvectors of the Laplacian of a (k, φ, ϵ)-clusterable graph proved in Section 3.4.

Once we establish (3.4) in Section 3.5.3 (see Lemma 3.19), we get for every $x, y \in V$

$$(M^t \mathbb{1}_x)^T M^t S \cdot \widetilde{\Psi} \cdot S^T M^t (M^t \mathbb{1}_y) \approx \mathbb{1}_x^T U_{[k]} U_{[k]}^T \mathbb{1}_y, \tag{3.6}$$

which is what we would like to compute. One issue remains at this point, which is that we cannot compute $M^t \mathbb{1}_x$ or $M^t \mathbb{1}_y$ explicitly, and neither can we store and compute our approximation $M^t S \cdot \Psi \cdot S^T M^t$, since it is a dense, albeit low rank, matrix. We resolve this problem by running an appropriate number of random walks out of the sampled nodes I_S, as well as the queried nodes $x, y \in V$. Specifically, we run $\approx n^{1/2+O(\epsilon)}$ random walks from every sampled node in I_S, defining an $n \times s$ matrix Q whose (a, b)-th entry is the fraction of walks from a that ended at b and using the matrix Q as a proxy for $M^t S$ (note that the expectation of Q is exactly $M^t S$). Such a matrix Q is computed as per line (2) and line (3) of Algorithm 7 (ESTIMATECOLLISIONPROBABILITIES). We note that Algorithm 9 (INITIALIZEORACLE) performs $O(\log n)$ independent estimates that we ultimately use to boost confidence (by the median trick). The entire preprocessing is summarized in Algorithm 9 (INITIALIZEORACLE) below:

Algorithm 9 INITIALIZEORACLE(G, δ, ξ) ▷ Need: $\epsilon/\varphi^2 \le \frac{1}{10^5}$

```
$t := \frac{20 \cdot \log n}{\varphi^2}$
$R_{\text{init}} := O(n^{1-\delta+980\cdot\epsilon/\varphi^2} \cdot k^{17}/\xi^2)$
$s := O(n^{480\cdot\epsilon/\varphi^2} \cdot \log n \cdot k^8/\xi^2)$
Let $I_S$ be the multiset of $s$ indices chosen independently and uniformly at random from $\{1, \dots, n\}$
for $i = 1$ to $O(\log n)$ do
    $\widehat{Q}_i :=$ ESTIMATETRANSITIONMATRIX$(G, I_S, R_{\text{init}}, t)$   ▷ $\widehat{Q}_i$ has at most $R_{\text{init}} \cdot s$ non-zeros
$\mathcal{G} :=$ESTIMATECOLLISIONPROBABILITIES$(G, I_S, R_{\text{init}}, t)$
Let $\frac{n}{s} \cdot \mathcal{G} := \widehat{W}\widehat{\Sigma}\widehat{W}^T$ be the eigendecomposition of $\frac{n}{s} \cdot \mathcal{G}$   ▷ $\mathcal{G} \in \mathbb{R}^{s \times s}$
if $\widehat{\Sigma}^{-1}$ exists then
    $\Psi := \frac{n}{s} \cdot \widehat{W}_{[k]} \widehat{\Sigma}_{[k]}^{-2} \widehat{W}_{[k]}^T$   ▷ $\Psi \in \mathbb{R}^{s \times s}$
    return $\mathcal{D} := \{\Psi, \widehat{Q}_1, \dots, \widehat{Q}_{O(\log n)}\}$
```

Equipped with the primitives presented above, we can now state our final dot product estimate:

$$\widehat{m}_x^T Q \Psi Q^T \widehat{m}_y \approx \mathbb{1}_x^T U_{[k]} U_{[k]}^T \mathbb{1}_y = \langle f_x, f_y \rangle, \tag{3.7}$$

where $\widehat{m}_x$ and $\widehat{m}_y$ are empirical distributions of $\approx n^{1/2+O(\epsilon/\phi^2)}$ out of x and y respectively, Q is an $n \times s$ matrix with $\approx n^{1/2+O(\epsilon/\phi^2)}$ nonzeros per column, and Ψ is a possibly dense $s \times s$ matrix, where the number of sampled vertices s is ultimately chosen to be $k^{O(1)} n^{O(\epsilon/\phi^2)}$. The analysis of the error incurred in replacing (3.4) with (3.7) is presented in Section 3.5.4. It relies on a birthday paradox style variance computation similar to previous sublinear time algorithms for testing graph cluster structure. The actual query procedure that implements (3.7) is given by Algorithm 10 below.

Algorithm 10 SpectralDotProductOracle($G, x, y, \delta, \xi, \mathscr{D}$) ▷ Need: $\epsilon/\varphi^2 \le \frac{1}{10^5}$
▷ $\mathscr{D} := \{\Psi, \widehat{Q}_1, \dots, \widehat{Q}_{O(\log n)}\}$

```
$R_{\text{query}} := O(n^{\delta+500\cdot\epsilon/\varphi^2} \cdot k^9/\xi^2)$
for $i = 1$ to $O(\log n)$ do
    $\widehat{m}_x^i :=$ RunRandomWalks($G, R_{\text{query}}, t, x$)
    $\widehat{m}_y^i :=$ RunRandomWalks($G, R_{\text{query}}, t, y$)
Let $\alpha_x$ be a vector obtained by taking the entrywise median of $(\widehat{Q}_i)^T(\widehat{m}_x^i)$ over all runs
Let $\alpha_y$ be a vector obtained by taking the entrywise median of $(\widehat{Q}_i)^T(\widehat{m}_y^i)$ over all runs
return $\langle f_x, f_y \rangle_{apx} := \alpha_x^T \Psi \alpha_y$
```

Trading off preprocessing time for query time. Finally, we note that one can reduce query time (i.e., runtime of SpectralDotProductOracle) at the expense of increased preprocessing time and size of data structure. Specifically, one can run $\approx n^{\delta+O(\epsilon/\phi^2)}$ random walks from nodes x, y whose dot product is being estimated by SpectralDotProductOracle at the expense of increasing the number of random walks run to generate the matrix Q in InitializeOracle to $\approx n^{1-\delta+O(\epsilon/\phi^2)}$, for any $\delta \le 1/2$. This in particular leads to a nearly linear time spectral clustering algorithm.

3.3.2 Geometry of the spectral embedding

We now describe our spectral clustering algorithm. Since we only have dot product access to the spectral embedding, the algorithm must be very simple. Indeed, our algorithm amounts to performing hyperplane partitioning in a sequence of carefully crafted subspaces of the embedding space, using (a good approximation to) cluster means μ_i.

We first present a simple hyperplane partitioning, then we give an example embedding to show why it might be hard to prove that this scheme works. After that we design a modification of the hyperplane partitioning scheme that, through the course of carving, carefully projects out some directions of the embedding. This modification is an idealized version of our final algorithm for which we can prove per cluster recovery guarantees.

First we assume that the cluster means (3.1) are known. In that case we define, for every $i = 1, \dots, k$, the sets

$$\widetilde{C}_i := \{x \in V : \langle f_x, \mu_i \rangle \ge 0.9||\mu_i||^2\}$$

of points that are nontrivially correlated with the i-th cluster mean μ_i. Note that $\widetilde{C}_i = C_{\mu_i,0.9}$ in terms of Definition 3.8, but since μ_i's are fixed in this overview, we use the simpler notation.

We next define, for every $i = 1, \ldots, k$,

$$\widehat{C}_i := \widetilde{C}_i \setminus \bigcup_{j=1}^{i-1} \widetilde{C}_j. \tag{3.8}$$

In other words, this is a natural 'hyperplane-carving' approach: points that belong to the first hyperplane $\widetilde{C}_1$ are taken as the first cluster, points in the second hyperplane $\widetilde{C}_2$ that were not captured by the first hyperplane are taken as the second cluster, etc. This is a natural high dimensional analog of the Cheeger cut that has been used in many results on spectral partitioning. The hope here would be to show that there exists a permutation π on $[k]$ such that

$$|\widehat{C}_i \Delta C_{\pi(i)}| \leq O(\epsilon) \cdot |C_{\pi(i)}|, \tag{3.9}$$

for every $i = 1, \ldots, k$, where we assume that the inner conductance ϕ of the clusters is constant. Here Δ stands for the symmetric difference operation.

One natural approach to establishing (3.9) would be to prove that for every $i = 1, \ldots, k$ vertices $x \in C_i$ concentrate well around cluster means μ_i (see Fig. 3.1). This would seem to suggest that $\widetilde{C}_i$'s are close to the C_i's, and so are the $\widehat{C}_i$'s. This property of the spectral embedding is quite natural to expect, and versions of this property have been used in the literature. For example, one can show that for every $\alpha \in \mathbb{R}^k, ||\alpha||_2 = 1$,

$$\sum_{i=1}^{k} \sum_{x \in C_i} \langle f_x - \mu_i, \alpha \rangle^2 \leq O(\epsilon). \tag{3.10}$$

The bound in (3.10) follows using rather standard techniques – see Section 3.4.1 for this and related claims. One can check that (3.10) suffices to show that $\widetilde{C}_i$'s are very close to C_i's, namely that for every $i = 1, \ldots, k$ there exists $j \in [k]$ such that

$$|\widetilde{C}_i \Delta C_j| = O(\epsilon) \cdot |C_j|. \tag{3.11}$$

The formal proof is given in Section 3.6.2. The result in (3.11) is encouraging and suggests that the clusters $\widehat{C}_i$ defined by the simple hyperplane partitioning process approximate the C_i's, but this is not the case! The problem lies in the fact that while $\widetilde{C}_i$'s approximate the C_i's well as per (3.11), the bound in (3.11) does not preclude nontrivial overlaps in the $\widetilde{C}_i$'s – we give an example in below.

Hard instance for natural hyperplane partitioning

We now give an example configuration of vertices in Euclidean space such that **(a)** the configuration does not contradict (3.10) and **(b)** the natural hyperplane partitioning algorithm (3.8) fails for this configuration. This shows why we develop a different algorithm that can deal with configurations like the one presented in this subsection.

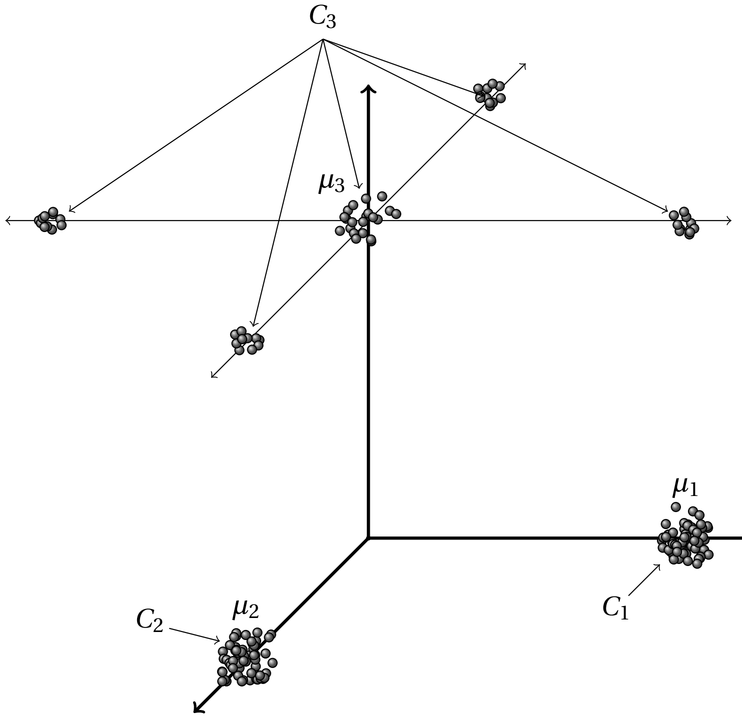

Figure 3.2 – Example of a spectral embedding that is consistent with (3.10) and (3.11) but for which the natural hyperplane partitioning would not work.

Consider the following configuration of C_i's and μ_i's. Suppose that all cluster sizes are equal $\frac{n}{k}$, and let $k = \frac{1}{\epsilon}$. Let μ_i's form an orthogonal system and for each $i \in [k]$ let $||\mu_i||_2 = \sqrt{\frac{k}{n}}$. For all $i < k = 1/\epsilon$ for all $x \in C_i$ we set $f_x = \mu_i$, that is points from all clusters except for $1/\epsilon$'th one are tightly concentrated around cluster means – see Fig. 3.2 for an illustration with $k = 3$. Then for cluster $C_{1/\epsilon}$ we distribute points as follows. For every $i = 1,\ldots,1/\epsilon - 1$ we move $\epsilon/2$ fraction of its points to $\mu_{1/\epsilon} + \mu_i$, and another $\epsilon/2$ fraction of the points to $\mu_{1/\epsilon} - \mu_i$. The remaining ϵ fraction of $C_{1/\epsilon}$ stays at $\mu_{1/\epsilon}$. Now observe that all cluster means are where they should be, since we applied symmetric perturbations. Secondly notice that (3.10) is satisfied for every direction α. Intuitively it is the case because we moved $1/\epsilon - 1$ disjoint subsets of $C_{1/\epsilon}$ of size $\epsilon\frac{n}{k}$ in $1/\epsilon - 1$ **orthogonal** directions. Lastly observe what happens to $\widetilde{C}_i$'s. For all $i = 1,\ldots,1/\epsilon - 1$ set $\widetilde{C}_i$ contains C_i and $\epsilon/2$ fraction of $C_{1/\epsilon}$ that was moved in direction μ_i. One can verify that this is perfectly consistent with (3.10), and in particular with (3.11). The problem is that many clusters have large overlap with one particular cluster, namely $C_{1/\epsilon}$. Indeed notice that the ball carving process returns $\widehat{C}_{1/\epsilon}$ such that $|\widehat{C}_{1/\epsilon} \cap C_{1/\epsilon}| = (\frac{1+\epsilon}{2})\frac{n}{k}$. That means that constant (almost 1/2) fraction of cluster $C_{1/\epsilon}$ is not recovered!

Our hyperplane partitioning scheme

The example in Section 3.3.2 suggests that we need to develop a diffferent algorithm. Our main contribution here is an algorithm that more carefully deals with the overlaps of $\widetilde{C}_i$'s. The high level idea for the algorithm is to recover clusters in stages and after every stage project out the directions corresponding to recovered clusters.

First we observe the following property of (k,φ,ϵ)-clusterable graphs (see Lemma 3.16). Any collection of pairwise disjoint sets with small outer-conductance matches the original clusters well. More precisely for every collection $\{\widehat{C}_1,\ldots,\widehat{C}_k\}$ of pairwise disjoint sets satisfying for every $i\in[k]$ $\phi(\widehat{C}_i)\le O(\epsilon\log(k))$ there exists a permutation π on $[k]$ such that

$$|\widehat{C}_i \Delta C_{\pi(i)}| \le O(\epsilon\log(k))\cdot|C_{\pi(i)}|, \tag{3.12}$$

In the algorithm we will test many candidate clusters and the property above allows us to test if a particular candidate $\widehat{C}$ is good by only computing its outer-conductance.

Now we describe our algorithm more formally. The algorithm proceeds in $O(\log(k))$ stages. In the first stage it considers k candidate clusters $\widehat{C}_i$, where $x\in\widehat{C}_i$ if it has big correlation with μ_i but small correlation with all other μ_j's. More formally

$$\widehat{C}_i := \widetilde{C}_i \setminus \bigcup_{j\neq i} \widetilde{C}_j, \tag{3.13}$$

which is equivalent to:

$$\langle f_x,\mu_i\rangle \ge 0.9||\mu_i||^2 \text{ and for all } j\neq i\, \langle f_x,\mu_j\rangle < 0.9||\mu_j||^2.$$

Note that by definition all these clusters are disjoint. At this point we return all candidate clusters $\widehat{C}_i$ for which $\phi(\widehat{C}_i)\le O(\epsilon)$, remove the corresponding vertices from the graph, remove the corresponding μ's from the set $\{\mu_1,\ldots,\mu_k\}$ of centers and proceed to the next stage.

In the next stage we restrict our attention to a lower dimensional subspace Π of $\mathbb{R}^k$. Intuitively we want to project out all the directions corresponding to the removed cluster centers. Formally we define Π to be the subspace orthogonal to all μ's removed up to this point (we overload notation by also using Π for the orthogonal projection onto this subspace). We will see that μ's are close to being orthogonal (see Lemma 3.7). This fact means that $\Pi\approx\text{span}(\{\mu_1,\ldots,\mu_b\})$, where $\{\mu_1,\ldots,\mu_b\}$ is the set of μ's that were not removed in the first step. Now the algorithm considers b candidate clusters where the condition for x being in a cluster i changes to:

$$\langle f_x,\Pi\mu_i\rangle \ge 0.9||\Pi\mu_i||^2 \text{ and for all } j\in[b], j\neq i\, \langle f_x,\Pi\mu_j\rangle < 0.9||\Pi\mu_j||^2.$$

Now we return all candidate clusters that satisfy $\phi(\widehat{C}_i)\le O(\epsilon)$ but this time the constant hidden in the O notation is bigger than in the first stage. In general at any stage t we change the test to $O(\epsilon\cdot t)$. At the end of the stage we proceed in a similar fashion by returning the clusters, removing the corresponding vertices and μ's and considering a lower dimensional subspace of Π in the next stage.

The algorithm continues in such a fashion for $O(\log(k))$ stages. Thus for all returned clusters

$\widehat{C}_i$ it is true that there exists j such that[5]:

$$|\widehat{C}_i \triangle C_j| \leq O\big(\epsilon \log(k)\big) \cdot |C_j|.$$

Let's analyze how this algorithm works for the configuration presented in Section 3.3.2. In the first stage we have that, for all $i \neq \frac{1}{\epsilon}$, $\widehat{C}_i = C_i$ and moreover $|\widehat{C}_{1/\epsilon} \cap C_{1/\epsilon}| = (\frac{1+\epsilon}{2})\frac{n}{k}$. So all candidate cluster $\widehat{C}_i$ for $i \neq 1/\epsilon$ are returned but crucially this time (in contrast with the natural hyperplane partitioning) cluster $C_{1/\epsilon}$ is left untouched. Then directions $\{\mu_1, \dots, \mu_{1/\epsilon - 1}\}$ are projected out. In the second stage the algorithm considers only vertices from $C_{1/\epsilon}$ projected onto one dimensional subspace span$(\mu_{1/\epsilon})$ and recovers this cluster up to $O(\epsilon)$ error.

Because of the robustness property (3.12), to show that this algorithm works we only need to argue that at the end of $O(\log(k))$ stages k sets are returned. We do that by showing that in every stage at least half of the remaining clusters is recovered. It is done in Lemma 3.37 and crucially relies on the following fact. When the algorithm considers a subspace Π then the number of points in the union of sets:

$$\{x \in V : \langle f_x, \Pi\mu_i \rangle \geq 0.9||\Pi\mu_i||^2\} \cap \{x \in V : \langle f_x, \Pi\mu_j \rangle \geq 0.9||\Pi\mu_j||^2\},$$

for all $i, j \in [b], i \neq j$ is bounded by $O(\epsilon \cdot b \cdot \frac{n}{k})$ (see Lemma 3.36 and Remark 3.7). To prove that we observe that every point x in this intersections has big projection onto some two μ_i, μ_j from $\{\mu_1, \dots, \mu_b\}$. Then using the fact that μ's are close to being orthogonal we deduce that $\Pi \approx \text{span}(\{\mu_1, \dots, \mu_b\})$ this in particular means that $\Pi\mu_i \approx \mu_i$, $\Pi\mu_j \approx \mu_j$. Because of that f_x is abnormally far (further by a factor of $1/\epsilon$ with respect to the average) from it's center μ_x. Now applying (3.10) for an orthonormal basis of Π and summing the inequalities we get that that the number of points in the intersections is bounded by $O(\epsilon \cdot b \cdot \frac{n}{k})$. Having this bound we can argue that at least half of the remaining clusters is recovered as on average only $O(\epsilon \cdot \frac{n}{k})$ points from each cluster belong to the intersections. The formal argument is given in Section 3.6.3.

The use of subspaces is crucial for our approach. If we relied solely on the bounds on norms (i.e. bounds on $||f_x||$) we could only claim a recovery guarantee of $O(\epsilon k)$ per cluster. One of the reasons is that there can be $\Theta(\epsilon n)$ vertices of abnormally big norm and all of them can belong to one cluster (as it happens in the example from Section 3.3.2). The use of carefully crafted sequence of subspaces solves this issue as it allows to derive better bounds for the number of abnormal vertices in each stage. It is possible as we can show that the "variance of the distribution" of f_x's cannot concentrate on subspaces. This leads to an $O(\epsilon \log(k))$ error guarantee per cluster.

What remains is to remove the assumption that the cluster means μ_i are known to the algorithm. We show, using our tail bounds from Lemma 3.4, that a random sample of

[5] Note that this algorithm may not return a partition of the graph but only a collection of disjoint clusters. Later, in Section 3.6.6 in Proposition 3.3, we present a simple reduction that shows that an algorithm that guarantees (3.12) is enough to construct a clustering oracle that, as required by Definition 3.4, returns a partition. The high level idea is to assign the remaining vertices to clusters randomly.

$O(1/\epsilon \cdot k^3 \log k)$ points in every cluster is likely to concentrate around the mean. This allows us to take a $O(1/\epsilon \cdot k^4 \log k)$ size sample of points, guess in exponential (in $1/\epsilon \cdot k^4 \log^2 k$) time which points belong to which cluster, and ultimately find surrogates $\widehat{\mu}_i$ that are sufficiently close to the actual μ_i's for the analysis to go through. This part of the analysis is presented in Section 3.6.4. We also need a mechanism for testing if a set of approximate $\widehat{\mu}$'s induces (via our partitioning algorithm) a good clustering. We accomplish this goal by designing a simple sampling based tester that determines whether or not the clusters induced by a particular collection of candidate cluster means have the right size and outer conductance properties. See Section 3.6.5 for this part of the analysis.

To design our spectral clustering algorithm we need to perform tests like $\langle f_x, \Pi\mu\rangle \overset{?}{\geq} 0.9||\Pi\mu||_2^2$ for a given vertex x, a candidate cluster mean μ, and the projection matrix Π. Hence, we need tools to approximate $\langle f_x, \Pi\mu\rangle$ and $||\Pi\mu||_2^2$. As explained above, instead of exact cluster means i.e. μ we will perform the test for approximate cluster means i.e, $\widehat{\mu} = \frac{1}{|S|}\sum_{y\in S} f_y$, where S is a small subset S of sampled nodes. First observe that for any vertex x one can estimate $\langle f_x, \widehat{\mu}\rangle_{apx}$ as follows:

$$\langle f_x, \widehat{\mu}\rangle_{apx} = \frac{1}{|S|}\sum_{y\in S}\langle f_x, f_y\rangle_{apx}$$

where $\langle f_x, f_y\rangle_{apx}$ can be computed using (SPECTRALDOTPRODUCTORACLE) Algorithm 10. Next we will explain how to compute $\langle f_x, \widehat{\Pi} f_y\rangle_{apx}$ for $x, y \in V$. Recall that $\widehat{\Pi}$ is the subspace orthogonal to all $\widehat{\mu}$'s removed so far. Let $\{\widehat{\mu}_1, \ldots, \widehat{\mu}_r\}$ denote the set of removed cluster means, and let $X \in \mathbb{R}^{k\times r}$ denote a matrix whose columns are $\widehat{\mu}_i$'s. Therefore the projection matrix onto the span of $\{\widehat{\mu}_1, \ldots, \widehat{\mu}_r\}$ is given by $X(X^TX)^{-1}X$. Hence, we have $\widehat{\Pi} = I - X(X^TX)^{-1}X$ and we can compute $\langle f_x, \widehat{\Pi} f_y\rangle_{apx}$ as follows:

$$\langle f_x, \widehat{\Pi} f_y\rangle_{apx} = \langle f_x, f_y\rangle_{apx} - (f_x^T X)(X^TX)^{-1}(Xf_y).$$

Note that the i-th column of matrix X is $\widehat{\mu}_i$, thus $f_x^T X \in \mathbb{R}^r$ is a vector whose i-th entry can be computed by $\langle f_x, \widehat{\mu}_i\rangle_{apx}$. Moreover notice that $X^TX \in \mathbb{R}^{r\times r}$ is matrix such that its (i, j)-th entry can be computed by $\langle \widehat{\mu}_i, \widehat{\mu}_j\rangle_{apx}$. Therefore $(f_x^T X)$, (Xf_y) and $(X^TX)^{-1}$ all can be computed explicitly which let us compute $\langle f_x, \widehat{\Pi} f_y\rangle_{apx}$. Given the primitive to compute $\langle f_x, \widehat{\Pi} f_y\rangle_{apx}$ we are able to estimate $\langle f_x, \Pi(\mu)\rangle$ and $||\Pi(\mu)||_2^2$ as follows:

$$\langle f_x, \widehat{\Pi}\widehat{\mu}\rangle_{apx} := \frac{1}{|B|}\cdot\sum_{y\in B}\langle f_x, \widehat{\Pi} f_y\rangle_{apx},$$

$$\left\|\widehat{\Pi}\widehat{\mu}\right\|_{apx}^2 := \frac{1}{|B|}\cdot\sum_{x\in B}\langle f_x, \widehat{\Pi}\widehat{\mu}\rangle_{apx}.$$

This part of the analysis is presented in Section 3.5.6.

3.4 Properties of the spectral embedding of (k, φ, ϵ)-clusterable graphs

In this section we study the spectral embedding of (k, φ, ϵ)-clusterable graphs. Recall that the spectral embedding maps every vertex $x \in V$ to a k-dimensional vector f_x. We are interested in understanding the geometric properties of this embedding. We start by recalling some standard properties of the embedding: We show that the cluster means

$$\mu_i = \frac{1}{|C_i|} \sum_{x \in C_i} f_x$$

are almost orthogonal and of length roughly $1/\sqrt{|C_i|}$ (Lemma 3.7 below). Then we give a bound on the directional variance, by which we mean the sum of squared distances of points f_x to their corresponding cluster centers when projected on direction α. We show in Lemma 3.6 below that the directional variance is bounded by $O(\epsilon/\varphi^2)$ for every direction $\alpha \in \mathbb{R}^k$, $\|\alpha\| = 1$. This in particular implies (see Lemma 3.9 below) that 'rounding' the spectral embedding by mapping each vertex to its corresponding cluster center results in a matrix U that spectrally approximates the matrix of bottom k eigenvectors of the Laplacian. These bounds are rather standard, and their proofs are provided for completeness. The main shortcoming of the standard bounds is that they can only allow us to apply averaging arguments, and are thus unable to rule out that some of the embedded points are quite far away from their corresponding cluster center. For example, they do not rule out the possibility of an $\Omega(1/k)$ fraction of the points being $\approx \sqrt{k}$ further away from their corresponding centers. Since we would like to recover every cluster to up an $O(\epsilon)$ error, such bounds are not sufficient on their own.

For this reason we consider the distribution of the projection of the embedded points on the direction of any of the first k eigenvectors and we give stronger tail bounds for these distributions (in Lemma 3.4) than what follows from variance calculations only. Basically, we give a strong bound on the $O(\varphi^2/\epsilon)$-th moment of the spectral embedding as opposed to just on the second moment, as above. These higher moment bounds are then crucially used to achieve sublinear time access to dot products in the embedded space in Section 3.5 (we need them to establish spectral concentration of a small number of random samples in Section 3.5.2) as well as to argue that a small sample of vertices contains a good approximation to the true cluster means $\mu_i, i = 1, \ldots, k$ in its span in Section 3.6.4.

3.4.1 Standard bounds on cluster means and directional variance

The lemma below bounds the variance of the spectral embedding in any direction.

Lemma 3.6. *(Variance bounds) Let $k \geq 2$ be an integer, $\varphi \in (0,1)$, and $\epsilon \in (0,1)$. Let $G = (V, E)$ be a d-regular graph that admits (k, φ, ϵ)-clustering $C_1, \ldots, C_k$. Then for all $\alpha \in \mathbb{R}^k$, with $\|\alpha\| = 1$ we have*

$$\sum_{i=1}^{k} \sum_{x \in C_i} \langle f_x - \mu_i, \alpha \rangle^2 \leq \frac{4\epsilon}{\varphi^2}.$$

Proof. For each $i \in [k]$, and any vertex $x \in C_i$, let $d_i(x)$ denote the degree of vertex x in the subgraph C_i. Let H_i be a graph obtained by adding $d - d_i(x)$ self-loops to each vertex $x \in C_i$. Let L denote the normalized Laplacian of graph G. For each $i \in [k]$ and let L_i denote the normalized Laplacian of H_i, and let $\lambda_2(H_i)$ be the second smallest eigenvalue of L_i.

Let $z = U_{[k]}\alpha$. Note that $||z||_2 = 1$. By Lemma 3.3 we have $\lambda_1 \le \ldots \le \lambda_k \le 2\epsilon$, where λ_i is the i^{th} smallest eigenvalue of of L. Therefore we have

$$\langle z, Lz\rangle \le \lambda_k \le 2\epsilon \tag{3.14}$$

Fix some $i \in [k]$, let $z' \in \mathbb{R}^n$ be a vector such that $z'(x) := z(x) - \langle \mu_i, \alpha\rangle$. For any $S \subseteq V$, we define $z'_S \in \mathbb{R}^n$ to be a vector such that for all $x \in V$ $z'_S(x) = z'(x)$ if $x \in S$ and $z'_S(x) = 0$ otherwise. Note that $z(x) = \left\langle f_x, \alpha\right\rangle$, thus we have

$$\sum_{x\in V} z'_{C_i}(x) = \sum_{x\in C_i} z'(x) = \sum_{x\in C_i} z(x) - \left\langle \mu_i, \alpha\right\rangle = \sum_{x\in C_i} \left\langle f_x - \mu_i, \alpha\right\rangle = 0$$

Thus we have $z'_{C_i} \perp \mathbb{1}$, so by properties of Rayleigh quotient we get

$$\frac{\langle z'_{C_i}, L_i z'_{C_i}\rangle}{\langle z'_{C_i}, z'_{C_i}\rangle} = \frac{1}{d}\frac{\sum_{x,y\in C_i,(x,y)\in E}(z'(x) - z'(y))^2}{\sum_{x\in C_i}(z'(x))^2} = \frac{1}{d}\frac{\sum_{x,y\in C_i,(x,y)\in E}(z(x) - z(y))^2}{\sum_{x\in C_i}(z(x) - \langle \mu_i, \alpha\rangle)^2} \ge \lambda_2(H_i) \tag{3.15}$$

Furthermore, by Cheeger's inequality for any $i \in [k]$ we have $\lambda_2(H_i) \ge \frac{\varphi^2}{2}$. Hence, for any $i \in [k]$ we have

$$\frac{\sum_{x,y\in C_i,(x,y)\in E}(z(x) - z(y))^2}{d\sum_{x\in C_i}(z(x) - \langle \mu_i, \alpha\rangle)^2} \ge \lambda_2(H_i) \ge \frac{\varphi^2}{2}$$

Now observe the following:

$$\begin{aligned}
2\epsilon &\ge \langle z, Lz\rangle && \text{By (3.14)}\\
&= \frac{1}{d}\cdot \sum_{(x,y)\in E}(z(x) - z(y))^2\\
&\ge \frac{1}{d}\cdot \sum_{i=1}^{k}\sum_{x,y\in C_i,(x,y)\in E}(z(x) - z(y))^2\\
&\ge \frac{\varphi^2}{2}\cdot \sum_{i=1}^{k}\sum_{x\in C_i}(z(x) - \langle \mu_i, \alpha\rangle)^2 && \text{By (3.15)}
\end{aligned}$$

Recall that for all $x \in V$, $z(x) = \left\langle f_x, \alpha\right\rangle$. Therefore for for any $\alpha \in \mathbb{R}^k$ with $\|\alpha\| = 1$ we have

$$\sum_{i=1}^{k}\sum_{x\in C_i}\langle f_x - \mu_i, \alpha\rangle^2 \le \frac{4\epsilon}{\varphi^2}$$

□

The following lemma shows that the length of the cluster mean of cluster C_i is roughly $1/\sqrt{|C_i|}$

and that cluster means are almost orthogonal.

Lemma 3.7. *(Cluster means) Let $k \ge 2$ be an integer, $\varphi \in (0,1)$, and $\epsilon \in (0,1)$. Let $G=(V,E)$ be a d-regular graph that admits (k,φ,ϵ)-clustering $C_1,\ldots,C_k$. Then we have*

1. *for all $i \in [k]$, $\left| ||\mu_i||_2^2 - \frac{1}{|C_i|}\right| \le \frac{4\sqrt{\epsilon}}{\varphi}\frac{1}{|C_i|}$*
2. *for all $i \neq j \in [k]$, $\left|\langle \mu_i,\mu_j\rangle\right| \le \frac{8\sqrt{\epsilon}}{\varphi}\frac{1}{\sqrt{|C_i||C_j|}}$*

To prove Lemma 3.7 we need Lemma 3.9 in which we will use the following result from [HJ90] (Theorem 1.3.20 on page 53).

Lemma 3.8 ([HJ90])**.** *Let h,m,n be integers such that $1 \le h \le m \le n$. For any matrix $A \in \mathbb{R}^{m\times n}$ and matrix $B \in \mathbb{R}^{n\times m}$, the multisets of nonzero eigenvalues of AB and BA are equal. In particular, if one of AB and BA is positive semidefinite, then $\nu_h(AB) = \nu_h(BA)$.*

Lemma 3.9. *Let $k \ge 2$ be an integer, $\varphi \in (0,1)$, and $\epsilon \in (0,1)$. Let $G=(V,E)$ be a d-regular graph that admits (k,φ,ϵ)-clustering $C_1,\ldots,C_k$. Let $H \in \mathbb{R}^{k\times k}$ be a matrix whose i-th column is μ_i. Let $W \in \mathbb{R}^{k\times k}$ be a diagonal matrix such that $W(i,i) = \sqrt{|C_i|}$. Then for any $\alpha \in \mathbb{R}^k$, $\|\alpha\|=1$, we have*

1. *$|\alpha^T\left((HW)(HW)^T - I\right)\alpha| \le \frac{4\sqrt{\epsilon}}{\varphi}$*
2. *$|\alpha^T\left((HW)^T(HW) - I\right)\alpha| \le \frac{4\sqrt{\epsilon}}{\varphi}$*

Proof. **Proof of item** (1)**:** Let $Y \in \mathbb{R}^{k\times n}$ denote a matrix whose x-th column is μ_x for any $x \in V$. Note that

$$YY^T = \sum_{i=1}^{k} |C_i|\mu_i\mu_i^T = (HW)(HW)^T.$$

We define $\widetilde{z} := Y^T\alpha$, and $z := U_{[k]}\alpha$. Note that $U_{[k]}^T U_{[k]} = I$. Therefore we have

$$\begin{aligned}
|\alpha^T\left((HW)(HW)^T - I\right)\alpha| &= |\alpha^T(YY^T - U_{[k]}^T U_{[k]})\alpha| \\
&= \left|\sum_{x\in V}\widetilde{z}(x)^2 - z(x)^2\right| && \text{From definition of } z(x) \text{ and } \widetilde{z}(x) \\
&= \left|\sum_{x\in V}(z(x)-\widetilde{z}(x))\,(z(x)+\widetilde{z}(x))\right| \\
&\le \sqrt{\sum_{x\in V}(z(x)-\widetilde{z}(x))^2 \sum_{x\in V}(\widetilde{z}(x)+z(x))^2} && \text{By Cauchy-Schwarz inequality}
\end{aligned} \tag{3.16}$$

Note that for any $x \in V$, we have $z(x) = \langle f_x,\alpha\rangle$ and $\widetilde{z}(x) = \langle \mu_x,\alpha\rangle$. Therefore by Lemma 3.6 we have

$$\sqrt{\sum_{x\in V}(z(x)-\widetilde{z}(x))^2} = \sqrt{\sum_{x\in V}\langle f_x - \mu_x,\alpha\rangle^2} \le \frac{2\sqrt{\epsilon}}{\varphi} \tag{3.17}$$

To complete the proof it suffices to show that $\sum_{x\in V}(\tilde{z}(x)+z(x))^2 \le 4$. Note that

$$\begin{aligned}
\sum_{x\in V} \tilde{z}(x)^2 &= \sum_{x\in V} \left\langle \alpha, \mu_x \right\rangle^2 \\
&= \sum_i |C_i| \left\langle \alpha, \frac{\sum_{x\in C_i} f_x}{|C_i|} \right\rangle^2 \\
&= \sum_i |C_i| \left(\frac{\sum_{x\in C_i} \left\langle \alpha, f_x \right\rangle}{|C_i|} \right)^2 \\
&\le \sum_i \sum_{x\in C_i} \left\langle \alpha, f_x \right\rangle^2 && \text{By Jensen's inequality} \\
&= \sum_{x\in V} z(x)^2
\end{aligned}$$

Thus we have

$$\sum_{x\in V}(\tilde{z}(x)+z(x))^2 \le \sum_{x\in V} 2(\tilde{z}(x)^2+z(x)^2) \le 2+2\sum_{x\in V}\tilde{z}(x)^2 \le 4 \tag{3.18}$$

In the first inequality we used the fact that $(\tilde{z}(x)-z(x))^2 \ge 0$ and for the second inequality we used the fact that $||z||_2^2 = ||U_{[k]}\alpha||_2^2 = 1$. Putting (3.18), (3.17), and (3.16) together we get

$$|\alpha^T \left((HW)(HW)^T - I\right)\alpha| \le \frac{4\sqrt{\epsilon}}{\varphi}.$$

Proof of item (2)**:** Note that by item (2) for any vector α with $||\alpha||_2 = 1$ we have

$$1-\frac{4\sqrt{\epsilon}}{\varphi} \le \alpha^T \left((HW)(HW)^T\right)\alpha \le 1+\frac{4\sqrt{\epsilon}}{\varphi}$$

Thus by Lemma 3.8 we have that the set of eigenvalues of $(HW)(HW)^T$ and $(HW)^T(HW)$ are the same, and all of the eigenvalues lie in the interval $[1-\frac{4\sqrt{\epsilon}}{\varphi}, 1+\frac{4\sqrt{\epsilon}}{\varphi}]$. Thus for any vector α with $||\alpha||_2 = 1$ we have

$$1-\frac{4\sqrt{\epsilon}}{\varphi} \le \alpha^T \left((HW)^T(HW)\right)\alpha \le 1+\frac{4\sqrt{\epsilon}}{\varphi}.$$

□

Now we are able to prove Lemma 3.7.

Lemma 3.7. *(Cluster means) Let $k \ge 2$ be an integer, $\varphi \in (0,1)$, and $\epsilon \in (0,1)$. Let $G=(V,E)$ be a d-regular graph that admits (k,φ,ϵ)-clustering $C_1,\ldots,C_k$. Then we have*

1. *for all $i\in[k]$, $\left| ||\mu_i||_2^2 - \frac{1}{|C_i|}\right| \le \frac{4\sqrt{\epsilon}}{\varphi}\frac{1}{|C_i|}$*
2. *for all $i \ne j \in [k]$, $\left|\langle \mu_i, \mu_j\rangle\right| \le \frac{8\sqrt{\epsilon}}{\varphi}\frac{1}{\sqrt{|C_i||C_j|}}$*

Proof. **Proof of item** (1)**:** Let $H \in \mathbb{R}^{k\times k}$ be a matrix whose i-th column is μ_i. Let $W \in \mathbb{R}^{k\times k}$ be a diagonal matrix whose such that $W(i,i) = \sqrt{|C_i|}$. Thus by Lemma 3.9 item (2) for any $\alpha \in \mathbb{R}^k$ with $\|\alpha\| = 1$, we have

$$|\alpha^T\left((HW)^T(HW) - I\right)\alpha| \le \frac{4\sqrt{\epsilon}}{\varphi}$$

Let $\alpha = \mathbb{1}_i$. Thus we have

$$|((HW)^T(HW))(i,i) - 1| \le \frac{4\sqrt{\epsilon}}{\varphi} \tag{3.19}$$

Note that $((HW)^T(HW))(i,i) = (WH^THW)(i,i) = ||\mu_i||_2^2|C_i|$.Therefore we get

$$\left|||\mu_i||_2^2 - \frac{1}{|C_i|}\right| \le \frac{4\sqrt{\epsilon}}{\varphi}\cdot\frac{1}{|C_i|}$$

Proof of item (2)**:** Let $\alpha = \frac{1}{\sqrt{2}}(\mathbb{1}_i + \mathbb{1}_j)$. Note that $||\alpha||_2 = 1$. Thus by Lemma 3.9 item (2) we have

$$|\alpha^T\left((HW)^T(HW) - I\right)\alpha| \le \frac{4\sqrt{\epsilon}}{\varphi}$$

Note that

$$\left|\alpha^T\left((HW)^T(HW) - I\right)\alpha\right| = \left|\frac{1}{2}\left(||\mu_i||_2^2|C_i| + ||\mu_j||_2^2|C_j| + 2\left\langle\mu_i,\mu_j\right\rangle\sqrt{|C_i||C_j|} - 2\right)\right|$$

Therefore we get

$$\left|||\mu_i||_2^2|C_i| + ||\mu_j||_2^2|C_j| + 2\left\langle\mu_i,\mu_j\right\rangle\sqrt{|C_i||C_j|} - 2\right| \le \frac{8\sqrt{\epsilon}}{\varphi}$$

Thus

$$\begin{aligned}\left|\left\langle\mu_i,\mu_j\right\rangle\sqrt{|C_i||C_j|}\right| &\le \left|\frac{1}{2}\left(1 - ||\mu_i||_2^2|C_i|\right) + \frac{1}{2}\left(1 - ||\mu_j||_2^2|C_j|\right)\right| + \frac{4\sqrt{\epsilon}}{\varphi} \\ &\le \frac{1}{2}\cdot\frac{4\sqrt{\epsilon}}{\varphi} + \frac{1}{2}\cdot\frac{4\sqrt{\epsilon}}{\varphi} + \frac{4\sqrt{\epsilon}}{\varphi} && \text{By item (1)} \\ &\le \frac{8\sqrt{\epsilon}}{\varphi}\end{aligned}$$

Therefore we get

$$\left|\left\langle\mu_i,\mu_j\right\rangle\right| \le \frac{8\sqrt{\epsilon}}{\varphi}\cdot\frac{1}{\sqrt{|C_i||C_j|}}.$$

□

3.4.2 Strong Tail Bounds on the Spectral Embedding

The main results of this section are the following two lemmas. The first lemma gives an upper bound on the length of the projection of any point f_x on an arbitrary direction $\alpha \in \mathbb{R}^k$. The second lemma considers the distribution of the lengths of projected f_x and we get tail bounds that show that the fraction of points whose projected length exceeds the 'expectation' (which is about $1/\sqrt{|C_i|}$ for the smallest cluster C_i) by a factor of β is bounded by $\beta^{-\varphi^2/10\epsilon}$. In other words, we bound the $O(\varphi^2/\epsilon)$-th moment as opposed to the second moment, which gives us tight control over the embedding when $\epsilon/\varphi^2 \ll 1/\log k$.

Lemma 3.5. *Let $\varphi \in (0,1)$ and $\epsilon \leq \frac{\varphi^2}{100}$, and let $G = (V,E)$ be a d-regular graph that admits (k,φ,ϵ)-clustering $C_1,\ldots,C_k$. Let u be a normalized eigenvector of L with $||u||_2 = 1$ and with eigenvalue at most 2ϵ. Then we have*

$$||u||_\infty \leq n^{20\cdot\epsilon/\varphi^2} \cdot \sqrt{\frac{160}{\min_{i\in k}|C_i|}}.$$

Lemma 3.4. *[Tail-bound] Let $\varphi \in (0,1)$ and $\epsilon \leq \frac{\varphi^2}{100}$, and let $G = (V,E)$ be a d-regular graph that admits (k,φ,ϵ)-clustering $C_1,\ldots,C_k$. Let L be the normalized Laplacian of G. Let u be a normalized eigenvector of L with $||u||_2 = 1$ and with eigenvalue at most 2ϵ. Then for any $\beta > 1$ we have*

$$\frac{1}{n} \cdot \left| \left\{ x \in V : |u(x)| \geq \beta \cdot \sqrt{\frac{10}{\min_{i\in[k]}|C_i|}} \right\} \right| \leq \left(\frac{\beta}{2}\right)^{-\varphi^2/20\cdot\epsilon}.$$

We are interested in deriving moment bounds for the distribution of the entries of the first k eigenvectors u of L (i.e., eigenvectors with eigenvalue smaller than 2ϵ), and specifically in the distribution of the absolute values of the entries of u. In order to be able to analyze this distribution, we define the sets of all entries in u that are bigger than a threshold θ:

Definition 3.6 (Threshold sets)**.** Let $G = (V,E)$ be a graph with normalized Laplacian L. Let u be a normalized eigenvector of L with $||u||_2 = 1$. Then for the vector u and a threshold $\theta \in \mathbb{R}^+$ we define the threshold set $S(\theta)$ with respect to the eigenvector u and threshold θ as

$$S(\theta) := \{x \in V : u(x) \geq \theta\}.$$

Our arguments will use that for every vertex x, we have $u(x) \approx \frac{1}{d}\sum_{\{x,y\}\in E} u(y)$. So nodes neighboring other nodes with large $u(\cdot)$ values are likely to have large $u(\cdot)$ values as well. This motivates the following definition of the potential of a threshold set.

Definition 3.7 (Potential of a threshold set)**.** Let $G = (V,E)$ be a graph with normalized Laplacian L. Let u be a normalized eigenvector of L with $||u||_2 = 1$. Then for vector u and a threshold $\theta \in \mathbb{R}^+$ we define the potential of a threshold set $S(\theta)$ as

$$p(\theta) = \sum_{x\in S(\theta)} u(x).$$

We start by proving a core bound on the threshold sets (Lemma 3.10 below) that forms the basis of our approach: the main technical results of this section (Lemma 3.5 and Lemma 3.4) essentially follow by repeated application of Lemma 3.10. Specifically, we now argue that if a threshold set $S(\theta)$ expands in the graph G and the relative potential of the set (i.e., $p(\theta)/|S(\theta)|$) is at most 2θ, then we can slightly decrease θ to obtain a new θ' such that the corresponding threshold set is a constant factor larger that $S(\theta)$ and the relative potential is bounded by $2\theta'$.

Lemma 3.10 (Threshold shift for expanding threshold sets)**.** *Let $G=(V,E)$ be a d-regular graph with normalized Laplacian L. Let u be a normalized eigenvector of L with $||u||_2 = 1$ and with eigenvalue $\lambda \le 2\epsilon$. Let $\theta \in \mathbb{R}^+$ be a threshold. Suppose that $S(\theta)$ is the threshold set with respect to u and θ such that $S(\theta)$ is non-empty, $\phi^G(S(\theta)) \ge \varphi$ and $\frac{p(\theta)}{|S(\theta)|} \le 2\theta$. Then the following holds for $\theta' = \theta\left(1-\frac{8\epsilon}{\varphi}\right)$:*

1. *$|S(\theta')| \ge (1+\varphi/2)|S(\theta)|$, and*
2. *$\frac{p(\theta')}{|S(\theta')|} \le 2\theta'$.*

Proof. **Proof of item** (1)**:** Note that $\lambda u = Lu = (I-\frac{A}{d})u$. Thus for any $x \in V$ we have $(Lu)(x) = u(x) - \frac{1}{d}\sum_{\{x,y\}\in E} u(y)$. Thus we have,

$$u(x) - \frac{1}{d}\sum_{\{x,y\}\in E} u(y) = \lambda \cdot u(x).$$

We write the above as

$$\sum_{y\in\mathcal{N}(x)} (u(x)-u(y)) = d\cdot\lambda\cdot u(x), \tag{3.20}$$

where $\mathcal{N}(x) = \{y \in V : \exists\{x,y\} \in E\}$. Summing (3.20) over all $x \in S(\theta)$ we get

$$\sum_{x\in S(\theta)}\sum_{y\in\mathcal{N}(x)} (u(x)-u(y)) = \sum_{x\in S(\theta)} \lambda\cdot d\cdot u(x) = \lambda\cdot d\cdot p(\theta), \tag{3.21}$$

and note that

$$\sum_{x\in S(\theta)}\sum_{y\in\mathcal{N}(x)} (u(x)-u(y)) = \sum_{\substack{\{x,y\}\in E\\ x\in S(\theta), y\notin S(\theta)}} (u(x)-u(y)). \tag{3.22}$$

For any edge $e=\{x,y\}\in E$, we define $\Delta(e) = |u(x)-u(y)|$. Note that for any $e=\{x,y\}$ such that $x\in S(\theta)$ and $y\notin S(\theta)$ we have $u(x)\ge\theta > u(y)$, hence $\Delta(e) = u(x)-u(y)$. Therefore, putting (3.22) and (3.21) together we get

$$\sum_{e\in E(S(\theta),V\setminus S(\theta))} \Delta(e) = \lambda\cdot d\cdot p(\theta).$$

By an averaging argument there exists a set $E_L \subseteq E(S_\theta, V\setminus S_\theta)$ such that $|E_L| \ge \frac{|E(S(\theta),V\setminus S(\theta))|}{2}$ and all edges $e\in E_L$ satisfy $\Delta(e) \le \frac{2\cdot\lambda\cdot d\cdot p(\theta)}{|E(S(\theta),V\setminus S(\theta))|}$. We define V_L as a subset of vertices of $V\setminus S(\theta)$ that are connected to vertices of $S(\theta)$ by edges in E_L, i.e.

$$V_L = \{y\in V\setminus S(\theta) : \exists\,\{x,y\}\in E_L, x\in S(\theta)\}.$$

Note that

$$|V_L| \geq \frac{|E_L|}{d} \geq \frac{|E(S(\theta), V \setminus S(\theta))|}{2d}. \tag{3.23}$$

Using the assumption of the lemma that $\phi^G(S(\theta)) \geq \varphi$ we obtain

$$|E(S(\theta), V \setminus S(\theta))| \geq \varphi \cdot d \cdot |S(\theta)|. \tag{3.24}$$

Putting (3.24) and (3.23) together we get

$$|V_L| \geq \frac{\varphi |S(\theta)|}{2}. \tag{3.25}$$

Recall that for all $e \in E_L$ we have $\Delta(e) \leq \frac{2 \cdot \lambda \cdot d \cdot p(\theta)}{|E(S(\theta), V \setminus S(\theta))|}$. We have $\lambda \leq 2\epsilon$, therefore for all $e \in E_L$ we have $\Delta(e) \leq \frac{4 \cdot \epsilon \cdot d \cdot p(\theta)}{|E(S(\theta), V \setminus S(\theta))|}$. Thus for all $y \in V_L$ we get

$$u(y) \geq \theta - \frac{4 \cdot \epsilon \cdot d \cdot p(\theta)}{|E(S(\theta), V \setminus S(\theta))|}. \tag{3.26}$$

By the assumption of the lemma we have $\frac{p(\theta)}{|S(\theta)|} \leq 2\theta$, hence, by inequality (3.24) we get

$$\theta - \frac{4 \cdot \epsilon \cdot d \cdot p(\theta)}{|E(S(\theta), V \setminus S(\theta))|} \geq \theta - \frac{4 \cdot \epsilon \cdot d \cdot p(\theta)}{\varphi \cdot d \cdot |S(\theta)|} = \theta - \frac{4\epsilon}{\varphi} \cdot \frac{p(\theta)}{|S(\theta)|} \geq \theta \left(1 - \frac{8\epsilon}{\varphi}\right). \tag{3.27}$$

Putting (3.27) and (3.26) together we get for all $y \in V_L$, $u(y) \geq \theta \left(1 - \frac{8\epsilon}{\varphi}\right)$. Let $\theta' := \theta(1 - \frac{8\epsilon}{\varphi})$. Thus

$$S(\theta) \cup V_L \subseteq S(\theta').$$

By definition of V_L we have $V_L \cap S(\theta) = \emptyset$. Therefore, $|S(\theta')| \geq |S(\theta)| + |V_L|$. Thus by inequality (3.25) we get

$$|S(\theta')| \geq |S(\theta)| \left(1 + \frac{\varphi}{2}\right). \tag{3.28}$$

This concludes the proof of the first part of the lemma.

Proof of item (2)**:** Now using that for all $x \notin S(\theta)$ we have $u(x) < \theta$ and that $p(\theta) \leq 2\theta |S(\theta)|$ by assumption of the lemma we obtain

$$\begin{aligned} p(\theta') &= \sum_{u \in S(\theta')} u(x) \\ &= \sum_{x \in S(\theta)} u(x) + \sum_{x \in S(\theta') \setminus S(\theta)} u(x) \\ &\leq p(\theta) + \theta |S(\theta') \setminus S(\theta)| \\ &\leq 2\theta |S(\theta)| + \theta |S(\theta') \setminus S(\theta)|. && \text{Since } p(\theta) \leq 2\theta |S(\theta)| \end{aligned}$$

By (3.28) we have $|S(\theta') \setminus S(\theta)| \geq \frac{\varphi}{2}|S(\theta)|$. Therefore, using $\epsilon \leq \frac{\varphi^2}{100}$ we get

$$\frac{p(\theta')}{|S(\theta')|} \leq \frac{2\theta|S(\theta)| + \theta|S(\theta') \setminus S(\theta)|}{|S(\theta)| + |S(\theta') \setminus S(\theta)|} = \theta \cdot \frac{2 + \frac{|S(\theta')\setminus S(\theta)|}{|S(\theta)|}}{1 + \frac{|S(\theta')\setminus S(\theta)|}{|S(\theta)|}} \leq \theta \cdot \frac{2 + \frac{\varphi}{2}}{1 + \frac{\varphi}{2}} \leq \theta \cdot 2\left(1 - \frac{8\epsilon}{\varphi}\right) \leq 2\theta'$$

□

We would like to apply Lemma 3.10 iteratively, but there is one hurdle: while the first condition on the threshold set $S(\theta)$ naturally follows as long as $S(\theta)$ is not too large (by Proposition 3.2), the second condition needs to be established at the beginning of the iterative process. Lemma 3.11 accomplishes exactly that: we prove that for any value θ_1 with threshold set $S(\theta_1)$ not empty or not too large, there exists a close value θ that meets the conditions of previous lemma.

Proposition 3.2. *Let $G = (V,E)$ be a d-regular graph that admits a (k,φ,ϵ)-clustering $C_1,\ldots,C_k$. For any set $S \subseteq V$ with size $|S| \leq \frac{1}{2} \cdot \min_{i\in k}|C_i|$ we have $\phi^G(S) \geq \varphi$.*

Proof. For any $1 \leq i \leq k$ we define $S_i = S \cap C_i$. Note that

$$|S_i| \leq |S| \leq \frac{1}{2} \cdot \min_{i\in k}|C_i| \leq \frac{|C_i|}{2}.$$

Therefore since $\phi^G(C_i) \geq \varphi$ we have $E(S_i, C_i \setminus S_i) \geq \varphi d|S_i|$. Thus we get

$$E(S, V\setminus S) \geq \sum_{i=1}^{k} E(S_i, C_i \setminus S_i) \geq \varphi d \sum_{i=1}^{k} |S_i| = \varphi d|S|.$$

Hence, $\phi^G(S) \geq \varphi$. □

Lemma 3.11. *Let $\varphi \in (0,1)$ and $\epsilon \leq \frac{\varphi^2}{100}$, and let $G = (V,E)$ be a d-regular graph that admits (k,φ,ϵ)-clustering $C_1,\ldots,C_k$. Let L denote the normalized Laplacian of G. Let u be a normalized eigenvector of L with $||u||_2 = 1$ and with eigenvalue $\lambda \leq 2\epsilon$. Let $\theta_1 \in \mathbb{R}^+$ be a threshold. Let $S(\theta_1)$ be the threshold set with respect to u and θ_1. Suppose that $1 \leq |S(\theta_1)| \leq \frac{1}{2} \cdot \min_{i\in\{1,\ldots,k\}}|C_i|$. Then there exists a threshold θ_2 such that the following holds:*

1. $\theta_1\left(1 - \frac{8\epsilon}{\varphi}\right) \leq \theta_2 \leq \theta_1$, *and*
2. $\frac{p(\theta_2)}{|S(\theta_2)|} \leq 2\theta_2$

Proof. Let

$$\theta^* := \min\left\{\theta \geq \theta_1 \,\middle|\, S(\theta) \neq \emptyset \text{ and } \frac{p(\theta)}{|S(\theta)|} \leq 2\theta\right\}.$$

We can conclude that θ^* exists, as by the assumption of the lemma we have $|S(\theta_1)| \geq 1$ and for $\theta_{\max} = \max_{x \in V} u(x)$ we have $\frac{p(\theta_{\max})}{|S(\theta_{\max})|} = \theta_{\max}$. We also have $|S(\theta^*)| \leq \min_{i \in \{1,\ldots,k\}} |C_i|/2$ as $\theta^* \geq \theta_1$ and by the assumption of the lemma. So Proposition 3.2 implies

$$\phi^G(S(\theta^*)) \geq \varphi. \tag{3.29}$$

Now Lemma 3.10 implies

$$\frac{p(\theta^*(1-\frac{8\epsilon}{\varphi}))}{|S(\theta^*(1-\frac{8\epsilon}{\varphi}))|} \leq 2\theta^*\left(1-\frac{8\epsilon}{\varphi}\right)$$

and by minimality of θ^* we have that:

$$\theta_1\left(1-\frac{8\epsilon}{\varphi}\right) \leq \theta^*\left(1-\frac{8\epsilon}{\varphi}\right) \leq \theta_1.$$

So we can set $\theta_2 := \theta^*\left(1-\frac{8\epsilon}{\varphi}\right)$. □

We are now ready to prove our tail bound. The main idea behind the proof is to use Lemma 3.10 and Lemma 3.11 to show that if a vertex has a large entry along one of the bottom k eigenvectors this implies that many other vertices also have a relatively large value along the same eigenvector. Thus, not too many f_x can have such a large value.

Lemma 3.4. *[Tail-bound] Let $\varphi \in (0,1)$ and $\epsilon \leq \frac{\varphi^2}{100}$, and let $G=(V,E)$ be a d-regular graph that admits (k,φ,ϵ)-clustering $C_1,\ldots,C_k$. Let L be the normalized Laplacian of G. Let u be a normalized eigenvector of L with $||u||_2 = 1$ and with eigenvalue at most 2ϵ. Then for any $\beta > 1$ we have*

$$\frac{1}{n}\cdot\left|\left\{x \in V : |u(x)| \geq \beta \cdot \sqrt{\frac{10}{\min_{i \in [k]} |C_i|}}\right\}\right| \leq \left(\frac{\beta}{2}\right)^{-\varphi^2/20\cdot\epsilon}.$$

Proof. Let $s_{\min} = \min_{i \in \{1,\ldots,k\}} |C_i|$. We define

$$S^+ = \left\{x \in V : u(x) \geq \beta \cdot \sqrt{\frac{10}{s_{\min}}}\right\},$$

and

$$S^- = \left\{x \in V : -u(x) \geq \beta \cdot \sqrt{\frac{10}{s_{\min}}}\right\}$$

Note that $-u$ is also an eigenvector of L with the same eigenvalue as u, hence, without loss of generality suppose that $|S^+| \geq |S^-|$. Let $T = \left\{x \in V : u(x)^2 \geq \frac{10}{s_{\min}}\right\}$. Since, $1 = \|u\|_2^2 = \sum_{x \in V} u(x)^2$, an averaging argument implies $|T| \leq \frac{s_{\min}}{10}$. Let

$$T^+ = \left\{x \in V : u(x) \geq \sqrt{\frac{10}{s_{\min}}}\right\}.$$

Note that $\beta > 1$, hence, $S^+ \subseteq T^+ \subseteq T$, and so we have $|S^+| \le |T^+| \le |T| \le \frac{s_{\min}}{10}$. We may assume that S^+ is non-empty as otherwise the lemma follows immediately. Let $\theta_0 = \beta \cdot \sqrt{\frac{10}{s_{\min}}}$. Note that $S^+ = S(\theta_0)$. Hence, $1 \le |S(\theta_0)| \le \frac{s_{\min}}{10}$. Therefore by Lemma 3.11 there exists a threshold θ_1 such that

$$\left(1-\frac{8\epsilon}{\varphi}\right)\beta\cdot\sqrt{\frac{10}{s_{\min}}} \le \theta_1 \le \beta\cdot\sqrt{\frac{10}{s_{\min}}}, \text{ and} \tag{3.30}$$

$$\frac{p(\theta_1)}{|S(\theta_1)|} \le 2\theta_1.$$

For any $t \ge 1$ we define $\theta_{t+1} = \theta_t(1-\frac{8\epsilon}{\varphi})$. For some $t' \ge 0$ we must have $\theta_{t'+1} \le \sqrt{\frac{10}{s_{\min}}} \le \theta_{t'}$. Thus by (3.30) we have

$$\theta_{t'} = \left(1-\frac{8\epsilon}{\varphi}\right)^{t'-1}\theta_1 \ge \left(1-\frac{8\epsilon}{\varphi}\right)^{t'}\cdot\beta\cdot\sqrt{\frac{10}{s_{\min}}}, \tag{3.31}$$

and

$$\theta_{t'} \le \frac{\theta_{t'+1}}{\left(1-\frac{8\epsilon}{\varphi}\right)} \le \frac{\sqrt{\frac{10}{s_{\min}}}}{\left(1-\frac{8\epsilon}{\varphi}\right)} \tag{3.32}$$

Putting (3.31) and (3.32) together we get

$$\beta \le \left(1-\frac{8\epsilon}{\varphi}\right)^{-t'-1} \tag{3.33}$$

Recall that for all $t \ge 1$ we have $\theta_{t+1} = \theta_t(1-\frac{8\epsilon}{\varphi})$, thus

$$S^+ = S(\theta_0) \subseteq S(\theta_1) \subseteq S(\theta_2) \subseteq \ldots \subseteq S(\theta_{t'}) \subseteq T^+.$$

Therefore for all $0 \le t \le t'$ we have

$$|S^+| \le |S(\theta_t)| \le |T^+| \le \frac{s_{\min}}{10}. \tag{3.34}$$

Since $|S(\theta_t)| \le \frac{\min_{i\in\{1,\ldots,k\}}|C_i|}{10} = \frac{s_{\min}}{10}$, by Lemma 3.10 for all $1 \le t \le t'$ we have

$$|S(\theta_{t+1})| \ge |S(\theta_t)|\left(1+\frac{\varphi}{2}\right). \tag{3.35}$$

Therefore

$$\begin{aligned}
t' &\le \log_{1+\frac{\varphi}{2}}\left(\frac{|T^+|}{|S^+|}\right) && \text{By (3.35)}\\
&\le \log_{1+\frac{\varphi}{2}}\left(\frac{s_{\min}}{10\cdot|S^+|}\right) && \text{By (3.34)}\\
&\le \log_{1+\frac{\varphi}{2}}\left(\frac{s_{\min}}{5\cdot|S^+\cup S^-|}\right) && \text{By the assumption } |S^+| \ge |S^-| \qquad (3.36)
\end{aligned}$$

Putting (3.33) and (3.36) together we get

$$\begin{aligned}
\beta &\le \left(1-\frac{8\epsilon}{\varphi}\right)^{-t'-1} && \text{By (3.33)}\\
&\le \left(1-\frac{8\epsilon}{\varphi}\right)^{-1-\log_{1+\frac{\varphi}{2}}\left(\frac{s_{\min}}{5\cdot|S^+\cup S^-|}\right)} && \text{By (3.36)}\\
&\le 2\cdot\left(\frac{s_{\min}}{5\cdot|S^+\cup S^-|}\right)^{-\log_{1+\frac{\varphi}{2}}\left(1-\frac{8\epsilon}{\varphi}\right)} && \text{Since } \frac{\epsilon}{\varphi^2}\le\frac{1}{100} \qquad (3.37)
\end{aligned}$$

Note that for any $x\in\mathbb{R}$ we have $1+x\le e^x$, and for any $x<0.01$ we have $1-x\ge e^{-1.2x}$, thus given $\frac{\epsilon}{\varphi}<0.01$ we have

$$\log_{1+\frac{\varphi}{2}}\left(1-\frac{8\epsilon}{\varphi}\right)=\frac{\ln\left(1-\frac{8\epsilon}{\varphi}\right)}{\ln\left(1+\frac{\varphi}{2}\right)}\ge\frac{-\frac{10\epsilon}{\varphi}}{\frac{\varphi}{2}}\ge-\frac{20\cdot\epsilon}{\varphi^2} \tag{3.38}$$

Putting (3.37) and (3.38) together we get

$$\frac{\beta}{2}\le\left(\frac{s_{\min}}{5\cdot|S^+\cup S^-|}\right)^{(20\cdot\epsilon/\varphi^2)}$$

Therefore we have

$$|S^+\cup S^-|\le s_{\min}\cdot\left(\frac{\beta}{2}\right)^{-(\varphi^2/20\cdot\epsilon)}\le n\cdot\left(\frac{\beta}{2}\right)^{-(\varphi^2/20\cdot\epsilon)}.$$

□

As a consequence of our tail bound we can prove a bound on ℓ_∞-norm on any unit vector in the eigenspace spanned by the bottom k eigenvectors of L, i.e. $U_{[k]}$.

Lemma 3.5. *Let $\varphi\in(0,1)$ and $\epsilon\le\frac{\varphi^2}{100}$, and let $G=(V,E)$ be a d-regular graph that admits (k,φ,ϵ)-clustering $C_1,\dots,C_k$. Let u be a normalized eigenvector of L with $||u||_2=1$ and with eigenvalue at most 2ϵ. Then we have*

$$||u||_\infty\le n^{20\cdot\epsilon/\varphi^2}\cdot\sqrt{\frac{160}{\min_{i\in k}|C_i|}}.$$

Proof. We define

$$S=\left\{x\in V:|u(x)|\ge n^{20\epsilon/\varphi^2}\cdot\sqrt{\frac{160}{\min_{i\in k}|C_i|}}\right\}$$

Let $\beta = 4 \cdot n^{20\epsilon/\varphi^2}$. By Lemma 3.4 we have

$$|S| \le n \cdot \left(\frac{\beta}{2}\right)^{-\varphi^2/20\cdot\epsilon} \le n \cdot \left(2 \cdot n^{20\epsilon/\varphi^2}\right)^{-\varphi^2/20\cdot\epsilon} < 1$$

Therefore $S = \emptyset$, hence

$$\|u\|_\infty \le n^{20\epsilon/\varphi^2} \cdot \sqrt{\frac{160}{\min_{i\in k}|C_i|}}.$$

□

3.4.3 Centers are strongly orthogonal

The main result of this section is Lemma 3.12 which generalizes Lemma 3.7 to the orthogonal projection of cluster centers into the subspace spanned by some of the centers. To prove Lemma 3.12 we first need to prove Lemma 3.13, Lemma 3.14 and Lemma 3.15.

Lemma 3.12. *Let $k \ge 2$, $\varphi \in (0,1)$ and $\frac{\epsilon}{\varphi^2}$ be smaller than an absolute positive constant. Let $G = (V,E)$ be a d-regular graph that admits (k,φ,ϵ)-clustering $C_1,\dots,C_k$. Let $S \subset \{\mu_1,\dots,\mu_k\}$ denote a subset of cluster means. Let $\Pi \in \mathbb{R}^{k\times k}$ denote the orthogonal projection matrix onto $span(S)^\perp$. Then the following holds:*

1. *For all $\mu_i \in \{\mu_1,\dots,\mu_k\} \setminus S$ we have $\left|\|\Pi\mu_i\|_2^2 - ||\mu_i||_2^2\right| \le \frac{16\sqrt{\epsilon}}{\varphi} \cdot ||\mu_i||_2^2$.*

2. *For all $\mu_i \neq \mu_j \in \{\mu_1,\dots,\mu_k\} \setminus S$ we have $|\langle\Pi\mu_i, \Pi\mu_j\rangle| \le \frac{40\sqrt{\epsilon}}{\varphi} \cdot \frac{1}{\sqrt{|C_i|\cdot|C_j|}}$.*

Matrix $A \in \mathbb{R}^n$ is poitive definite if $x^TAx > 0$ for all $x \neq 0$, and it is positive semidefinite if $x^TAx \ge 0$ for all $x \in \mathbb{R}^n$. We write $A \succ 0$ to indicate that A is positive definite, and $A \succcurlyeq 0$ to indicate that it is positive semidefinite. We use the semidefinite ordering on matrices, writing $A \succcurlyeq B$ if and only if $A - B \succcurlyeq 0$.

Theorem 3.4 ([Tod11])**.** *Let $A, B \in \mathbb{R}^{n\times n}$ be invertible, positive definite matrices. Then $A \succcurlyeq B \implies B^{-1} \succcurlyeq A^{-1}$.*

Proof. By symmetry, we only need to show $A \succcurlyeq B \implies B^{-1} \succcurlyeq A^{-1}$. Since $B \succ 0$ for any $x, y \in \mathbb{R}^n$ we obtain

$$\begin{aligned} 0 &\le \langle y - B^{-1}x, B(y - B^{-1}x)\rangle \\ &= \langle y, By\rangle - \langle y, x\rangle - \langle B^{-1}x, By\rangle + \langle x, B^{-1}x\rangle \\ &= \langle y, By\rangle - 2\langle x, y\rangle + \langle x, B^{-1}x\rangle \end{aligned}$$

so

$$2\langle x, y\rangle - \langle y, By\rangle \le \langle x, B^{-1}x\rangle \tag{3.39}$$

Since $A \succcurlyeq B$ it follows from (3.39) that

$$2\langle x,y\rangle - \langle y,Ay\rangle \leq 2\langle x,y\rangle - \langle y,Ay\rangle \leq \langle x,B^{-1}x\rangle \tag{3.40}$$

Letting $y = A^{-1}x$ in the leftmost expression of (3.40) we obtain

$$\langle x,A^{-1}x\rangle \leq \langle x,B^{-1}x\rangle$$

Since $x \in \mathbb{R}^n$ is is arbitrary, we get $B^{-1} \succcurlyeq A^{-1}$. □

Lemma 3.13. *Let $H, \widetilde{H} \in \mathbb{R}^{n\times n}$ be invertible, positive definite matrices. Let $\delta < 1$. Suppose that for any vector $x \in \mathbb{R}^n$ with $\|x\|_2 = 1$ we have $(1-\delta)x^T Hx \leq x^T \widetilde{H} x \leq (1+\delta)x^T Hx$. Then for any vector $y \in \mathbb{R}^n$ with $\|y\|_2 = 1$ we have $\frac{1}{1+\delta} y^T H^{-1} y \leq y^T \widetilde{H}^{-1} y \leq \frac{1}{1-\delta} y^T H^{-1} y$.*

Proof. Note that we have $(1-\delta)H \preceq \widetilde{H} \preceq (1+\delta)H$ therefore, by Theorem 3.4 we have

$$\frac{1}{(1-\delta)} \cdot H^{-1} \succcurlyeq \widetilde{H}^{-1} \succcurlyeq \frac{1}{(1+\delta)} \cdot H^{-1}$$

□

Lemma 3.14. *Let $k \geq 2$, $\varphi \in (0,1)$ and $\frac{\epsilon}{\varphi^2}$ be smaller than an absolute positive constant. Let $G = (V,E)$ be a d-regular graph that admits (k,φ,ϵ)-clustering $C_1,\dots,C_k$. Let $S = \{\mu_1,\dots,\mu_k\} \setminus \{\mu_i\}$. Let $H = [\mu_1,\mu_2,\dots,\mu_{i-1},\mu_{i+1},\dots,\mu_k]$ denote a matrix such that its columns are the vectors in S. Let $W \in \mathbb{R}^{(k-1)\times(k-1)}$ denote a diagonal matrix such that for all $j < i$ we have $W(j,j) = \sqrt{|C_j|}$ and for all $j \geq i$ we have $W(j,j) = \sqrt{|C_{j+1}|}$. Let $Z = HW$. Then $Z^T Z$ is invertible, and for any vector $x \in \mathbb{R}^{k-1}$ with $||x||_2 = 1$ we have*

$$|x^T((Z^T Z)^{-1} - I)x| \leq \frac{5\sqrt{\epsilon}}{\varphi}.$$

Proof. Let $Y \in \mathbb{R}^{k\times k}$ be a matrix, whose i-th column is equal to $\sqrt{C_i} \cdot \mu_i$. By Lemma 3.9 item (2) for any vector $z \in \mathbb{R}^k$ with $||\alpha||_2 = 1$ we have

$$|\alpha^T(Y^T Y - I)\alpha| \leq \frac{4\sqrt{\epsilon}}{\varphi}$$

Let $x \in \mathbb{R}^{k-1}$ be a vector with $||x||_2 = 1$, and let $\alpha \in \mathbb{R}^k$ be a vector defined as follows:

$$\alpha_j = \begin{cases} x_j & j < i \\ 0 & j = i \\ x_{j+1} & j > i \end{cases}$$

Thus we have $||\alpha||_2 = ||x||_2 = 1$ and $Y\alpha = Zx$. Hence, we get

$$|x^T(Z^TZ - I)x| = |\alpha^T(Y^TY - I)\alpha| \leq \frac{4\sqrt{\epsilon}}{\varphi}$$

Thus for any vector $x \in \mathbb{R}^{k-1}$ with $||x||_2 = 1$ we have

$$1 - \frac{4\sqrt{\epsilon}}{\varphi} \leq x^T(Z^TZ)x \leq 1 + \frac{4\sqrt{\epsilon}}{\varphi}$$

Note that Z^TZ is symmetric and positive semidefinit. Also note that Z^TZ is spectrally close to I, hence, Z^TZ is invertible. Thus by Lemma 3.13 for any vector $x \in \mathbb{R}^{k-1}$ we have

$$1 - \frac{5\sqrt{\epsilon}}{\varphi} \leq x^T(Z^TZ)^{-1}x \leq 1 + \frac{5\sqrt{\epsilon}}{\varphi}$$

Therefore we get

$$|x^T((Z^TZ)^{-1} - I)x| \leq \frac{5\sqrt{\epsilon}}{\varphi}.$$

□

Lemma 3.15. *Let $k \geq 2$ be an integer, $\varphi \in (0,1)$, and $\epsilon \in (0,1)$. Let $G = (V,E)$ be a d-regular graph that admits (k,φ,ϵ)-clustering $C_1,\dots,C_k$. Let $S = \{\mu_1,\dots,\mu_k\}\backslash\{\mu_i\}$. Let $H = [\mu_1,\mu_2,\dots,\mu_{i-1},\mu_{i+1},\dots,\mu_k]$ denote a matrix such that its columns are the vectors in S. Let $W \in \mathbb{R}^{(k-1)\times(k-1)}$ denote a diagonal matrix such that for all $j < i$ we have $W(j,j) = \sqrt{|C_j|}$ and for all $j \geq i$ we have $W(j,j) = \sqrt{|C_{j+1}|}$. Let $Z = HW$. Then we have*

$$\mu_i^T ZZ^T \mu_i \leq \frac{8\sqrt{\epsilon}}{\varphi} \cdot ||\mu_i||_2^2.$$

Proof. Note that $ZZ^T = (\sum_{j=1}^k |C_j|\mu_j\mu_j^T) - |C_i|\mu_i\mu_i^T$. Thus we have

$$\mu_i^T ZZ^T \mu_i = \mu_i^T \left(\sum_{j=1}^k |C_j|\mu_j\mu_j^T\right)\mu_i - |C_i| \cdot ||\mu_i||_2^4. \tag{3.41}$$

By Lemma 3.9 item (1) for any vector x with $||x||_2 = 1$ we have

$$x^T\left(\sum_{j=1}^k |C_j|\mu_j\mu_j^T - I\right)x \leq \frac{4\sqrt{\epsilon}}{\varphi}$$

Hence we can write

$$\mu_i^T\left(\sum_{j=1}^k |C_j|\mu_j\mu_j^T\right)\mu_i = \mu_i^T\left(\sum_{j=1}^k |C_j|\mu_j\mu_j^T - I\right)\mu_i + \mu_i^T\mu_i \leq \left(1 + \frac{4\sqrt{\epsilon}}{\varphi}\right)||\mu_i||_2^2$$

Therefore by (3.41) we get

$$\begin{aligned}\mu_i^T ZZ^T \mu_i &= \mu_i^T \left(\sum_{j=1}^{k} |C_j| \mu_j \mu_j^T\right) \mu_i - |C_i| \cdot ||\mu_i||_2^4 \\ &\leq \left(1 + \frac{4\sqrt{\epsilon}}{\varphi} - |C_i| \cdot ||\mu_i||_2^2\right) ||\mu_i||_2^2\end{aligned}$$

By Lemma 3.7 we have $|C_i| \cdot ||\mu_i||_2^2 \geq \left(1 - \frac{4\sqrt{\epsilon}}{\varphi}\right)$. Thus we get

$$\begin{aligned}\mu_i^T ZZ^T \mu_i &\leq \left(1 + \frac{4\sqrt{\epsilon}}{\varphi} - |C_i| \cdot ||\mu_i||_2^2\right) ||\mu_i||_2^2 \\ &\leq \left(1 + \frac{4\sqrt{\epsilon}}{\varphi} - 1 + \frac{4\sqrt{\epsilon}}{\varphi}\right) ||\mu_i||_2^2 \\ &\leq \frac{8\sqrt{\epsilon}}{\varphi} \cdot ||\mu_i||_2^2\end{aligned}$$

□

Now we prove the main result of the subsection (Lemma 3.12).

Lemma 3.12. *Let $k \geq 2$, $\varphi \in (0,1)$ and $\frac{\epsilon}{\varphi^2}$ be smaller than an absolute positive constant. Let $G = (V,E)$ be a d-regular graph that admits (k,φ,ϵ)-clustering $C_1,\ldots,C_k$. Let $S \subset \{\mu_1,\ldots,\mu_k\}$ denote a subset of cluster means. Let $\Pi \in \mathbb{R}^{k\times k}$ denote the orthogonal projection matrix onto $span(S)^\perp$. Then the following holds:*

1. *For all $\mu_i \in \{\mu_1,\ldots,\mu_k\} \setminus S$ we have $\left| ||\Pi\mu_i||_2^2 - ||\mu_i||_2^2 \right| \leq \frac{16\sqrt{\epsilon}}{\varphi} \cdot ||\mu_i||_2^2$.*

2. *For all $\mu_i \neq \mu_j \in \{\mu_1,\ldots,\mu_k\} \setminus S$ we have $|\langle \Pi\mu_i, \Pi\mu_j\rangle| \leq \frac{40\sqrt{\epsilon}}{\varphi} \cdot \frac{1}{\sqrt{|C_i|\cdot|C_j|}}$.*

Proof. **Proof of item** (1)**:** Since Π is a orthogonal projection matrix we have $||\Pi||_2 = 1$. Hence, we have $||\Pi\mu_i||_2^2 \leq ||\mu_i||_2^2 \leq \left(1 + \frac{16\sqrt{\epsilon}}{\varphi}\right) ||\mu_i||_2^2$. Thus it's left to prove $||\Pi\mu_i||_2^2 \geq \left(1 - \frac{16\sqrt{\epsilon}}{\varphi}\right) ||\mu_i||_2^2$. Note that by Pythagoras' theorem $||\Pi\mu_i||_2^2 = ||\mu_i||_2^2 - ||(I-\Pi)\mu_i||_2^2$. We will prove $||(I-\Pi)\mu_i||_2^2 \leq \frac{16\sqrt{\epsilon}}{\varphi} ||\mu_i||_2^2$ which implies

$$||\Pi\mu_i||_2^2 \geq \left(1 - 16\frac{\sqrt{\epsilon}}{\varphi}\right) ||\mu_i||_2^2.$$

Let $S' = \{\mu_1,\ldots,\mu_k\} \setminus \{\mu_i\}$. Let Π' denote the orthogonal projection matrix onto $span(S')^\perp$. Note that $S \subseteq S'$, hence $span(S)$ is a subspace of $span(S')$, therefore we have $||(I-\Pi)\mu_i||_2^2 \leq ||(I-\Pi')\mu_i||_2^2$. Thus it suffices to prove $||(I-\Pi')\mu_i||_2^2 \leq \frac{16\sqrt{\epsilon}}{\varphi} ||\mu_i||_2^2$. Let $H = [\mu_1, \mu_2, \ldots, \mu_{i-1}, \mu_{i+1}, \ldots, \mu_k]$ denote a matrix such that its columns are the vectors in S'. Let $W \in \mathbb{R}^{(k-1)\times(k-1)}$ denote a diagonal matrix such that for all $j < i$ we have $W(j,j) = \sqrt{|C_j|}$ and for all $j \geq i$ we have $W(j,j) = \sqrt{|C_{j+1}|}$. Let $Z = HW$. The orthogonal projection matrix onto the span of S' is defined as $(I - \Pi') = Z(Z^T Z)^{-1} Z^T$, and using Lemma 3.14 we get

$$\begin{aligned} ||(I - \Pi')\mu_i||_2^2 &= \mu_i^T Z(Z^T Z)^{-1} Z^T \mu_i \\ &= \mu_i^T Z((Z^T Z)^{-1} - I) Z^T \mu_i + \mu_i^T Z Z^T \mu_i \end{aligned}$$

By Lemma 3.14 $(Z^T Z)^{-1}$ is spectrally close to I, therefore we have

$$\left|\mu_i^T Z\left((Z^T Z)^{-1} - I\right) Z^T \mu_i\right| \leq \frac{5\sqrt{\epsilon}}{\varphi} ||Z^T \mu_i||_2^2$$

Thus we get

$$||(I - \Pi')\mu_i||_2^2 \leq \left(\frac{5\sqrt{\epsilon}}{\varphi} + 1\right) ||Z^T \mu_i||_2^2 \leq 2||Z^T \mu_i||_2^2$$

By Lemma 3.15 we have

$$||Z^T \mu_i||_2^2 = \mu_i^T Z Z^T \mu_i \leq \frac{8\sqrt{\epsilon}}{\varphi} \cdot ||\mu_i||_2^2$$

Therefore we get

$$||(I - \Pi)\mu_i||_2^2 \leq ||(I - \Pi')\mu_i||_2^2 \leq 2||Z^T \mu_i||_2^2 \leq \frac{16\sqrt{\epsilon}}{\varphi} ||\mu_i||_2^2 \tag{3.42}$$

Hence,

$$||\Pi \mu_i||_2^2 \geq \left(1 - 16\frac{\sqrt{\epsilon}}{\varphi}\right) ||\mu_i||_2^2.$$

Proof of item (2)**:** Note that

$$\langle \mu_i, \mu_j \rangle = \langle (I - \Pi)\mu_i + \Pi\mu_i, (I - \Pi)\mu_j + \Pi\mu_j \rangle = \langle (I - \Pi)\mu_i, (I - \Pi)\mu_j \rangle + \langle \Pi\mu_i, \Pi\mu_j \rangle$$

Thus by triangle inequality we have

$$|\langle \Pi\mu_i, \Pi\mu_j \rangle| \leq |\langle \mu_i, \mu_j \rangle| + |\langle (I - \Pi)\mu_i, (I - \Pi)\mu_j \rangle| \tag{3.43}$$

By Cauchy Schwarz we have

$$\begin{aligned} |\langle (I - \Pi)\mu_i, (I - \Pi)\mu_j \rangle| &\leq ||(I - \Pi)\mu_i||_2 ||(I - \Pi)\mu_i||_2 \\ &\leq \frac{16\sqrt{\epsilon}}{\varphi} \cdot ||\mu_i||_2 ||\mu_j||_2 && \text{By (3.42)} \\ &\leq \frac{32\sqrt{\epsilon}}{\varphi} \cdot \frac{1}{\sqrt{|C_i||C_j|}} && \text{By Lemma 3.7 for small enough } \frac{\epsilon}{\varphi^2} \end{aligned} \tag{3.44}$$

Also by Lemma 3.7 we have

$$|\langle \mu_i, \mu_j \rangle| \leq \frac{8\sqrt{\epsilon}}{\varphi} \cdot \frac{1}{\sqrt{|C_i||C_j|}} \tag{3.45}$$

Therefore by (3.43), (3.44) and (3.45) we get

$$|\langle \Pi \mu_i, \Pi \mu_j \rangle| \le |\langle \mu_i, \mu_j \rangle| + |\langle (I-\Pi)\mu_i, (I-\Pi)\mu_j \rangle| \le \frac{40\sqrt{\epsilon}}{\varphi} \cdot \frac{1}{\sqrt{|C_i||C_j|}}.$$

□

3.4.4 Robustness property of (k,φ,ϵ)-clusterable graphs

In this subsection we show a Lemma that establishes a robustness property of (k,φ,ϵ)-clusterable graphs. That is we show that any collection $\{S_1, S_2, \ldots, S_k\}$ of pairwise disjoint subsets of vertices must match clusters $\{C_1, \ldots, C_k\}$ well.

Lemma 3.16. *Let $G = (V, E)$ be a d-regular graph that admits a (k,φ,ϵ)-clustering $C_1, \ldots, C_k$. Let $k \ge 2$, $\varphi \in (0,1)$ and $\frac{\epsilon}{\varphi^3}$ be smaller than an absolute positive constant. If $S_1, S_2, \ldots, S_k \subseteq V$ are k disjoint sets such that for all $i \in [k]$*

$$\phi(S_i) \le O\left(\frac{\epsilon}{\varphi^2} \cdot \log(k)\right)$$

then there exists a permutation π on k elements so that for all $i \in [k]$:

$$|C_{\pi(i)} \triangle S_i| \le O\left(\frac{\epsilon}{\varphi^3} \cdot \log(k)\right) |C_{\pi(i)}|$$

Proof. Fix $i \in [k]$ and let $J_i = \{j : |S_i \cap C_j| \le |C_j|/2\}$. Then observe that because the inner conductance of every C_i is at least φ we get:

$$\varphi \sum_{j \in J_i} |S_i \cap C_j| \le O\left(\frac{\epsilon}{\varphi^2} \cdot \log(k)\right) |S_i| \tag{3.46}$$

Using (3.46) and the assumption $\frac{\epsilon}{\varphi^3}$ is sufficiently small we get that

$$\sum_{j \in J_i} |S_i \cap C_j| \le O\left(\frac{\epsilon}{\varphi^3} \cdot \log(k)\right) |S_i| < |S_i| \tag{3.47}$$

(3.47) and $\sum_{j \in [k]} |S_i \cap C_j| = |S_i|$ gives us that

$$\text{For all } i \in [k], J_i \ne [k] \tag{3.48}$$

We will show that for each i: $|[k] \setminus J_i| = 1$ and that a function $i \mapsto \pi(i) \in [k] \setminus J_i$ (that is $\pi(i)$ is the only element of $[k] \setminus J_i$) is a permutation and that it satisfies the claim of the Lemma.

Assume that there exist $i_1 \ne i_2 \in [k]$ and $j \in ([k] \setminus J_{i_1}) \cap ([k] \setminus J_{i_2})$. By definition of J_i's we get that $|S_{i_1} \cap C_j|, |S_{i_2} \cap C_j| > |C_j|/2$ but S_i's are disjoint so it's impossible that two of them intersect more than half of the same C_j. That means that sets $([k] \setminus J_i)$ are pairwise disjoint for all i's. But we also know from (3.48) that for all i $([k] \setminus J_i) \ne \emptyset$. So we have k nonempty, pairwise disjoint subsets of $[k]$, which means that every set contains one element and all elements are different. That in turn means that we can define π as a function $i \mapsto \pi(i) \in [k] \setminus J_i$ and π is a permutation.

Now we show that π satisfies the claim of the Lemma. Observe that because for all $i \in [k]$ the set $[k] \setminus J_i$ contains only one element we get for all $i \in [k]$.

$$\sum_{j \in J_i} |S_i \cap C_j| = |S_i \setminus C_{\pi(i)}| \tag{3.49}$$

Note that because of (3.46) and (3.49) for all $i \in [k]$:

$$|S_i \setminus C_{\pi(i)}| \le O\left(\frac{\epsilon}{\varphi^3} \cdot \log k\right) |S_i|. \tag{3.50}$$

Moreover because inner conductance of every C_i is at least φ and $|C_{\pi(i)} \setminus S_i| < |C_{\pi(i)}|/2$ we get that for all $i \in [k]$

$$\varphi \cdot |C_{\pi(i)} \setminus S_i| \le O\left(\frac{\epsilon}{\varphi^2} \cdot \log(k)\right) |S_i| \tag{3.51}$$

Finally combining (3.50) and (3.51) we get that:

$$|C_{\pi(i)} \triangle S_i| \le O\left(\frac{\epsilon}{\varphi^3} \cdot \log(k)\right) |C_{\pi(i)}|$$

□

3.5 A spectral dot product oracle

Our goal in this section is to develop what we call a *spectral dot product oracle.* The oracle is a sublinear time and space data structure that has oracle access to a (k,φ,ϵ)-clusterable graph G and after a preprocessing step can answer dot products queries for the spectral embedding. Specifically, if $L = U\Lambda U^T$ is the normalized Laplacian of G and the x-th column of $F = U_{[k]}^T$ is called f_x for $x \in V$ then our oracle gets as input two vertices x, y and returns an approximation of $\langle f_x, f_y\rangle$. Both the preprocessing time and the time to evaluate an oracle query are $k^{O(1)} \cdot n^{1/2+O(\epsilon/\varphi^2)} \cdot (\log n)^{O(1)}$, that is, sublinear in n for $\epsilon \ll \varphi^2$. We now state the main theorem that we prove in this section. The algorithms mentioned in Theorem 3.2 can be found later in this section.

Theorem 3.2. *[Spectral Dot Product Oracle] Let $\epsilon, \varphi \in (0,1)$ with $\epsilon \le \frac{\varphi^2}{10^5}$. Let $G = (V,E)$ be a d-regular graph that admits a (k,φ,ϵ)-clustering $C_1,\ldots,C_k$. Let $\frac{1}{n^5} < \xi < 1$. Then* InitializeOracle$(G, 1/2, \xi)$ *(Algorithm 4) computes in time $(\frac{k}{\xi})^{O(1)} \cdot n^{1/2+O(\epsilon/\varphi^2)} \cdot (\log n)^3 \cdot \frac{1}{\varphi^2}$ a sublinear space data structure $\mathcal{D}$ of size $(\frac{k}{\xi})^{O(1)} \cdot n^{1/2+O(\epsilon/\varphi^2)} \cdot (\log n)^3$ such that with probability at least $1-n^{-100}$ the following property is satisfied:*

For every pair of vertices $x, y \in V$, SpectralDotProduct$(G, x, y, 1/2, \xi, \mathcal{D})$ *(Algorithm 5) computes an output value $\langle f_x, f_y\rangle_{apx}$ such that with probability at least $1-n^{-100}$*

$$\left| \langle f_x, f_y\rangle_{apx} - \langle f_x, f_y\rangle \right| \le \frac{\xi}{n}.$$

The running time of SpectralDotProduct$(G, x, y, 1/2, \xi, \mathcal{D})$ *is $(\frac{k}{\xi})^{O(1)} \cdot n^{1/2+O(\epsilon/\varphi^2)} \cdot (\log n)^2 \cdot \frac{1}{\varphi^2}$.*

Furthermore, for any $0 \le \delta \le 1/2$, one can obtain the following trade-offs between preprocessing time and query time: Algorithm SpectralDotProduct$(G, x, y, \delta, \xi, \mathcal{D})$ *requires $(\frac{k}{\xi})^{O(1)} \cdot n^{\delta+O(\epsilon/\varphi^2)} \cdot (\log n)^2 \cdot \frac{1}{\varphi^2}$ per query when the prepressing time of Algorithm* InitializeOracle(G, δ, ξ) *is increased to $(\frac{k}{\xi})^{O(1)} \cdot n^{1-\delta+O(\epsilon/\varphi^2)} \cdot (\log n)^3 \cdot \frac{1}{\varphi^2}$.*

3.5.1 The spectral dot product oracle - overview

In the following sections we provide the proof of the spectral dot product oracle. Recall from the technical overview that we are using the following algorithms (we restate them for convenience of the reader). Our main tool for accessing the spectral embedding of the graph is a primitive that runs a few short (logarithmic length) random walks from a given vertex.

Algorithm 1 RunRandomWalks(G, R, t, x)

```
Run $R$ random walks of length $t$ starting from $x$
Let $\widehat{m}_x(y)$ be the fraction of random walks that ends at $y$    ▷ vector $\widehat{m}_x$ has support at most $R$
return $\widehat{m}_x$
```

Another key primitive uses collision statistics to estimate the Gram matrix of random walk distributions started at vertices in a set S.

Algorithm 2 EstimateCollisionProbabilities(G, I_S, R, t)

```
for $i = 1$ to $O(\log n)$ do
    $\widehat{Q}_i :=$ EstimateTransitionMatrix$(G, I_S, R, t)$
    $\widehat{P}_i :=$ EstimateTransitionMatrix$(G, I_S, R, t)$
    $\mathcal{G}_i := \frac{1}{2}\left(\widehat{P}_i^T \widehat{Q}_i + \widehat{Q}_i^T \widehat{P}_i\right)$    ▷ $\mathcal{G}_i$ is symmetric
Let $\mathcal{G}$ be a matrix obtained by taking the entrywise median of $\mathcal{G}_i$'s    ▷ $\mathcal{G}$ is symmetric
return $\mathcal{G}$
```

We also need the following procedure.

Algorithm 3 EstimateTransitionMatrix(G, I_S, R, t)

```
for each sample $x \in I_S$ do
    $\widehat{m}_x :=$ RunRandomWalks$(G, R, t, x)$
Let $\widehat{Q}$ be the matrix whose columns are $\widehat{m}_x$ for $x \in I_S$
return $\widehat{Q}$    ▷ $\widehat{Q}$ has at most $Rs$ non-zeros
```

Then we can initialize the dot product oracle.

Algorithm 4 InitializeOracle(G, δ, ξ) ▷ Need: $\epsilon/\varphi^2 \le \frac{1}{10^5}$

```
$t := \frac{20 \cdot \log n}{\varphi^2}$
$R_{\text{init}} := O(n^{1-\delta+980\cdot\epsilon/\varphi^2} \cdot k^{17}/\xi^2)$
$s := O(n^{480\cdot\epsilon/\varphi^2} \cdot \log n \cdot k^8/\xi^2)$
Let $I_S$ be the multiset of $s$ indices chosen independently and uniformly at random from $\{1,\dots,n\}$
for $i = 1$ to $O(\log n)$ do
    $\widehat{Q}_i :=$ EstimateTransitionMatrix($G, I_S, R_{\text{init}}, t$)    ▷ $\widehat{Q}_i$ has at most $R_{\text{init}} \cdot s$ non-zeros
$\mathcal{G} :=$EstimateCollisionProbabilities($G, I_S, R_{\text{init}}, t$)
Let $\frac{n}{s} \cdot \mathcal{G} := \widehat{W}\widehat{\Sigma}\widehat{W}^T$ be the eigendecomposition of $\frac{n}{s} \cdot \mathcal{G}$    ▷ $\mathcal{G} \in \mathbb{R}^{s\times s}$
if $\widehat{\Sigma}^{-1}$ exists then
    $\Psi := \frac{n}{s} \cdot \widehat{W}_{[k]}\widehat{\Sigma}_{[k]}^{-2}\widehat{W}_{[k]}^T$    ▷ $\Psi \in \mathbb{R}^{s\times s}$
    return $\mathcal{D} := \{\Psi, \widehat{Q}_1, \dots, \widehat{Q}_{O(\log n)}\}$
```

Finally, we have the query algorithm.

Algorithm 5 SpectralDotProductOracle($G, x, y, \delta, \xi, \mathcal{D}$) ▷ Need: $\epsilon/\varphi^2 \le \frac{1}{10^5}$
▷ $\mathcal{D} := \{\Psi, \widehat{Q}_1, \dots, \widehat{Q}_{O(\log n)}\}$

```
$R_{\text{query}} := O(n^{\delta+500\cdot\epsilon/\varphi^2} \cdot k^9/\xi^2)$
for $i = 1$ to $O(\log n)$ do
    $\widehat{m}_x^i :=$ RunRandomWalks($G, R_{\text{query}}, t, x$)
    $\widehat{m}_y^i :=$ RunRandomWalks($G, R_{\text{query}}, t, y$)
Let $\alpha_x$ be a vector obtained by taking the entrywise median of $(\widehat{Q}_i)^T(\widehat{m}_x^i)$ over all runs
Let $\alpha_y$ be a vector obtained by taking the entrywise median of $(\widehat{Q}_i)^T(\widehat{m}_y^i)$ over all runs
return $\langle f_x, f_y \rangle_{apx} := \alpha_x^T \Psi \alpha_y$
```

Let $I_S = \{i_1, \dots, i_s\}$ be a multiset of s indices chosen independently and uniformly at random from $\{1,\dots,n\}$. Let S be the $n \times s$ matrix whose j-th column equals $\mathbb{1}_{i_j}$. As already explained in detail in the technical overview, we first prove stability bounds for the pseudoinverse. Then we show that that M^t is approximated by $M^t S$ and finally we show that algorithm RunRandomWalks approximates the $M^t \mathbb{1}_x$ sufficiently well. We conclude with the proof of Theorem 3.2.

3.5.2 Stability bounds for the low rank approximation

The main result of this section is a bound on the stability of the pseudoinverse of the rank-k approximation of two symmetric, positive semi-definite matrices $A, \widetilde{A} \in \mathbb{R}^{n\times n}$ that are spectrally close and that have an eigenvalue gap between the k-th and $(k+1)$-st eigenvalue. In order to prove this result, we use Weyl's inequality, which gives bounds on the eigenvalues of the sum of a matrix A and a perturbation matrix P. Recall that for a symmetric matrix A, we write $\nu_i(A)$ (resp. $\nu_{\max}(A), \nu_{\min}(A)$) to denote the i^{th} largest (resp. maximum, minimum) eigenvalue of A.

Lemma 3.17 (Weyl's Inequality)**.** *Let $A, P \in \mathbb{R}^{n\times n}$ be two symmetric matrices. Then we have for all $i \in \{1,\dots,n\}$:*

$$\nu_i(A) + \nu_{\min}(P) \le \nu_i(A+P) \le \nu_i(A) + \nu_{\max}(P),$$

where for a symmetric matrix $H \in \mathbb{R}^{n\times n}$ $\nu_i(H)$ denotes its i^{th} largest eigenvalue and $\nu_{\min}(H)$ and $\nu_{\max}(H)$ refer to the smallest and largest eigenvalues of H.

We will use the Davis-Kahan $\sin(\theta)$ Theorem [DK70] (the version given in the note [DK]).

Theorem 3.5 (Davis-Kahan $sin(\theta)$-Theorem [DK70])**.** . *Let $H = E_0 A_0 E_0^T + E_1 A_1 E_1^T$ and $\widetilde{H} = F_0 \Lambda_0 F_0^T + F_1 \Lambda_1 F_1^T$ be symmetric real-valued matrices with E_0, E_1 and F_0, F_1 orthogonal. If the eigenvalues of A_0 are contained in an interval (a, b), and the eigenvalues of Λ_1 are excluded from the interval $(a-\eta, b+\eta)$ for some $\eta > 0$, then for any unitarily invariant norm $\|.\|$*

$$\|F_1^T E_0\| \le \frac{\|F_1^T(\widetilde{H} - H)E_0\|}{\eta}.$$

Let $m \le n$ be integers. For any matrix $A \in \mathbb{R}^{n\times m}$ with singular value decomposition (SVD) $A = Y\Gamma Z^T$ we assume $Y \in \mathbb{R}^{n\times n}$, $\Gamma \in \mathbb{R}^{n\times n}$ is a diagonal matrix of singular values and $Z \in \mathbb{R}^{m\times n}$ (this is a slightly non-standard definition of the SVD, but having Γ be a square matrix will be convenient). Y has orthonormal columns, the first m columns of Z are orthonormal, and the rest of the columns of Z are zero. For any integer $q \in [m]$ we denote $Y_{[q]} \in \mathbb{R}^{n\times q}$ as the first q columns of Y and $Y_{-[q]}$ to denote the matrix of the remaining columns of Y. We also denote by $Z_{[q]} \in \mathbb{R}^{m\times q}$ as the first q columns of Z and $Z_{-[q]}$ to denote the matrix of the remaining $n-q$ columns of Z. Finally we denote by $\Gamma_{[q]} \in \mathbb{R}^{q\times q}$ the submatrix of Γ corresponding to the first q rows and columns of Γ and we use $\Gamma_{-[q]}$ to denote the submatrix corresponding to the last $n-q$ rows and $n-q$ columns of Γ. So for any $q \in [m]$ the span of $Y_{-[q]}$ is the orthogonal complement of the span of $Y_{[q]}$ in $\mathbb{R}^n$, also the span of the columns of $Z_{-[q]}$ is the orthogonal complement of the span of $Z_{[q]}$ in $\mathbb{R}^m$. Thus we can write $A = Y_{[q]}\Gamma_{[q]}Z_{[q]}^T + Y_{-[q]}\Gamma_{-[q]}Z_{-[q]}^T$.

Claim 3.1. *For every symmetric matrix E and every pair of orthogonal projection matrices $P, \widetilde{P}$ one has*

$$||P\cdot E\cdot P - \widetilde{P}\cdot E\cdot \widetilde{P}||_2 \le 2\|E\|_2 \cdot (\|P\cdot(I-\widetilde{P})\|_2 + \|\widetilde{P}\cdot(I-P)\|_2).$$

Proof. Since $\widetilde{P} + (I-\widetilde{P}) = I$ we can write

$$\begin{aligned} P\cdot E\cdot P &= (\widetilde{P} + (I-\widetilde{P}))P\cdot E\cdot P\cdot(\widetilde{P} + (I-\widetilde{P})) \\ &= P\cdot E\cdot P\cdot(I-\widetilde{P}) + \widetilde{P}\cdot P\cdot E\cdot P\cdot\widetilde{P} + (I-\widetilde{P})\cdot P\cdot E\cdot P\cdot\widetilde{P} \qquad (3.52) \end{aligned}$$

Since $P+(I-P)=I$ we have

$$\begin{aligned} \widetilde{P}\cdot E\cdot\widetilde{P} &= \widetilde{P}(P+(I-P))\cdot E\cdot(P+(I-P))\,\widetilde{P}||_2 \\ &= \widetilde{P}\cdot E\cdot(I-P)\widetilde{P}+\widetilde{P}\cdot P\cdot E\cdot P\cdot\widetilde{P}+\widetilde{P}\cdot(I-P)\cdot E\cdot P\cdot\widetilde{P} \end{aligned} \tag{3.53}$$

Putting (3.52) and (3.53) together and by triangle inequality we get

$$\begin{aligned} &||P\cdot E\cdot P-\widetilde{P}\cdot E\cdot\widetilde{P}||_2 \\ &\leq \|P\cdot E\cdot P\cdot(I-\widetilde{P})\|_2+\|(I-\widetilde{P})\cdot P\cdot E\cdot P\cdot\widetilde{P}\|_2+\|\widetilde{P}\cdot E\cdot(I-P)\widetilde{P}\|_2+\|\widetilde{P}\cdot(I-P)\cdot E\cdot P\cdot\widetilde{P}\|_2 \end{aligned}$$

Thus by submultiplicativity of the operator norm we get

$$\begin{aligned} &||P\cdot E\cdot P-\widetilde{P}\cdot E\cdot\widetilde{P}||_2 \\ &\leq \|P\|_2\|E\|_2\|P\cdot(I-\widetilde{P})\|_2+\|(I-\widetilde{P})\cdot P\|_2\|E\|_2\|P\|_2\|\widetilde{P}\|_2+\|\widetilde{P}\|_2\|E\|_2\|(I-P)\widetilde{P}\|_2+\|\widetilde{P}\|_2\|E\|_2\|(I-P)\widetilde{P}\|_2 \\ &\leq \|E\|_2\left(\|P\cdot(I-\widetilde{P})\|_2+\|(I-\widetilde{P})\cdot P\|_2+\|(I-P)\widetilde{P}\|_2+\|\widetilde{P}(I-P)\|_2\right) \text{ Since } ||P||=||\widetilde{P}||_2=1 \\ &= 2\cdot\|E\|_2\cdot(\|P\cdot(I-\widetilde{P})\|_2+\|\widetilde{P}\cdot(I-P)\|_2), \end{aligned}$$

where the last equality holds since $\|P\cdot(I-\widetilde{P})\|_2=\|(I-\widetilde{P})^T\cdot P^T\|_2=\|(I-\widetilde{P})\cdot P\|_2$ and similarly since $\|\widetilde{P}\cdot(I-P)\|_2=\|(I-P)^T\cdot\widetilde{P}^T\|_2=\|(I-P)\cdot\widetilde{P}\|_2$. □

Recall that for matrices $A,\widetilde{A}\in\mathbb{R}^{n\times n}$, we write $A\preccurlyeq\widetilde{A}$, if $\forall x\in\mathbb{R}^n$ we have $x^TAx\leq x^T\widetilde{A}x$ and we write $A\prec\widetilde{A}$, if $\forall x\in\mathbb{R}^n$ we have $x^TAx<x^T\widetilde{A}x$. Now we can state the main technical result of this section (Lemma 3.18), whose proof relies on matrix perturbation bounds Davis-Kahan $\sin\theta$ theorem (Theorem 3.5).

We will need the following claim, whose proof is inspired by the proof of the operator monotonicity of negative matrix inverse [Tod11]:

Claim 3.2. *Let $A,B\in\mathbb{R}^{n\times n}$ be symmetric positive semidefinite matrices. Let Π_B denote orthogonal projection operator onto the range space of B. Then if $A\succeq B$, we have for every orthogonal projection Π_A satisfying $\Pi_AA^+=A^+\Pi_A$ that*

$$(\Pi_AA\Pi_A)^+\preceq B^++2\|\Pi_AA^+\|_2\|\Pi_A(I-\Pi_B)\|_2\cdot I.$$

Proof. For every $x\in\mathbb{R}^n$, and every $y\in\mathbb{R}^n$ (to be chosen as $y=A^+x$ later) since B is positive semidefinite we have

$$(y-B^+x)^TB(y-B^+x)\geq 0,$$

which in particular implies that

$$y^TBy-2x^TB^+By+x^TB^+x\geq 0,$$

and since $A \succeq B$ by assumption,

$$y^T Ay - 2x^T B^+ By + x^T B^+ x \geq 0.$$

We now chose $y = \Pi_A A^+ x$ and rearrange, getting

$$2x^T B^+ B\Pi_A A^+ x - x^T \Pi_A A^+ \Pi_A x \leq x^T B^+ x. \tag{3.54}$$

Noting that $B^+ B = \Pi_B$ and $\Pi_A \Pi_A A^+ = \Pi_A A^+ \Pi_A$, we write the lhs of (3.54) as

$$\begin{aligned}2x^T \Pi_B \Pi_A A^+ x - x^T \Pi_A A^+ \Pi_A x &= 2x^T \Pi_A A^+ \Pi_A x + 2x^T (\Pi_B \Pi_A - \Pi_A)\Pi_A A^+ x - x^T \Pi_A A^+ \Pi_A x \\ &= x^T \Pi_A A^+ \Pi_A x + 2x^T ((\Pi_B - I)\Pi_A)\Pi_A A^+ x.\end{aligned}$$

Substituting the above into (3.54), and noting that

$$|x^T (\Pi_B \Pi_A - \Pi_A)\Pi_A A^+ x| \leq \|\Pi_A A^+\|_2 \cdot \|(\Pi_B - I)\Pi_A\|_2 \cdot x^T x,$$

we get

$$x^T \Pi_A A^+ \Pi_A x \leq x^T B^+ x + 2\|\Pi_A A^+\|_2 \cdot \|(\Pi_B - I)\Pi_A\|_2 \cdot x^T x.$$

The above holds for all $x \in \mathbb{R}^n$. Also, $\|(\Pi_B - I)\Pi_A\|_2 = \|\Pi_A(I - \Pi_B)\|_2$, since Π_A, Π_B are projection matrices. Therefore, for all $x \in \mathbb{R}^n$ we have

$$\Pi_A A^+ \Pi_A \preceq B^+ + 2\|\Pi_A A^+\|_2 \cdot \|\Pi_A(I - \Pi_B)\|_2 \cdot I,$$

as required. □

Lemma 3.18. *Let $A, \widetilde{A} \in \mathbb{R}^{n\times n}$ be symmetric matrices with eigendecompositions $A = Y\Gamma Y^T$ and $\widetilde{A} = \widetilde{Y}\widetilde{\Gamma}\widetilde{Y}^T$. Let the eigenvalues of A be $1 \geq \gamma_1 \geq \cdots \geq \gamma_n \geq 0$. Suppose that $\|A - \widetilde{A}\|_2 \leq \frac{\gamma_k}{100}$ and $\gamma_{k+1} < \gamma_k/4$. Then we have*

$$\|Y_{[k]}\Gamma_{[k]}^{-1} Y_{[k]}^T - \widetilde{Y}_{[k]}\widetilde{\Gamma}_{[k]}^{-1}\widetilde{Y}_{[k]}^T\|_2 \leq \frac{16\|A - \widetilde{A}\|_2 + 4\gamma_{k+1}}{\gamma_k^2}.$$

Proof. We define $P = Y_{[k]}Y_{[k]}^T$ and $\widetilde{P} = \widetilde{Y}_{[k]}\widetilde{Y}_{[k]}^T$, and let $M = PAP = Y_{[k]}\Gamma_{[k]}Y_{[k]}^T$ and $\widetilde{M} = \widetilde{P}\widetilde{A}\widetilde{P} =$

$\widetilde{Y}_{[k]}\widetilde{\Gamma}_{[k]}\widetilde{Y}_{[k]}^T$. First note that

$$\begin{aligned}
\widetilde{M} &= \widetilde{P}\widetilde{A}\widetilde{P} \\
&\preceq \widetilde{P}(\widetilde{A} + \|A - \widetilde{A}\|_2 \cdot I)\widetilde{P} \\
&\preceq \widetilde{P}(\widetilde{A} + \|A - \widetilde{A}\|_2 \cdot I)\widetilde{P} + (I - \widetilde{P})(\widetilde{A} + \|A - \widetilde{A}\|_2 \cdot I)(I - \widetilde{P}) \\
&= \widetilde{A} + \|A - \widetilde{A}\|_2 \cdot I \\
&\preceq A + 2\|A - \widetilde{A}\|_2 \cdot I \\
&= P(A + 2\|A - \widetilde{A}\|_2 \cdot I)P + (I - P)(A + 2\|A - \widetilde{A}\|_2 \cdot I)(I - P) \\
&\preceq M + (2\|A - \widetilde{A}\|_2 + \gamma_{k+1})I \\
&= M + \eta \cdot I,
\end{aligned} \tag{3.55}$$

where we let $\eta = 2\|A - \widetilde{A}\|_2 + \gamma_{k+1}$. The transition from line 2 to line 3 is due to the fact that $\widetilde{A} + \|A - \widetilde{A}\|_2 \cdot I \succeq A \succeq 0$, and therefore $(I - \widetilde{P})(\widetilde{A} + \|A - \widetilde{A}\|_2 \cdot I)(I - \widetilde{P}) \succeq 0$. The transition from line 4 to line 5 is due to $\widetilde{A} \preceq A + \|A - \widetilde{A}\|_2 \cdot I$. The transition from line 6 to line 7 is due to the fact that $(I - P)A(I - P) \preceq \gamma_{k+1}I$.

Similarly,

$$\begin{aligned}
M &= PAP \\
&\preceq PAP + (I - P)A(I - P) \\
&= A \\
&\preceq \widetilde{A} + \|A - \widetilde{A}\|_2 \cdot I \\
&= \widetilde{P}(\widetilde{A} + \|A - \widetilde{A}\|_2 \cdot I)\widetilde{P} + (I - \widetilde{P})(\widetilde{A} + \|A - \widetilde{A}\|_2 \cdot I)(I - \widetilde{P}) \\
&\preceq \widetilde{M} + (2\|A - \widetilde{A}\|_2 + \gamma_{k+1})I.
\end{aligned} \tag{3.56}$$

The transition from line 1 to line 2 is due to the fact that $A \succeq 0$, and therefore $(I-P)A(I-P)^T \succeq 0$. The transition from line 3 to line 4 is due to $A \preceq \widetilde{A} + \|A - \widetilde{A}\|_2 \cdot I$. The transition from line 5 to line 6 is due to the fact that

$$(I - \widetilde{P})\widetilde{A}(I - \widetilde{P}) \preceq \nu_{k+1}(\widetilde{A}) \cdot I \preceq (\|A - \widetilde{A}\|_2 + \gamma_{k+1})I.$$

We now apply Claim 3.2 with $A = M + (2\|A - \widetilde{A}\|_2 + \gamma_{k+1})I$, $\Pi_A = P$, $B = \widetilde{M}$ and $\Pi_B = \widetilde{P}$. Note that A is symmetric and positive semidefinite. Also, B is symmetric and positive semidefinite because $\nu_{\min}(B) = \nu_k(\widetilde{A}) \geq \nu_k(A) - ||A - \widetilde{A}||_2 \geq \frac{99 \cdot \gamma_k}{100} \geq 0$ by Weyl's inequality and the fact that

$||A-\widetilde{A}||_2 \le \frac{\gamma_k}{100}$. Note that $\Pi_A A^+ = A^+ \Pi_A$, as required, and $A \succeq B$ by (3.55). We get

$$\begin{aligned}
\widetilde{M}^+ &\succeq (P(M+\eta I)P)^+ - 2\|P(M+\eta I)^+P\|_2 \cdot \|P(I-\widetilde{P})\|_2 \cdot I \\
&\succeq Y_{[k]}(\Gamma_{[k]}+\eta I_k)^{-1} Y_{[k]}^T + \frac{2}{\gamma_k} \cdot \|P(I-\widetilde{P})\|_2 \cdot I \quad (\text{since } \|P(M+\eta I)^+P\|_2 \le 1/\gamma_k) \\
&\succeq M^+ - \left(\frac{\eta}{\gamma_k^2} + \frac{2}{\gamma_k} \cdot \|P(I-\widetilde{P})\|_2\right) \cdot I \\
&\succeq M^+ - \left(\frac{\eta}{\gamma_k^2} + \frac{8\|A-\widetilde{A}\|_2}{\gamma_k^2}\right) \cdot I. \qquad (3.57)
\end{aligned}$$

The transition from line 2 to line 3 used the fact that

$$\| Y_{[k]}(\Gamma_{[k]}+\eta I_k)^{-1} Y_{[k]}^T - M^+ \| \le \frac{\eta}{\gamma_k^2}. \qquad (3.58)$$

The transition from line 3 to line 4 used

$$\|P(I-\widetilde{P})\|_2 \le \frac{\|A-\widetilde{A}\|_2}{\gamma_k/4}. \qquad (3.59)$$

We verify both (3.58) and (3.59) below.

Similarly, to upper bound $\widetilde{M}^+$ in terms of M^+ we apply Claim 3.2 with $A = \widetilde{M} + (2\|A-\widetilde{A}\|_2 + \gamma_{k+1})I$, $\Pi_A = \widetilde{P}$, $B = M$ and $\Pi_B = P$. Note that $\Pi_A A = A \Pi_A$, as required, A and B are both symmetric and positive semidefinite, and $A \succeq B$ by (3.56). We get

$$\begin{aligned}
M^+ &\succeq (\widetilde{P}(\widetilde{M}+\eta \cdot I)\widetilde{P})^+ + 2\|\widetilde{P}(\widetilde{M}+\eta \cdot I)^+\|_2 \cdot \|\widetilde{P}(I-P)\|_2 \cdot I \\
&\succeq \widetilde{Y}_{[k]}(\widetilde{\Gamma}_{[k]} + I_k)^{-1} \widetilde{Y}_{[k]}^T + \frac{2}{\gamma_k} \cdot \|\widetilde{P}(I-P)\|_2 \cdot I \\
&\succeq \widetilde{M}^+ - \left(\frac{4\eta}{\gamma_k^2} + \frac{2}{\gamma_k} \cdot \|\widetilde{P}(I-P)\|_2\right) \cdot I \\
&\succeq \widetilde{M}^+ - \left(\frac{4\eta}{\gamma_k^2} + \frac{8\|A-\widetilde{A}\|_2}{\gamma_k^2}\right) \cdot I. \qquad (3.60)
\end{aligned}$$

The transition from line 1 to line 2 uses the fact that by Weil's inequality

$$\|\widetilde{P}(\widetilde{M}+\eta \cdot I)^+\|_2 = \frac{1}{\nu_k(\widetilde{A}+\eta \cdot I)} \le \frac{1}{\nu_k(A) - \|A-\widetilde{A}\|_2 + \eta} = \frac{1}{\nu_k(A) + \|A-\widetilde{A}\|_2 + \gamma_{k+1}} \le \frac{1}{\gamma_k},$$

since $\eta = 2\|A-\widetilde{A}\|_2 + \gamma_{k+1}$. The transition from line 2 to line 3 used the fact that

$$\| \widetilde{Y}_{[k]}(\widetilde{\Gamma}_{[k]}+\eta I_k)^{-1} Y_{[k]}^T - \widetilde{M}^+ \| \le \frac{4\eta}{\gamma_k}. \qquad (3.61)$$

The transition from line 3 to line 4 used

$$\|\widetilde{P}(I-P)\|_2 \le \frac{\|A-\widetilde{A}\|_2}{\gamma_k/4}. \tag{3.62}$$

We verify both (3.61) and (3.62) below.

Putting (3.57) and (3.60) together, we get

$$\|M^+ - \widetilde{M}^+\|_2 \le \frac{4\eta}{\gamma_k^2} + \frac{8\|A-\widetilde{A}\|_2}{\gamma_k^2} \le \frac{16\|A-\widetilde{A}\|_2 + 4\gamma_{k+1}}{\gamma_k^2}$$

as required.

We now verify (3.58), (3.59), (3.61) and (3.62). First, one has

$$\begin{aligned}\|Y_{[k]}(\Gamma_{[k]}^{-1} - (\Gamma_{[k]} + \eta\cdot I_k)^{-1})Y_{[k]}^T\|_2 &\le \max_{\xi\ge\gamma_k}\left(\frac{1}{\xi} - \frac{1}{\xi+\eta}\right)\\ &= \max_{\xi\ge\gamma_k}\frac{\eta}{\xi(\xi+\eta)}\\ &\le \frac{\eta}{\gamma_k^2}\end{aligned}$$

and similarly, since $\nu_k(\widetilde{A}) \ge \nu_k(A) - \|A-\widetilde{A}\|_2$ by Weyl's inequality (Lemma 3.17),

$$\begin{aligned}\|\widetilde{Y}_{[k]}(\widetilde{\Gamma}_{[k]}^{-1} - (\widetilde{\Gamma}_{[k]} + \eta\cdot I_k)^{-1})\widetilde{Y}_{[k]}^T\|_2 &\le \max_{\xi\ge\gamma_k - \|A-\widetilde{A}\|_2}\left(\frac{1}{\xi} - \frac{1}{\xi+\eta}\right)\\ &= \max_{\xi\ge\gamma_k - \|A-\widetilde{A}\|_2}\frac{\eta}{\xi(\xi+\eta)}\\ &\le \frac{4\eta}{\gamma_k^2} && \text{Since } \|A-\widetilde{A}\|_2 \le \gamma_k/2 \text{ by assumption}\end{aligned}$$

This verifies (3.58) and (3.61).

It remains to verify (3.59) and (3.62). In order to bound $\|P\cdot(I-\widetilde{P})\|_2$ and $\|\widetilde{P}\cdot(I-P)\|_2$, we first note that by Weyl's inequality

$$\nu_{k+1}(\widetilde{A}) \le \nu_{k+1}(A) + ||A-\widetilde{A}||_2 \le \gamma_k/4 + \gamma_k/100 < (3/4)\gamma_k$$

and $\nu_k(A) = \gamma_k$ by assumption of the lemma. Hence we can apply Theorem 3.5 by choice of $H = A$, $E_0 = Y_{[k]}$, $E_1 = Y_{-[k]}$, $A_0 = \Gamma_{[k]}$, $A_1 = \Gamma_{-[k]}$, and $\widetilde{H} = \widetilde{A}$, $F_0 = \widetilde{Y}_{[k]}$, $F_1 = \widetilde{Y}_{-[k]}$, $\Lambda_0 = \widetilde{\Gamma}_{[k]}$, $\Lambda_1 = \widetilde{\Gamma}_{-[k]}$. Let $\eta = \frac{\gamma_k}{4}$. Note that the eigenvalues of $A_0 = \Gamma_{[k]}$ are at least γ_k and the eigenvalues of $\Lambda_1 = \widetilde{\Gamma}_{-[k]}$ are at most $(3/4)\gamma_k = \gamma_k - \eta$. Therefore, by Theorem 3.5 we have

$$\|\widetilde{Y}_{-[k]}^T Y_{[k]}\|_2 = \|F_1^T E_0\|_2 \le \frac{\|F_1^T(\widetilde{A}-A)E_0\|_2}{\eta} \le \frac{\|A-\widetilde{A}\|_2}{\gamma_k/4}.$$

Thus we have $\| Y_{[k]}^T \widetilde{Y}_{-[k]} \|_2 \le \frac{\|A-\widetilde{A}\|_2}{\gamma_k/4}$. Similarly, we have

$$\nu_{k+1}(A) \le \gamma_k/4$$

and $\nu_k(\widetilde{A}) \ge \nu_k(A) - \|A - \widetilde{A}\|_2 \ge \gamma_k - \gamma_k/100$. Hence we can apply Theorem 3.5 by choice of $H = A$, $E_0 = Y_{-[k]}$, $E_1 = Y_{[k]}$, $A_0 = \Gamma_{-[k]}$, $A_1 = \Gamma_{[k]}$, and $\widetilde{H} = \widetilde{A}$, $F_0 = \widetilde{Y}_{-[k]}$, $F_1 = \widetilde{Y}_{[k]}$, $\Lambda_0 = \widetilde{\Gamma}_{-[k]}$, $\Lambda_1 = \widetilde{\Gamma}_{[k]}$. Let $\eta = \frac{\gamma_k}{4}$. Note that the eigenvalues of $A_0 = \Gamma_{-[k]}$ are at most γ_{k+1} and the eigenvalues of $\Lambda_1 = \widetilde{\Gamma}_{[k]}$ are at least $\gamma_k - \gamma_k/100 \ge \gamma_k - \eta$. Therefore, by Theorem 3.5 we have

$$\| \widetilde{Y}_{[k]}^T Y_{-[k]} \|_2 = \|F_1^T E_0\| \le \frac{\|F_1^T (\widetilde{A} - A) E_0\|}{\eta} \le \frac{\|A - \widetilde{A}\|}{\gamma_k/4}.$$

Thus, we have $\| \widetilde{Y}_{[k]}^T Y_{-[k]} \|_2 \le \frac{\|A-\widetilde{A}\|_2}{\gamma_k/4}$. Putting these two bounds together, we get

$$\|P(I - \widetilde{P})\|_2 = \| Y_{[k]} Y_{[k]}^T \widetilde{Y}_{-[k]} \widetilde{Y}_{-[k]}^T \|_2 = \| Y_{[k]}^T \widetilde{Y}_{-[k]} \|_2 \le \frac{\|A - \widetilde{A}\|_2}{\gamma_k/4},$$

and similarly

$$\|\widetilde{P}(I - P)\|_2 \le \frac{\|A - \widetilde{A}\|_2}{\gamma_k/4}.$$

□

3.5.3 Stability bounds under sampling of vertices

The main result of this section is Lemma 3.19, in which we give bounds for the stability of the pseudoinverse of the rank-k-approximation when we are sampling columns of the k-step random walk matrix of a (k,φ,ϵ)-clusterable graph.

Lemma 3.19. *Let $k \ge 2$ be an integer, $\varphi \in (0,1)$ and $\epsilon \in (0,1)$. Let $G = (V,E)$ be a d-regular and (k,φ,ϵ)-clusterable graph. Let M be the random walk transition matrix of G. Let $1/n^6 < \xi < 1$, $t \ge \frac{20 \log n}{\varphi^2}$. Let $c > 1$ be a large enough constant and let $s \ge c \cdot n^{(480\cdot\epsilon/\varphi^2)} \cdot \log n \cdot k^8/\xi^2$. Let $I_S = \{i_1, \ldots, i_s\}$ be a multiset of s indices chosen independently and uniformly at random from $\{1, \ldots, n\}$. Let S be the $n \times s$ matrix whose j-th column equals $\mathbb{1}_{i_j}$. Let $M^t = U \Sigma^t U^T$ be an eigendecomposition of M^t. Let $\sqrt{\frac{n}{s}} \cdot M^t S = \widetilde{U} \widetilde{\Sigma} \widetilde{W}^T$ be an SVD of $\sqrt{\frac{n}{s}} \cdot M^t S$ where $\widetilde{U} \in \mathbb{R}^{n \times n}$, $\widetilde{\Sigma} \in \mathbb{R}^{n \times n}$, $\widetilde{W} \in \mathbb{R}^{s \times n}$. If $\frac{\epsilon}{\varphi^2} \le \frac{1}{10^5}$ then with probability at least $1 - n^{-100}$ matrix $\widetilde{\Sigma}_{[k]}^{-4}$ exists and we have*

$$\left| \mathbb{1}_x^T U_{[k]} U_{[k]}^T \mathbb{1}_y - (M^t \mathbb{1}_x)^T (M^t S) \left(\frac{n}{s} \cdot \widetilde{W}_{[k]} \widetilde{\Sigma}_{[k]}^{-4} \widetilde{W}_{[k]}^T \right) (M^t S)^T (M^t \mathbb{1}_y) \right| \le \frac{\xi}{n}.$$

To prove Lemma 3.19 we require the following matrix concentration bound, which is a generalization of Bernstein's inequality to matrices.

Lemma 3.20 (Matrix Bernstein [Tro12])**.** *Consider a finite sequence X_i of independent, random matrices with dimensions $d_1 \times d_2$. Assume that each random matrix satisfies $\mathbb{E}[X_i] = 0$ and*

$\|X_i\|_2 \leq b$ *almost surely. Define* $\sigma^2 = \max\{\|\sum_i \mathbb{E}[X_i X_i^T]\|_2, \|\sum_i \mathbb{E}[X_i^T X_i]\|_2\}$. *Then for all* $t \geq 0$,

$$\mathbb{P}\left[\|\sum_i X_i\|_2 \geq t\right] \leq (d_1 + d_2) \cdot \exp\left(\frac{-t^2/2}{\sigma^2 + bt/3}\right).$$

Equiped with the Matrix Bernstein bound, we can show that under certain spectral conditions we can approximate a matrix AA^T by $(AS)(AS)^T$, i.e. by sampling rows of M. The idea is to write $AA^T = \sum_{i=1}^n (A\mathbb{1}_i)(A\mathbb{1}_i)^T$ as a sum over the outer products of its columns and make the sample size depend on the spectral norm of the summands.

Lemma 3.21. *Let* $A \in \mathbb{R}^{n\times n}$ *be a matrix. Let* $B = \max_{\ell\in\{1,\dots,n\}} \|(A\mathbb{1}_\ell)(A\mathbb{1}_\ell)^T\|_2$. *Let* $1 > \xi > 0$. *Let* $s \geq \frac{40n^2B^2\log n}{\xi^2}$. *Let* $I_S = \{i_1, \dots, i_s\}$ *be a multiset of s indices chosen independently and uniformly at random from* $\{1, \dots, n\}$. *Let S be the* $n \times s$ *matrix whose* j*-th column equals* $\mathbb{1}_{i_j}$. *Then we have*

$$\mathbb{P}\left[\left\|AA^T - \frac{n}{s}(AS)(AS)^T\right\|_2 \geq \xi\right] \leq n^{-100}.$$

Proof. Observe that

$$AA^T = \sum_{\ell\in\{1,\dots,n\}} (A\mathbb{1}_\ell)(A\mathbb{1}_\ell)^T. \tag{3.63}$$

and

$$\frac{n}{s}(AS)(AS)^T = \frac{n}{s} \cdot \sum_{i_j \in I_S} (A\mathbb{1}_{i_j})(A\mathbb{1}_{i_j})^T. \tag{3.64}$$

For every $j = 1, 2, \dots, s$ let $X_j = \frac{n}{s} \cdot (A\mathbb{1}_{i_j})(A\mathbb{1}_{i_j})^T$. Thus we have

$$\mathbb{E}[X_j] = \frac{n}{s} \cdot \mathbb{E}[(A\mathbb{1}_{i_j})(A\mathbb{1}_{i_j})^T] = \frac{n}{s} \cdot \frac{1}{n} \sum_{\ell\in\{1,\dots,n\}} (A\mathbb{1}_\ell)(A\mathbb{1}_\ell)^T = \frac{1}{s} \cdot AA^T \tag{3.65}$$

By equality (3.64) we have $\frac{n}{s}(AS)(AS)^T = \sum_{j=1}^s X_j$. Thus by equality (3.65) we get

$$\left\|\frac{n}{s}(AS)(AS)^T - AA^T\right\|_2 = \left\|\sum_{j=1}^s (X_j - \mathbb{E}[X_j])\right\|_2. \tag{3.66}$$

Let $Z_j = X_j - \mathbb{E}[X_j]$. We then have $\|Z_j\|_2 = \|X_j - \mathbb{E}[X_j]\|_2 \leq \|X_j\|_2 + \|\mathbb{E}[X_j]\|_2$ Now let $B = \max_{\ell\in\{1,\dots,n\}} \|(A\mathbb{1}_\ell)(A\mathbb{1}_\ell)^T\|_2$. Furthermore, by our assumption we have

$$\|X_j\|_2 = \left\|\frac{n}{s} \cdot (A\mathbb{1}_j)(A\mathbb{1}_j)^T\right\|_2 \leq \frac{n}{s} \cdot B \tag{3.67}$$

By subadditivity of the spectral norm and (3.65) we get

$$\|\mathbb{E}[X_j]\|_2 \leq \frac{n}{s} \cdot B \tag{3.68}$$

Putting (3.67) and (3.68) together we get

$$\|Z_j\|_2 = \|X_j - \mathbb{E}[X_j]\|_2 \leq \|X_j\|_2 + \|\mathbb{E}[X_j]\|_2 \leq 2 \cdot \frac{n}{s} \cdot B \tag{3.69}$$

We now bound for the variance. Since Z_j is symmetric, we have $Z_j^T Z_j = Z_j Z_j^T = Z_j^2$.

$$\left\| \sum_{j=1}^{s} \mathbb{E}[Z_j^2] \right\|_2 = s \cdot \|\mathbb{E}[Z_j^2]\|_2 = s \cdot \|\mathbb{E}[X_j^2] - \mathbb{E}[X_j]^2\|_2 \leq s \cdot \|\mathbb{E}[X_j^2]\|_2 + s \cdot \|\mathbb{E}[X_j]^2\|_2$$

By submultiplicativity of the spectral norm we get

$$\|\mathbb{E}[X_j^2]\|_2 = \left\| \frac{1}{n} \cdot \frac{n^2}{s^2} \sum_{\ell \in \{1,\ldots,n\}} ((A\mathbb{1}_\ell)(A\mathbb{1}_\ell)^T)^2 \right\|_2 \leq \frac{n^2}{s^2} \cdot B^2 \tag{3.70}$$

Moreover by submultiplicativity of spectral norm we have $\|\mathbb{E}[X_j]^2\|_2 \leq \|\mathbb{E}[X_j]\|_2^2 \leq \frac{n^2}{s^2} \cdot B^2$. Putting things together we obtain

$$\| \sum_{j=1}^{s} \mathbb{E}[Z_j^2] \|_2 \leq \frac{2n^2 B^2}{s}$$

Now we can apply Lemma 3.20 and we get with $b = 2\frac{n}{s}B$ and $\sigma^2 \leq \frac{2n^2B^2}{s}$ using $s \geq \frac{40n^2B^2\log n}{\xi^2}$

$$\mathbb{P}\left[\| \sum_{j=1}^{s} Z_j \|_2 > \xi \right] \leq 2n \cdot \exp\left(\frac{\frac{-\xi^2}{2}}{\sigma^2 + \frac{b\xi}{3}} \right) \leq n^{-100} \tag{3.71}$$

□

The following lemma upper bounds the collision probability from **every** vertex in a (k, φ, ϵ)-clusterable graph using our ℓ_∞ norm bounds on the bottom k eigenvectors of the Laplacian of such graphs[6]:

Lemma 3.22. *Let $k \geq 2$ be an integer, $\varphi \in (0,1)$ and $\epsilon \in (0,1)$. Let $G = (V, E)$ be a d-regular and that admits a (k, φ, ϵ)-clustering $C_1, \ldots, C_k$. Let M be the random walk transition matrix of G. For any $t \geq \frac{20\log n}{\varphi^2}$ and any $x \in V$ we have*

$$\|M^t \mathbb{1}_x\|_2 \leq O(k \cdot n^{-1/2+(20\epsilon/\varphi^2)}).$$

Proof. Let L be the normalized Laplacian of G. Recall that $(u_1, \ldots, u_n)$ are an orthonormal basis of eigenvectors of L with corresponding eigenvalues $0 = \lambda_1 \leq \ldots \leq \lambda_n$. Observe that each

[6] It is interesting to note that a weaker average case version of this lemma was used in two prior works on testing graph cluster structure [CPS15] and [CKK+18]. The stronger version of the lemma presented here is important for spectral concentration bounds that we present, which are in turn crucial for sublinear time dot product access to the spectral embedding.

u_i is also an eigenvector of M, with eigenvalue $1-\frac{\lambda_i}{2}$. We write $\mathbb{1}_x$ in the eigenbasis of L as $\mathbb{1}_x = \sum_{j=1}^{n} \beta_j u_j$ and note that the β_j correspond to the row of x in the matrix U. We have

$$M^t \mathbb{1}_x = M^t \left(\sum_{j=1}^{n} \beta_j u_j \right) = \sum_{j=1}^{n} \beta_j M^t u_j = \sum_{j=1}^{n} \beta_j \left(1-\frac{\lambda_j}{2}\right)^t u_j.$$

Thus we get

$$\|M^t \mathbb{1}_x\|_2^2 = \sum_{j=1}^{n} \beta_j^2 \left(1-\frac{\lambda_j}{2}\right)^{2t} \leq \sum_{j=1}^{k} \beta_j^2 + \left(1-\frac{\lambda_{k+1}}{2}\right)^{2t} \cdot \sum_{j=k+1}^{n} \beta_j^2. \tag{3.72}$$

Note that G is (k,φ,ϵ)-clusterable, therefore by Lemma 3.3 we have $\lambda_{k+1} \geq \frac{\varphi^2}{2}$. Note that $t \geq \frac{20\log n}{\varphi^2}$. Hence, we have

$$\left(1-\frac{\lambda_{k+1}}{2}\right)^{2t} \leq n^{-10}. \tag{3.73}$$

Moreover since G is (k,φ,ϵ)-clusterable and $\min_i |C_i| \geq \Omega(\frac{n}{k})$ by Lemma 3.5 for all $j \in [k]$ we have

$$\beta_j \leq \|u_j\|_\infty \leq O(\sqrt{k} \cdot n^{-1/2+(20\epsilon/\varphi^2)}). \tag{3.74}$$

Thus by (3.72), (3.73) and (3.74) we get

$$\|M^t \mathbb{1}_x\|_2^2 \leq O(k \cdot k \cdot \frac{1}{n} \cdot n^{40\epsilon/\varphi^2}) + n \cdot n^{-10}.$$

Therefore we have

$$\|M^t \mathbb{1}_x\|_2 \leq O(k \cdot n^{-1/2+(20\epsilon/\varphi^2)}).$$

□

Combining the previous lemmas and Lemma 3.18 we obtain Lemma 3.23. We show that for (k,φ,ϵ)-clusterable graphs, the outer products of the columns of the t-step random walk transition matrix have small spectral norm. This is because the matrix power is mostly determined by the first k eigenvectors and by the fact that these eigenvectors have bounded infinity norm.

Lemma 3.23. *Let $k \geq 2$ be an integer, $\varphi \in (0,1)$ and $\epsilon \in (0,1)$. Let $G=(V,E)$ be a d-regular and (k,φ,ϵ)-clusterable graph. Let M be the random walk transition matrix of G. Let $1 > \xi > 1/n^8$, $t \geq \frac{20\log n}{\varphi^2}$. Let $c > 1$ be a large enough constant and let $s \geq c \cdot k^4 \cdot n^{(400\cdot\epsilon/\varphi^2)} \log n/\xi^2$. Let $I_S = \{i_1,\ldots,i_s\}$ be a multiset of s indices chosen independently and uniformly at random from $\{1,\ldots,n\}$. Let S be the $n \times s$ matrix whose j-th column equals $\mathbb{1}_{i_j}$. Let $M^t = U\Sigma^t U^T$ be an eigendecomposition of M^t. Let $\sqrt{\frac{n}{s}} \cdot M^t S = \widetilde{U}\widetilde{\Sigma}\widetilde{W}^T$ be an SVD of $\sqrt{\frac{n}{s}} \cdot M^t S$ where $\widetilde{U} \in \mathbb{R}^{n\times n}$, $\widetilde{\Sigma} \in \mathbb{R}^{n\times n}$, $\widetilde{W} \in \mathbb{R}^{s\times n}$. If $\frac{\epsilon}{\varphi^2} \leq \frac{1}{10^5}$ then with probability at least $1-n^{-100}$ matrix $\widetilde{\Sigma}_{[k]}^{-2}$ exists and we have*

$$\left\| U_{[k]}\Sigma_{[k]}^{-2t}U_{[k]}^T - \widetilde{U}_{[k]}\widetilde{\Sigma}_{[k]}^{-2}\widetilde{U}_{[k]}^T \right\|_2 < \xi$$

Proof. Let

$$A = (M^t)(M^t)^T = U\Sigma^{2t}U^T,$$

and

$$\widetilde{A} = \frac{n}{s}\left(M^t S\right)\left(M^t S\right)^T = \widetilde{U}\widetilde{\Sigma}^2\widetilde{U}^T.$$

Let γ_k and γ_{k+1} denote the k-th and $(k+1)$-th largest eigenvalues of A. Let U be an orthonormal basis of eigenvectors of L with corresponding eigenvalues $\lambda_1 \le \ldots \le \lambda_n$. Observe that each u_i is also an eigenvector of M, with eigenvalue $1 - \frac{\lambda_i}{2}$. Note that G is (k, φ, ϵ)-clusterable, therefore by Lemma 3.3 we have $\lambda_k \le 2\epsilon$ and $\lambda_{k+1} \ge \frac{\varphi^2}{2}$. Note that $t \ge \frac{20 \log n}{\varphi^2}$. Hence, we have

$$\gamma_{k+1} = \left(1 - \frac{\lambda_{k+1}}{2}\right)^{2t} \le n^{-10} \tag{3.75}$$

and

$$\gamma_k = \left(1 - \frac{\lambda_k}{2}\right)^{2t} \ge n^{(-80\epsilon/\varphi^2)}. \tag{3.76}$$

In order to apply Lemma 3.21 we need to derive an upper bound on the spectral norm of $(M^t\mathbb{1}_x)(M^t\mathbb{1}_x)^T$ for any column of A corresponding to vertex x. By Lemma 3.22 we have

$$B = \|(M^t\mathbb{1}_x)(M^t\mathbb{1}_x)^T\|_2 = \|M^t\mathbb{1}_x\|_2^2 \le O(k^2 \cdot n^{-1+(40\epsilon/\varphi^2)}).$$

Thus, with $1 \ge \xi > 1/n^8$ and for large enough c we have $s \ge c \cdot k^4 n^{(400\cdot\epsilon/\varphi^2)} \log n/\xi^2 \ge \frac{40 n^2 B^2 \log n}{1/32^2 \cdot \xi^2 n^{-320\epsilon/\varphi}}$. Thus by Lemma 3.21 we obtain that with probability at least $1 - n^{-100}$ that

$$\|A - \widetilde{A}\|_2 \le \frac{1}{32} \cdot \xi \cdot n^{-160\epsilon/\varphi^2}. \tag{3.77}$$

We observe that equation 3.77 together with our bound on γ_k (3.76) and the positive semi-definiteness of $\widetilde{A}$ imply that the k largest eigenvalues of $\widetilde{A}$ are non-zero and so $\widetilde{\Sigma}_{[k]}^{-2}$ is exists with high probability.

Now observe that A is positive semi-definite, we have $\gamma_k/4 > \gamma_{k+1}$ and $\|A - \widetilde{A}\| \le \gamma_k/100$, so the preconditions of Lemma 3.18 are met and we have with probability $1 - n^{-100}$

$$\left\|U_{[k]}\Sigma_{[k]}^{-2t}U_{[k]}^T - \widetilde{U}_{[k]}\widetilde{\Sigma}_{[k]}^{-2}\widetilde{U}_{[k]}^T\right\|_2 \le \frac{16\|A - \widetilde{A}\|_2 + 4\gamma_{k+1}}{\gamma_k^2} \le \frac{16 \cdot \frac{1}{32} \cdot \xi \cdot n^{(-160\cdot\epsilon/\varphi^2)} + 4 \cdot n^{-10}}{n^{(-160\cdot\epsilon/\varphi^2)}} \le \frac{\xi}{2} + \frac{\xi}{2} = \xi.$$

□

Now we are ready to prove Lemma 3.19.

Lemma 3.19. *Let $k \ge 2$ be an integer, $\varphi \in (0,1)$ and $\epsilon \in (0,1)$. Let $G = (V, E)$ be a d-regular and (k, φ, ϵ)-clusterable graph. Let M be the random walk transition matrix of G. Let $1/n^6 < \xi < 1$, $t \ge \frac{20 \log n}{\varphi^2}$. Let $c > 1$ be a large enough constant and let $s \ge c \cdot n^{(480\cdot\epsilon/\varphi^2)} \cdot \log n \cdot k^8/\xi^2$. Let $I_S = \{i_1, \ldots, i_s\}$ be a multiset of s indices chosen independently and uniformly at random from*

$\{1,\dots,n\}$. *Let* S *be the* $n\times s$ *matrix whose* j*-th column equals* $\mathbb{1}_{i_j}$. *Let* $M^t = U\Sigma^t U^T$ *be an eigendecomposition of* M^t. *Let* $\sqrt{\frac{n}{s}}\cdot M^t S = \widetilde{U}\widetilde{\Sigma}\widetilde{W}^T$ *be an SVD of* $\sqrt{\frac{n}{s}}\cdot M^t S$ *where* $\widetilde{U}\in\mathbb{R}^{n\times n}, \widetilde{\Sigma}\in\mathbb{R}^{n\times n}, \widetilde{W}\in\mathbb{R}^{s\times n}$. *If* $\frac{\epsilon}{\varphi^2}\le\frac{1}{10^5}$ *then with probability at least* $1-n^{-100}$ *matrix* $\widetilde{\Sigma}_{[k]}^{-4}$ *exists and we have*

$$\left|\mathbb{1}_x^T U_{[k]}U_{[k]}^T\mathbb{1}_y - (M^t\mathbb{1}_x)^T(M^tS)\left(\frac{n}{s}\cdot\widetilde{W}_{[k]}\widetilde{\Sigma}_{[k]}^{-4}\widetilde{W}_{[k]}^T\right)(M^tS)^T(M^t\mathbb{1}_y)\right|\le\frac{\xi}{n}.$$

Proof. Let $m_x = M^t\mathbb{1}_x$ and $m_y = M^t\mathbb{1}_y$. We first prove $m_x^T(U_{[k]}\Sigma_{[k]}^{-2t}U_{[k]}^T)m_y = \mathbb{1}_x^T U_{[k]}U_{[k]}^T\mathbb{1}_y$ and $m_x^T(M^tS)(\widetilde{W}_{[k]}\widetilde{\Sigma}_{[k]}^{-4}\widetilde{W}_{[k]}^T)(M^tS)^T m_y = m_x^T\widetilde{U}_{[k]}\widetilde{\Sigma}_{[k]}^{-2}\widetilde{U}_{[k]}^T m_y$. Then we upper bound

$$\left|m_x^T U_{[k]}\Sigma_{[k]}^{-2t}U_{[k]}^T m_y - m_x^T\widetilde{U}_{[k]}\widetilde{\Sigma}_{[k]}^{-2}\widetilde{U}_{[k]}^T m_y\right|.$$

Step 1:Note that $M^t = U\Sigma^tU^T$. Therefore we get $M^t\mathbb{1}_x = U\Sigma^tU^T\mathbb{1}_x$, and $M^t\mathbb{1}_y = U\Sigma^tU^T\mathbb{1}_y$. Thus we have

$$m_x^T U_{[k]}\Sigma_{[k]}^{-2t}U_{[k]}^T m_y = \mathbb{1}_x^T\left(\left(U\Sigma^tU^T\right)\left(U_{[k]}\Sigma_{[k]}^{-2t}U_{[k]}^T\right)\left(U\Sigma^tU^T\right)\right)\mathbb{1}_y \tag{3.78}$$

Note that $U^TU_{[k]}$ is an $n\times k$ matrix such that the top $k\times k$ matrix is $I_{k\times k}$ and the rest is zero. Also $U_{[k]}^TU$ is a $k\times n$ matrix such that the left $k\times k$ matrix is $I_{k\times k}$ and the rest is zero. Therefore we have

$$U\Sigma^t\left(U^TU_{[k]}\right)\Sigma_{[k]}^{-2t}\left(U_{[k]}^TU\right)\Sigma^tU^T = UHU^T,$$

where H is an $n\times n$ matrix such that the top left $k\times k$ matrix is $I_{k\times k}$ and the rest is zero. Hence, we have

$$UHU^T = U_{[k]}U_{[k]}^T.$$

Thus we have

$$m_x^T(U_{[k]}\Sigma_{[k]}^{-2t}U_{[k]}^T)m_y = \mathbb{1}_x^TU_{[k]}U_{[k]}^T\mathbb{1}_y \tag{3.79}$$

Step 2:We have $\sqrt{\frac{n}{s}}\cdot M^tS = \widetilde{U}\widetilde{\Sigma}\widetilde{W}^T$ where $\widetilde{U}\in\mathbb{R}^{n\times n}$, $\widetilde{\Sigma}\in\mathbb{R}^{n\times n}$ and $\widetilde{W}\in\mathbb{R}^{s\times n}$. Therefore,

$$\begin{aligned}
&(m_x)^T(M^tS)\left(\frac{n}{s}\cdot\widetilde{W}_{[k]}\widetilde{\Sigma}_{[k]}^{-4}\widetilde{W}_{[k]}^T\right)(M^tS)^T(m_y)\\
&= m_x^T\left(\sqrt{\frac{s}{n}}\cdot\widetilde{U}\widetilde{\Sigma}\widetilde{W}^T\right)\left(\frac{n}{s}\cdot\widetilde{W}_{[k]}\widetilde{\Sigma}_{[k]}^{-4}\widetilde{W}_{[k]}^T\right)\left(\sqrt{\frac{s}{n}}\cdot\widetilde{W}\widetilde{\Sigma}\widetilde{U}^T\right)m_y\\
&= m_x^T\left(\widetilde{U}\widetilde{\Sigma}\widetilde{W}^T\right)\left(\widetilde{W}_{[k]}\widetilde{\Sigma}_{[k]}^{-4}\widetilde{W}_{[k]}^T\right)\left(\widetilde{W}\widetilde{\Sigma}\widetilde{U}^T\right)m_y
\end{aligned} \tag{3.80}$$

Note that $\widetilde{W}^T\widetilde{W}_{[k]}$ is an $n\times k$ matrix such that the top $k\times k$ matrix is $I_{k\times k}$ and the rest is zero. Also $\widetilde{W}_{[k]}^T\widetilde{W}$ is a $k\times n$ matrix such that the left $k\times k$ matrix is $I_{k\times k}$ and the rest is zero. Therefore we have

$$\widetilde{\Sigma}\left(\widetilde{W}^T\widetilde{W}_{[k]}\right)\widetilde{\Sigma}_{[k]}^{-4}\left(\widetilde{W}_{[k]}^T\widetilde{W}\right)\widetilde{\Sigma} = \widetilde{H},$$

where $\widetilde{H}$ is an $n \times n$ matrix such that the top left $k \times k$ matrix is $\widetilde{\Sigma}_{[k]}^{-2}$ and the rest is zero. Hence, we have

$$(\widetilde{U}\widetilde{\Sigma}\widetilde{W}^T)\left(\frac{n}{s}\cdot\widetilde{W}_{[k]}\widetilde{\Sigma}_{[k]}^{-4}\widetilde{W}_{[k]}^T\right)(\widetilde{W}\Sigma\widetilde{U}^T) = \widetilde{U}\widetilde{H}\widetilde{U}^T = \widetilde{U}_{[k]}\widetilde{\Sigma}_{[k]}^{-2}\widetilde{U}_{[k]}^T \tag{3.81}$$

Putting (3.81) and (3.80) together we get

$$m_x^T(M^tS)(\widetilde{W}_{[k]}\widetilde{\Sigma}_{[k]}^{-4}\widetilde{W}_{[k]}^T)(M^tS)^T m_y = m_x^T\widetilde{U}_{[k]}\widetilde{\Sigma}_{[k]}^{-2}\widetilde{U}_{[k]}^T m_y \tag{3.82}$$

Put together: Let $c' > 1$ be a large enough constant we will set later. Let $\xi' = \frac{\xi}{c'\cdot k^2\cdot n^{40\epsilon/\varphi^2}}$. Let c_1 be a constant in front of s in Lemma 3.23. Thus for large enough c we have $s \geq c\cdot n^{(480\cdot\epsilon/\varphi^2)}\cdot\log n\cdot k^8/\xi^2 \geq c_1\cdot k^4\cdot n^{(400\cdot\epsilon/\varphi^2)}\log n/\xi'^2$, hence, by Lemma 3.23 applied with ξ', with probability at least $1 - n^{-100}$ we have

$$\left|\left|U_{[k]}\Sigma_{[k]}^{-2t}U_{[k]}^T - \widetilde{U}_{[k]}\widetilde{\Sigma}_{[k]}^{-2}\widetilde{U}_{[k]}^T\right|\right|_2 \leq \xi'$$

Therefore by submultiplicativity of norm we have

$$\begin{aligned}\left|m_x^T U_{[k]}\Sigma_{[k]}^{-2t}U_{[k]}^T m_y - m_x^T\widetilde{U}_{[k]}\widetilde{\Sigma}_{[k]}^{-2}\widetilde{U}_{[k]}^T m_y\right| &\leq \left|\left|U_{[k]}\Sigma_{[k]}^{-2t}U_{[k]}^T - \widetilde{U}_{[k]}\widetilde{\Sigma}_{[k]}^{-2}\widetilde{U}_{[k]}^T\right|\right|_2 \|m_x\|_2\|m_y\|_2 \\ &\leq \xi'\|m_x\|_2\|m_y\|_2 \end{aligned}\tag{3.83}$$

Therefore we have

$$\begin{aligned}&\left|m_x^T(M^tS)\left(\frac{n}{s}\cdot\widetilde{W}_{[k]}\widetilde{\Sigma}_{[k]}^{-4}\widetilde{W}_{[k]}^T\right)(M^tS)^T m_y - \mathbb{1}_x^T U_{[k]}U_{[k]}^T\mathbb{1}_y\right| \\ &= \left|m_x^T\widetilde{U}_{[k]}\widetilde{\Sigma}_{[k]}^{-2}\widetilde{U}_{[k]}^T m_y - m_x^T U_{[k]}\Sigma_{[k]}^{-2t}U_{[k]}^T m_y\right| && \text{By (3.79) and (3.82)} \\ &\leq \xi'\cdot\|m_x\|_2\|m_y\|_2 && \text{By (3.83)}\end{aligned}\tag{3.84}$$

By Lemma 3.22 for any vertex $x \in V$ we have

$$\|m_x\|_2^2 = \|M^t\mathbb{1}_x\|_2^2 \leq O(k^2\cdot n^{-1+(40\epsilon/\varphi^2)}). \tag{3.85}$$

Therefore by choice of c' as a large enough constant and choosing $\xi' = \frac{\xi}{c'\cdot k^2\cdot n^{40\epsilon/\varphi^2}}$ we have

$$\left|m_x^T(M^tS)\left(\frac{n}{s}\cdot\widetilde{W}_{[k]}\widetilde{\Sigma}_{[k]}^{-4}\widetilde{W}_{[k]}^T\right)(M^tS)^T m_y - \mathbb{1}_x^T U_{[k]}U_{[k]}^T\mathbb{1}_y\right| \leq O\left(\xi'\cdot k^2\cdot n^{-1+(40\epsilon/\varphi^2)}\right) \leq \frac{\xi}{n}. \tag{3.86}$$

□

3.5.4 Stability bounds under approximations of columns by random walks

The main result of this section is Lemma 3.24, which shows that if a graph is (k,φ,ϵ)-clusterable, then the pseudoinverseve of the low rank approximation of a random walk matrix are stable when it is empirically approximated by running random walks from sample vertices.

Lemma 3.24. *Let $k \geq 2$ be an integer, $\varphi \in (0,1)$ and $\epsilon \in (0,1)$. Let $G = (V,E)$ be a d-regular and (k,φ,ϵ)-clusterable graph. Let $1/n^8 < \xi < 1$ and $t \geq \frac{20\log n}{\varphi^2}$. Let $c_1 > 1$ and $c_2 > 1$ be a large enough constants. Let $s \geq c_1 \cdot n^{240\epsilon/\varphi^2} \cdot \log n \cdot k^4$ and $R \geq \frac{c_2 \cdot k^9 \cdot n^{(1/2+820\cdot\epsilon/\varphi^2)}}{\xi^2}$. Let $I_S = \{i_1,\ldots,i_s\}$ be a multiset of s indices chosen independently and uniformly at random from $\{1,\ldots,n\}$. Let S be the $n \times s$ matrix whose j-th column equals $\mathbb{1}_{i_j}$. Let $\mathcal{G} \in \mathbb{R}^{s\times s}$ be the output of* EstimateCollisionProbabilities(G, I_S, R, t) *(Algorithm 2). Let M be the random walk transition matrix of G. Let $\sqrt{\frac{n}{s}} \cdot M^t S = \widetilde{U}\widetilde{\Sigma}\widetilde{W}^T$ be an SVD of $\sqrt{\frac{n}{s}} \cdot M^t S$ where $\widetilde{U} \in \mathbb{R}^{n\times n}, \widetilde{\Sigma} \in \mathbb{R}^{n\times n}, \widetilde{W} \in \mathbb{R}^{s\times n}$. Let $\frac{n}{s} \cdot \mathcal{G} = \widehat{W}\widehat{\Sigma}\widehat{W}^T$ be an eigendecomposition of $\frac{n}{s} \cdot \mathcal{G}$. If $\frac{\epsilon}{\varphi^2} \leq \frac{1}{10^5}$ then with probability at least $1 - 2\cdot n^{-100}$ matrices $\widehat{\Sigma}_{[k]}^{-2}$ and $\widetilde{\Sigma}_{[k]}^{-4}$ exist and we have*

$$\left|\left|\widehat{W}_{[k]}\widehat{\Sigma}_{[k]}^{-2}\widehat{W}_{[k]}^T - \widetilde{W}_{[k]}\widetilde{\Sigma}_{[k]}^{-4}\widetilde{W}_{[k]}^T\right|\right|_2 < \xi$$

To prove Lemma 3.24 we need the following lemma.

Lemma 3.25. *Let $k \geq 2$ be an integer, $\varphi \in (0,1)$ and $\epsilon \in (0,1)$. Let $G = (V,E)$ be a d-regular and (k,φ,ϵ)-clusterable graph. Let L and M be the normalized Laplacian and transition matrix of G respectively. For any $t \geq \frac{10\log n}{\varphi^2}$ and any r and any $x \in V$ we have*

$$\|M^t \mathbb{1}_x\|_r \leq O\left(k^2 \cdot n^{-1+1/r+(40\epsilon/\varphi^2)}\right).$$

Proof. Let L be the normalized Laplacian of G with eigenvectors $u_1,\ldots,u_n$ and corresponding eigenvalues $\lambda_1 \leq \ldots \leq \lambda_n$. Observe that each u_i is also an eigenvector of M, with eigenvalue $1 - \frac{\lambda_i}{2}$. Note that G is (k,φ,ϵ)-clusterable. Therefore by Lemma 3.3 we have

$$\lambda_{k+1} \geq \frac{\varphi^2}{2}. \tag{3.87}$$

We write $\mathbb{1}_x$ in the eigenbasis of L as $\mathbb{1}_x = \sum_{j=1}^n \beta_j u_j$ where $\beta_j = u_j \cdot \mathbb{1}_x = u_j(x)$. Thus for any vertex u we have

$$M^t \mathbb{1}_x = M^t \left(\sum_{j=1}^n \beta_j u_j\right) = \sum_{j=1}^n \beta_j M^t u_j = \sum_{j=1}^n \beta_j \left(1 - \frac{\lambda_j}{2}\right)^t u_j.$$

Let $m_x = M^t \mathbb{1}_x$. Therefore for any vertex $y \in V$ we have

$$\begin{aligned} m_x(y) &= \sum_{j=1}^n \beta_j \left(1 - \frac{\lambda_j}{2}\right)^t u_j(y) \\ &= \sum_{j=1}^k \beta_j \left(1 - \frac{\lambda_j}{2}\right)^t u_j(y) + \sum_{j=k+1}^n \beta_j \left(1 - \frac{\lambda_j}{2}\right)^t u_j(y) \end{aligned}$$

Therefore,

$$|m_x(y)| \leq \left(1 - \frac{\lambda_1}{2}\right)^t \sum_{j=1}^{k} |\beta_j| \cdot |u_j(y)| + \left(1 - \frac{\lambda_{k+1}}{2}\right)^t \sum_{j=k+1}^{n} |\beta_j| \cdot |u_j(y)| \tag{3.88}$$

By (3.87) we have $\lambda_{k+1} \geq \frac{\varphi^2}{2}$, and $t \geq \frac{8 \log n}{\varphi^2}$. Thus we have

$$\left(1 - \frac{\lambda_{k+1}}{2}\right)^t \leq n^{-2}$$

Note that for any $j \in [n]$

$$|\beta_j| \leq \sqrt{\sum_{j=1}^{n} \beta_j^2} = \|\mathbb{1}_x\|_2 = 1. \tag{3.89}$$

Morover for any $j \in [n]$ and any $y \in V$

$$|u_j(y)| \leq \|u_j\|_2 = 1 \tag{3.90}$$

Putting (3.89), (3.90) and (3.88) together we get

$$\begin{aligned} |m_x(y)| &\leq \sum_{j=1}^{k} |\beta_j| \cdot |u_j(y)| + \left(1 - \frac{\lambda_{k+1}}{2}\right)^t \sum_{j=k+1}^{n} |\beta_j| \cdot |u_j(y)| \\ &\leq \sum_{j=1}^{k} |\beta_j| \cdot |u_j(y)| + n^{-2} \cdot n \end{aligned} \tag{3.91}$$

Note that G is (k, φ, ϵ)-clusterable and $\min_i |C_i| \geq \Omega(\frac{n}{k})$. Therefore by Lemma 3.5 for all $j \leq k$ we have

$$\beta_j = u_j(x) \leq \|u_j\|_\infty \leq O\left(\sqrt{k} \cdot n^{-1/2 + (20\epsilon/\varphi^2)}\right).$$

Moreover

$$u_j(y) \leq \|u_j\|_\infty \leq O\left(\sqrt{k} \cdot n^{-1/2 + (20\epsilon/\varphi^2)}\right)$$

Thus, we get

$$\sum_{j=1}^{k} |\beta_j| \cdot |u_j(y)| \leq O\left(k \cdot k \cdot n^{-1 + (40\epsilon/\varphi^2)}\right). \tag{3.92}$$

Therefore by (3.91) and (3.92) we get

$$\begin{aligned} |m_x(y)| &\leq O\left(k^2 \cdot n^{-1 + (40\epsilon/\varphi^2)}\right) + n^{-1} \\ &\leq O\left(k^2 \cdot n^{-1 + (40\epsilon/\varphi^2)}\right). \end{aligned} \tag{3.93}$$

Therefore we have

$$\|m_x\|_r \leq \left(n \cdot O\left(k^2 \cdot n^{-1 + (40\epsilon/\varphi^2)}\right)^r\right)^{1/r} = O\left(k^2 \cdot n^{-1 + 1/r + (40\epsilon/\varphi^2)}\right).$$

□

Lemma 3.26. *Let $k \geq 2$ be an integer, $\varphi \in (0,1)$ and $\epsilon \in (0,1)$. Let $G = (V,E)$ be a d-regular and (k,φ,ϵ)-clusterable graph. Let M be the random walk transition matrix of G. Let $\sigma_{err} > 0$. Let t, R_1 and R_2 be integers. Let $a,b \in V$. Suppose that we run R_1 random walks of length t from vertex a and R_2 random walks of length t from vertex b. For any $x \in V$, let $\widehat{m}_a(x)$ (resp. $\widehat{m}_b(x)$) be a random variable which denotes the fraction out of the R_1 (resp. R_2) random walks starting from a (resp. b), which end in x. Let $c > 1$ be a large enough constant. If*

$$\min(R_1,R_2) \geq \frac{c\cdot k^5 \cdot n^{-2+(100\epsilon/\varphi^2)}}{\sigma_{err}^2}, \textit{ and } R_1R_2 \geq \frac{c\cdot k^2\cdot n^{-1+(40\epsilon/\varphi^2)}}{\sigma_{err}^2}$$

then with probability at least 0.99 we have

$$|\widehat{m}_a^T\widehat{m}_b - (M^t\mathbb{1}_a)^T(M^t\mathbb{1}_b)| \leq \sigma_{err}.$$

Remark 3.5. *The success probability of Lemma 3.26 can be boosted up to $1-n^{-100}$ using standard techniques (taking the median of $O(\log n)$ independent runs).*

Proof. Let $m_a = M^t\mathbb{1}_a$ and $m_b = M^t\mathbb{1}_b$. Let $X_{a,r}^i$ be a random variable which is 1 if the r^{th} random walk starting from a, ends at vertex i, and 0 otherwise. Let $Y_{b,r}^i$ be a random variable which is 1 if the r^{th} random walk starting from b, ends at vertex i, and 0 otherwise. Thus, $\mathbb{E}[X_{a,r}^i] = m_a(i)$ and $\mathbb{E}[Y_{b,r}^i] = m_b(i)$. For any two vertices $a,b \in S$, let $Z_{a,b} = \widehat{m}_a^T\widehat{m}_b$ be a random variable given by

$$Z_{a,b} = \frac{1}{R_1R_2}\sum_{i\in V}(\sum_{r_1=1}^{R_1} X_{a,r_1}^i)(\sum_{r_2=1}^{R_2} Y_{b,r_2}^i).$$

Thus,

$$\begin{aligned}\mathbb{E}[Z_{a,b}] &= \frac{1}{R_1R_2}\sum_{i\in V}(\sum_{r_1=1}^{R_1}\mathbb{E}[X_{a,r_1}^i])(\sum_{r_2=1}^{R_2}\mathbb{E}[Y_{b,r_2}^i])\\ &= \frac{1}{R_1R_2}\sum_{i\in V}(R_1\cdot m_a(i))(R_2\cdot m_b(i))\\ &= \sum_{i\in V} m_a(i)\cdot m_b(i) = (m_a)^T(m_b). \end{aligned} \tag{3.94}$$

We know that $\mathrm{Var}(Z_{a,b}) = \mathbb{E}[Z_{a,b}^2] - \mathbb{E}[Z_{a,b}]^2$. Let us first compute $\mathbb{E}[Z_{a,b}^2]$.

$$\begin{aligned}\mathbb{E}[Z_{a,b}^2] &= \mathbb{E}\left[\frac{1}{(R_1R_2)^2}\sum_{i\in V}\sum_{j\in V}\sum_{r_1=1}^{R_1}\sum_{r_2=1}^{R_2}\sum_{r_1'=1}^{R_1}\sum_{r_2'=1}^{R_2} X_{a,r_1}^i Y_{b,r_2}^i X_{a,r_1'}^j Y_{b,r_2'}^j\right]\\ &= \frac{1}{(R_1R_2)^2}\sum_{i\in V}\sum_{j\in V}\sum_{r_1=1}^{R_1}\sum_{r_2=1}^{R_2}\sum_{r_1'=1}^{R_1}\sum_{r_2'=1}^{R_2}\mathbb{E}[X_{a,r_1}^i Y_{b,r_2}^i X_{a,r_1'}^j Y_{b,r_2'}^j]\end{aligned}$$

To compute $\mathbb{E}[X^i_{a,r_1} Y^i_{b,r_2} X^j_{a,r'_1} Y^j_{b,r'_2}]$, we need to consider the following cases.

1. $i \neq j$: $\mathbb{E}[X^i_{a,r_1} Y^i_{b,r_2} X^j_{a,r'_1} Y^j_{b,r'_2}] \leq m_a(i) \cdot m_b(i) \cdot m_a(j) \cdot m_b(j)$. (This is an equality if $r_1 \neq r'_1$ and $r_2 \neq r'_2$. Otherwise, the expectation is zero.)
2. $i = j, \quad r_1 = r'_1, \quad r_2 = r'_2$: $\mathbb{E}[X^i_{a,r_1} Y^i_{b,r_2} X^j_{a,r'_1} Y^j_{b,r'_2}] = m_a(i) \cdot m_b(i)$.
3. $i = j, \quad r_1 = r'_1, \quad r_2 \neq r'_2$: $\mathbb{E}[X^i_{a,r_1} Y^i_{b,r_2} X^j_{a,r'_1} Y^j_{b,r'_2}] = m_a(i) \cdot m_b(i) \cdot \cdot m_b(i)$.
4. $i = j, \quad r_1 \neq r'_1, \quad r_2 = r'_2$: $\mathbb{E}[X^i_{a,r_1} Y^i_{b,r_2} X^j_{a,r'_1} Y^j_{b,r'_2}] = m_a(i) \cdot m_b(i) \cdot m_a(i)$.
5. $i = j, \quad r_1 \neq r'_1, \quad r_2 \neq r'_2$: $\mathbb{E}[X^i_{a,r_1} Y^i_{b,r_2} X^j_{a,r'_1} Y^j_{b,r'_2}] = m_a(i) \cdot m_b(i) \cdot m_a(i) \cdot m_b(i)$.

Thus we have,

$$\begin{aligned}
\mathbb{E}[Z^2_{a,b}] &= \frac{1}{(R_1R_2)^2} \sum_{i\in V} \sum_{j \in V} \sum_{r_1=1}^{R_1} \sum_{r_2=1}^{R_2} \sum_{r'_1=1}^{R_1} \sum_{r'_2=1}^{R_2} \mathbb{E}[X^i_{a,r_1} Y^i_{b,r_2} X^j_{a,r'_1} Y^j_{b,r'_2}] \\
&\leq \sum_{i \in V} \sum_{j \in V\setminus\{i\}} m_a(i) \cdot m_a(j) \cdot m_b(i) \cdot m_b(j) + \sum_{i\in V} m_a(i)^2 \cdot m_b(i)^2 \\
&+ \frac{1}{R_1R_2} \sum_{i \in V} m_a(i) \cdot m_b(i) + \frac{1}{R_1} \sum_{i\in V} m_a(i) \cdot m_b(i)^2 + \frac{1}{R_2} \sum_{i \in V} m_a(i)^2 \cdot m_b(i) \\
&= \sum_{i,j \in V} m_a(i) \cdot m_a(j) \cdot m_b(i) \cdot m_b(j) + \frac{1}{R_1R_2} \sum_{i\in V} m_a(i) \cdot m_b(i) \\
&+ \frac{1}{R_1} \sum_{i \in V} m_a(i) \cdot m_b(i)^2 + \frac{1}{R_2} \sum_{i\in V} m_a(i)^2 \cdot m_b(i).
\end{aligned}$$

Therefore we get,

$$\begin{aligned}
\mathrm{Var}(Z_{a,b}) &= \mathbb{E}[Z^2_{a,b}] - \mathbb{E}[Z_{a,b}]^2 \\
&\leq \sum_{i,j\in V} m_a(i) \cdot m_a(j) \cdot m_b(i) \cdot m_b(j) + \frac{1}{R_1R_2} \sum_{i \in V} m_a(i) \cdot m_b(i) \qquad (3.95) \\
&+ \frac{1}{R_1} \sum_{i\in V} m_a(i) \cdot m_b(i)^2 + \frac{1}{R_2} \sum_{i \in V} m_a(i)^2 \cdot m_b(i) - \left(\sum_{i \in V} m_a(i) \cdot m_b(i) \right)^2 \\
&= \frac{1}{R_1R_2} \sum_{i\in V} m_a(i) \cdot m_b(i) + \frac{1}{R_1} \sum_{i \in V} m_a(i) \cdot m_b(i)^2 + \frac{1}{R_2} \sum_{i\in V} m_a(i)^2 \cdot m_b(i) \\
&\leq \frac{1}{R_1R_2} \|m_a\|_2 \|m_b\|_2 + \frac{1}{R_1} \|m_a\|_2 \|m_b\|_4^2 + \frac{1}{R_2} \|m_a\|_4^2 \|m_b\|_2 \qquad \text{By Cauchy-Schwarz}
\end{aligned}$$

Since $G = (V,E)$ is (k,φ,ϵ) clusterable by Lemma 3.25 we have

$$\|m_a\|_4 \leq O\left(k^2 \cdot n^{-3/4+(40\epsilon/\varphi^2)}\right).$$

and by Lemma 3.22 we have

$$\|m_a\|_2 \le O(k \cdot n^{-1/2+(20\epsilon/\varphi^2)}).$$

Thus we get

$$\operatorname{Var}(Z_{a,b}) \le O\left(\frac{k^2 \cdot n^{-1+(40\epsilon/\varphi^2)}}{R_1 R_2} + \left(\frac{1}{R_1} + \frac{1}{R_2}\right) \cdot k^5 \cdot n^{-2+(100\epsilon/\varphi^2)}\right) \tag{3.96}$$

Then by Chebyshev's inequality, we get,

$$\begin{aligned}\Pr\left[|Z_{a,b} - \mathbb{E}[Z_{a,b}]| > \sigma_{\mathrm{err}}\right] &\le \frac{\operatorname{Var}[Z_{a,b}]}{{\sigma_{\mathrm{err}}}^2} \\ &\le O\left(\frac{1}{{\sigma_{\mathrm{err}}}^2} \cdot \left(\frac{k^2 \cdot n^{-1+(40\epsilon/\varphi^2)}}{R_1 R_2} + \left(\frac{1}{R_1} + \frac{1}{R_2}\right) \cdot k^5 \cdot n^{-2+(100\epsilon/\varphi^2)}\right)\right) \qquad (3.97) \\ &\le \frac{1}{100}.\end{aligned}$$

The last inequality holds by our choice of R_1 and R_2 as follows where c is a large enough constant that cancels the constant hidden in $O(\cdot)$ in (3.97).

$$\min(R_1, R_2) \ge \frac{c \cdot k^5 \cdot n^{-2+(100\epsilon/\varphi^2)}}{\sigma_{\mathrm{err}}^2}$$

and

$$R_1 R_2 \ge \frac{c \cdot k^2 \cdot n^{-1+(40\epsilon/\varphi^2)}}{\sigma_{\mathrm{err}}^2}$$

□

Lemma 3.27. *Let $k \ge 2$ be an integer, $\varphi \in (0,1)$ and $\epsilon \in (0,1)$. Let $G = (V, E)$ be a d-regular and (k, φ, ϵ)-clusterable graph. Let $\sigma_{err} > 0$ and let $s > 0$, $R > 0$, $t > 0$ be integers. Let $I_S = \{i_1, \ldots, i_s\}$ be a multiset of s indices chosen from $\{1, \ldots, n\}$. Let S be the $n \times s$ matrix whose j-th column equals $\mathbb{1}_{i_j}$. Let $c > 1$ be a large enough constant. Let $R \ge \max\left\{\frac{c \cdot k^5 \cdot n^{-2+100\epsilon/\varphi^2}}{\sigma_{err}^2}, \frac{c \cdot k \cdot n^{-1/2+20\epsilon/\varphi^2}}{\sigma_{err}}\right\}$ Let $\mathcal{G} \in \mathbb{R}^{s \times s}$ be the output of Algorithm* EstimateCollisionProbabilities(G, I_S, R, t) *(Algorithm 2). Let M be the random walk transition matrix of G. then with probability at least $1 - n^{-100}$ we have*

$$\|\mathcal{G} - (M^t S)^T (M^t S)\|_2 \le s \cdot \sigma_{err}.$$

Proof. Note that as per line (2) and (3) of Algorithm 2 we first construct matrices $\widehat{P}_i \in \mathbb{R}^{n \times s}$ and $\widehat{Q}_i \in \mathbb{R}^{n \times s}$ using Algorithm 3. as per line (3) of Algorithm 3 matrix $\widehat{P}_i$ (or $\widehat{Q}_i$) has s columns each corresponds to a vertex $x \in S$. The column corresponding to vertex x is $\widehat{m}_x$. as per line 2 of Algorithm 3, $\widehat{m}_x$ is defined as the empirical probability distribution of running R random walks of length t starting from vertex x. Thus for any $x, y \in S$ we have the entry corresponding to the x^{th} row and y^{th} column of $\widehat{Q}_i^T \widehat{P}_i$ (or $\widehat{P}_i^T \widehat{Q}_i$) is $\langle \widehat{m}_x, \widehat{m}_y \rangle$. Since

$$R \geq \max\left\{ \frac{c \cdot k^5 \cdot n^{-2+100\epsilon/\varphi^2}}{\sigma_{\text{err}}^2}, \frac{c \cdot k \cdot n^{-1/2+20\epsilon/\varphi^2}}{\sigma_{\text{err}}} \right\}$$

then by Lemma 3.26 with probability at least 0.99 we have

$$|\widehat{m}_x^T \widehat{m}_y - (M^t \mathbb{1}_x)^T (M^t \mathbb{1}_y)| \leq \sigma_{\text{err}}.$$

Note that as per line 4 of Algorithm 2 we define $\mathcal{G}_i := \frac{1}{2}\left(\widehat{P}_i^T \widehat{Q}_i + \widehat{Q}_i^T \widehat{P}_i\right)$. Thus for any $x, y \in I_S$ we have the entry corresponding to the x^{th} row and y^{th} column of $\mathcal{G}_i$ (i.e., $\mathcal{G}_i(x,y)$) with probability 0.99 satisfies the following:

$$|\mathcal{G}_i(x,y) - (M^t \mathbb{1}_x)^T (M^t \mathbb{1}_y)| \leq \sigma_{\text{err}}.$$

Note that as Line 5 of Algorithm 2 we define $\mathcal{G}$ as a matrix obtained by taking the entrywises median of $\mathcal{G}_i$'s over $O(\log n)$ runs. Thus with probability at least $1 - n^{-100}$ we have for all $x, y \in I_S$

$$|\mathcal{G}(x,y) - (M^t \mathbb{1}_x)^T (M^t \mathbb{1}_y)| \leq \sigma_{\text{err}}.$$

which implies

$$\|\mathcal{G} - (M^t S)^T (M^t S)^T\|_F \leq s \cdot \sigma_{\text{err}}.$$

Since the Frobenius norm of a matrix bounds its maximum eigenvalue from above we get

$$\|\mathcal{G} - (M^t S)^T (M^t S)^T\|_2 \leq s \cdot \sigma_{\text{err}}.$$

□

Recall that for a symmetric matrix A, we write $\nu_i(A)$ (resp. $\nu_{\max}(A), \nu_{\min}(A)$) to denote the i^{th} largest (resp. maximum, minimum) eigenvalue of A.

Lemma 3.28. *Let $k \geq 2$ be an integer, $\varphi \in (0,1)$ and $\epsilon \in (0,1)$. Let $G = (V,E)$ be a d-regular and (k,φ,ϵ)-clusterable graph. Let $t \geq \frac{20 \log n}{\varphi^2}$. Let $c > 1$ be a large enough constant and $s \geq c \cdot n^{240 \cdot \epsilon/\varphi^2} \cdot \log n \cdot k^4$. Let $I_S = \{i_1, \ldots, i_s\}$ be a multiset of s indices chosen independently and uniformly at random from $\{1, \ldots, n\}$. Let S be the $n \times s$ matrix whose j-th column equals $\mathbb{1}_{i_j}$. Let M be the random walk transition matrix of G. If $\frac{\epsilon}{\varphi^2} \leq \frac{1}{10^5}$ then with probability at least $1 - n^{-100}$ we have*

1. $\nu_k\left(\frac{n}{s} \cdot (M^t S)(M^t S)^T\right) \geq \frac{n^{-80\epsilon/\varphi^2}}{2}$
2. $\nu_{k+1}\left(\frac{n}{s} \cdot (M^t S)(M^t S)^T\right) \leq n^{-9}$.

Proof. Let $(u_1, \ldots, u_n)$ be an orthonormal basis of eigenvectors of L with corresponding eigenvalues $0 \le \lambda_1 \le \ldots \le \lambda_n$. Observe that each u_i is also an eigenvector of M, with eigenvalue $1-\frac{\lambda_i}{2}$. Note that G is (k,φ,ϵ)-clusterable, therefore by Lemma 3.3 we have $\lambda_k \le 2\epsilon$ and $\lambda_{k+1} \ge \frac{\varphi^2}{2}$. We have

$$\nu_{k+1}(M^{2t}) = \left(1-\frac{\lambda_{k+1}}{2}\right)^{2t} \le n^{-10}, \text{ and} \tag{3.98}$$

$$\nu_k(M^{2t}) = \left(1-\frac{\lambda_k}{2}\right)^{2t} \ge n^{-80\epsilon/\varphi^2} \tag{3.99}$$

Proof of item (1)**:** Let $A = (M^t)\left(M^t\right)^T$, and $\widetilde{A} = \frac{n}{s}\cdot(M^tS)\left(M^tS\right)^T$. By Lemma 3.22 we have

$$B = \|(M^t\mathbb{1}_x)(M^t\mathbb{1}_x)^T\|_2 \le \|M^t\mathbb{1}_x\|_2^2 \le O\left(k^2\cdot n^{-1+40\epsilon/\varphi^2)}\right).$$

Let $\xi = n^{-80\epsilon/\varphi^2}/2$. Therefore for large enough constant c and by choice of $s = c\cdot k^4 n^{240\epsilon/\varphi^2}\log n$ we have $s \ge \frac{40n^2B^2\log n}{(\xi)^2}$. Thus Lemma 3.21 yields that with probability at least $1-\frac{1}{n^{100}}$ we have

$$\|A-\widetilde{A}\|_2 \le \frac{n^{-80\epsilon/\varphi^2}}{2}. \tag{3.100}$$

Hence, by Weyl's Inequality (see Lemma 3.17) we have

$$\nu_k(\widetilde{A}) \ge \nu_k(A) + \nu_{\min}(\widetilde{A}-A) = \nu_k(A) - \nu_{\max}(A-\widetilde{A}) = \nu_k(A) - \|A-\widetilde{A}\|_2$$

By (3.99) we have $\nu_k(A) = \nu_k(M^{2t}) \ge n^{-10\epsilon/\varphi^2}$ and so

$$\nu_k(\widetilde{A}) \ge \nu_k(A) - \|\widetilde{A}-A\|_2 \ge n^{-80\epsilon/\varphi^2} - \frac{n^{-80\epsilon/\varphi^2}}{2} \ge \frac{n^{-80\epsilon/\varphi^2}}{2}.$$

Proof of item (2)**:** By Lemma 3.8 we have

$$\nu_{k+1}(\widetilde{A}) = \frac{n}{s}\cdot\nu_{k+1}((M^tS)(M^tS)^T) = \frac{n}{s}\cdot\nu_{k+1}((M^tS)^T(M^tS)) = \frac{n}{s}\cdot\nu_{k+1}(S^TM^{2t}S).$$

Recall that $1-\frac{\lambda_1}{2} \ge \cdots \ge 1-\frac{\lambda_n}{2}$ are the eigenvalues of M, and Σ is the diagonal matrix of these eigenvalues in descending order, and U is the matrix whose columns are orthonormal eigenvectors of M arranged in descending order of their eigenvalues. We have $M^{2t} = U\Sigma^{2t}U^T$. Recall that $\Sigma_{[k]}$ is $k\times k$ diagonal matrix with entries $1-\frac{\lambda_1}{2} \ge \cdots \ge 1-\frac{\lambda_k}{2}$, and $\Sigma_{-[k]}$ is a $(n-k)\times(n-k)$ diagonal matrix with entries $1-\frac{\lambda_{k+1}}{2} \ge \cdots \ge 1-\frac{\lambda_n}{2}$. We can write $U\Sigma^{2t}U = U_{[k]}\Sigma_{[k]}^{2t}U_{[k]}^T +$

$U_{-[k]}\Sigma^{2t}_{-[k]}U^T_{-[k]}$, thus we get

$$\begin{aligned}
\nu_{k+1}(\widetilde{A}) &= \frac{n}{s}\cdot \nu_{k+1}\left(S^T M^{2t} S\right)\\
&= \frac{n}{s}\cdot \nu_{k+1}\left(S^T (U\Sigma^{2t}U^T) S\right)\\
&= \frac{n}{s}\cdot \nu_{k+1}\left(S^T \left(U_{[k]}\Sigma^{2t}_{[k]}U^T_{[k]} + U_{-[k]}\Sigma^{2t}_{-[k]}U^T_{-[k]}\right) S\right)\\
&\le \frac{n}{s}\cdot \nu_{k+1}\left(S^T U_{[k]}\Sigma^{2t}_{[k]}U^T_{[k]} S\right) + \frac{n}{s}\cdot \nu_{\max}\left(S^T U_{-[k]}\Sigma^{2t}_{-[k]}U^T_{-[k]} S\right) \quad \text{By Weyl's inequality (Lemma 3.17)}
\end{aligned}$$

Here $\nu_{k+1}(S^T U_{[k]}\Sigma^{2t}_{[k]}U^T_{[k]}S) = 0$, because the rank of $\Sigma^{2t}_{[k]}$ is k. We then need to bound $\nu_{\max}(S^T U_{-[k]}\Sigma^{2t}_{-[k]}U^T_{-[k]}S)$. We have,

$$\begin{aligned}
&\nu_{\max}\left(S^T U_{-[k]}\Sigma^{2t}_{-[k]}U^T_{-[k]} S\right) = \nu_{\max}\left(U_{-[k]}\Sigma^{2t}_{-[k]}U^T_{-[k]} SS^T\right) && \text{By Lemma 3.8}\\
&\le \nu_{\max}\left(U_{-[k]}\Sigma^{2t}_{-[k]}U^T_{-[k]}\right)\cdot \nu_{\max}\left(SS^T\right) && \text{By submultiplicativity of norm}\\
&= \nu_{\max}\left(\Sigma^{2t}_{-[k]}U^T_{-[k]}U_{-[k]}\right)\cdot \nu_{\max}\left(SS^T\right) && \text{By Lemma 3.8}\\
&= \nu_{\max}\left(\Sigma^{2t}_{-[k]}\right)\cdot \nu_{\max}\left(SS^T\right) && \text{Since } U^T_{-[k]}U_{-[k]} = I
\end{aligned}$$

Next, observe that $SS^T \in \mathbb{R}^{n\times n}$ is a diagonal matrix whose $(a,a)^{\text{th}}$ entry is the multiplicity of vertex a is sampled in S. Thus, $\nu_{\max}(SS^T)$ is the maximum multiplicity over all vertices, which is at most s. Also note that $\nu_{\max}(\Sigma^{2t}_{-[k]}) = \left(1-\frac{\lambda_{k+1}}{2}\right)^{2t}$. Thus by (3.98) we get,

$$\nu_{k+1}(\widetilde{A}) \le \frac{n}{s}\cdot \nu_{\max}\left(S^T U_{-[k]}\Sigma^{2t}_{-[k]}U^T_{-[k]} S\right) \le \frac{n}{s}\cdot s\cdot \left(1-\frac{\lambda_{k+1}}{2}\right)^{2t} \le n\cdot n^{-10} = n^{-9}.$$

□

Now we are ready to prove the main result of this section (Lemma 3.24).

Lemma 3.24. *Let $k \ge 2$ be an integer, $\varphi \in (0,1)$ and $\epsilon \in (0,1)$. Let $G = (V,E)$ be a d-regular and (k,φ,ϵ)-clusterable graph. Let $1/n^8 < \xi < 1$ and $t \ge \frac{20\log n}{\varphi^2}$. Let $c_1 > 1$ and $c_2 > 1$ be a large enough constants. Let $s \ge c_1 \cdot n^{240\epsilon/\varphi^2}\cdot \log n\cdot k^4$ and $R \ge \frac{c_2\cdot k^9\cdot n^{(1/2+820\cdot\epsilon/\varphi^2)}}{\xi^2}$. Let $I_S = \{i_1,\dots,i_s\}$ be a multiset of s indices chosen independently and uniformly at random from $\{1,\dots,n\}$. Let S be the $n\times s$ matrix whose j-th column equals $\mathbb{1}_{i_j}$. Let $\mathcal{G}\in\mathbb{R}^{s\times s}$ be the output of* EstimateCollisionProbabilities(G, I_S, R, t) *(Algorithm 2). Let M be the random walk transition matrix of G. Let $\sqrt{\frac{n}{s}}\cdot M^t S = \widetilde{U}\widetilde{\Sigma}\widetilde{W}^T$ be an SVD of $\sqrt{\frac{n}{s}}\cdot M^t S$ where $\widetilde{U}\in\mathbb{R}^{n\times n}, \widetilde{\Sigma}\in\mathbb{R}^{n\times n}, \widetilde{W}\in\mathbb{R}^{s\times n}$. Let $\frac{n}{s}\cdot\mathcal{G} = \widehat{W}\widehat{\Sigma}\widehat{W}^T$ be an eigendecomposition of $\frac{n}{s}\cdot\mathcal{G}$. If $\frac{\epsilon}{\varphi^2}\le\frac{1}{10^5}$ then with probability at least $1-2\cdot n^{-100}$ matrices $\widehat{\Sigma}^{-2}_{[k]}$ and $\widetilde{\Sigma}^{-4}_{[k]}$ exist and we have*

$$\left\|\widehat{W}_{[k]}\widehat{\Sigma}^{-2}_{[k]}\widehat{W}^T_{[k]} - \widetilde{W}_{[k]}\widetilde{\Sigma}^{-4}_{[k]}\widetilde{W}^T_{[k]}\right\|_2 < \xi$$

Proof. Let $\widetilde{A} = \frac{n}{s} \cdot (M^t S)^T \left(M^t S\right) = \widetilde{W}\widetilde{\Sigma^2}\widetilde{W}^T$ and $\widehat{A} = \frac{n}{s} \cdot \mathcal{G}$. Thus we have

$$\widetilde{A}^2 = \left(\frac{n}{s} \cdot (M^t S)^T \left(M^t S\right)\right)^2 = \widetilde{W}\widetilde{\Sigma^4}\widetilde{W}^T$$

and

$$\widehat{A}^2 = \left(\frac{n}{s} \cdot \mathcal{G}\right)^2 = \widehat{W}\widehat{\Sigma^2}\widehat{W}^T.$$

Recall that for a symmetric matrix A, we write $\nu_i(A)$ to denote the i^{th} largest eigenvalue of A. We want to apply Lemma 3.18 to get

$$\left|\left|\widetilde{W}_{[k]}\widetilde{\Sigma}_{[k]}^{-4}\widetilde{W}_{[k]}^T - \widehat{W}_{[k]}\widehat{\Sigma}_{[k]}^{-2}\widehat{W}_{[k]}^T\right|\right|_2 \le \frac{16 \cdot \|\widetilde{A}^2 - \widehat{A}^2\|_2 + 4 \cdot \nu_{k+1}(\widetilde{A}^2)}{\nu_k(\widetilde{A}^2)^2}$$

Hence, we first need to verify the prerequisites of Lemma 3.18. Let $c_3 > 1$ be a large enough constant that we will define soon, and let $\sigma_{\text{err}} = \frac{\xi \cdot n^{(-1-360\cdot\epsilon/\varphi^2)}}{c_3 \cdot k^2}$. Let c be a constant from Lemma 3.27. By the assumption of the lemma for large enough constant $c_2 > 1$ we have

$$R \ge \frac{c_2 \cdot k^9 \cdot n^{1/2+820\cdot\epsilon/\varphi^2}}{\xi^2} \ge \max\left\{\frac{c \cdot k^5 \cdot n^{-2+100\epsilon/\varphi^2}}{\sigma_{\text{err}}^2}, \frac{c \cdot k \cdot n^{-1/2+20\epsilon/\varphi^2}}{\sigma_{\text{err}}}\right\}.$$

Thus we can apply Lemma 3.27. Hence, with probability at least $1 - n^{-100}$ we have

$$\|\mathcal{G} - (M^t S)^T (M^t S)\|_2 \le s \cdot \sigma_{\text{err}}. \tag{3.101}$$

Therefore we have

$$\begin{aligned}
\|\mathcal{G}^2 - \left((M^t S)^T (M^t S)\right)^2\|_2 &= \|\mathcal{G}\left(\mathcal{G} - (M^t S)^T (M^t S)\right) + \left(\mathcal{G} - (M^t S)^T (M^t S)\right)(M^t S)^T (M^t S)\|_2 \\
&\le \|\mathcal{G} - (M^t S)^T (M^t S)\|_2 \left(\|\mathcal{G}\|_2 + \|(M^t S)^T (M^t S)\|_2\right) \\
&\le s \cdot \sigma_{\text{err}} \left((s \cdot \sigma_{\text{err}} + \|(M^t S)^T (M^t S)\|_2) + \|(M^t S)^T (M^t S)\|_2\right) \\
&= (s \cdot \sigma_{\text{err}})^2 + 2 \cdot s \cdot \sigma_{\text{err}} \|(M^t S)^T (M^t S)\|_2
\end{aligned} \tag{3.102}$$

Note that

$$\begin{aligned}
\|(M^t S)^T (M^t S)\|_2 &\le \|(M^t S)^T (M^t S)\|_F \\
&= \sqrt{\sum_{x,y \in S} \left((M^t \mathbb{1}_x)^T (M^t \mathbb{1}_y)\right)^2} \\
&\le \sqrt{\sum_{x,y \in S} \|M^t \mathbb{1}_x\|_2^2 \|M^t \mathbb{1}_y\|_2^2} && \text{By Cauchy Schwarz} \\
&\le O\left(\sqrt{s^2 \cdot \left(k^2 \cdot n^{-1+(40\epsilon/\varphi^2)}\right)^2}\right) && \text{By Lemma 3.22} \\
&= O\left(s \cdot k^2 \cdot n^{-1+(40\epsilon/\varphi^2)}\right) && .
\end{aligned} \tag{3.103}$$

Puuting (3.103) and (3.102) and by choice of $\sigma_{\text{err}} = \frac{\xi \cdot n^{(-1-360\cdot\epsilon/\varphi^2)}}{c_3 \cdot k^2}$ we get

$$\|\widetilde{A}^2 - \widehat{A}^2\|_2 = \left(\frac{n}{s}\right)^2 \|\mathscr{G}^2 - \left((M^t S)^T (M^t S)\right)^2\|_2 \leq O\left(\frac{\xi^2 \cdot n^{-720\cdot\epsilon/\varphi^2}}{(c_3)^2 \cdot k^4} + \frac{\xi \cdot n^{-320\epsilon/\varphi^2}}{c_3}\right) = O\left(\frac{\xi \cdot n^{-320\epsilon/\varphi^2}}{c_3}\right) \tag{3.104}$$

By Lemma 3.8 for any $i \in [s]$ we have

$$\nu_i(\widetilde{A}) = \nu_i\left(\frac{n}{s} \cdot (M^t S)\left(M^t S\right)^T\right) = \nu_i\left(\frac{n}{s} \cdot (M^t S)^T \left(M^t S\right)\right)$$

Let c_1 be the constant from Lemma 3.28. Since $s \geq c_1 \cdot n^{240\epsilon/\varphi^2} \cdot \log n \cdot k^4$ therefore by Lemma 3.28 with probability at least $1 - n^{-100}$ we have

$$\nu_k\left(\widetilde{A}^2\right) = \nu_k\left(\left(\frac{n}{s} \cdot (M^t S)^T \left(M^t S\right)\right)^2\right) \geq \left(\frac{n^{-80\epsilon/\varphi^2}}{2}\right)^2 \geq \frac{n^{-160\epsilon/\varphi^2}}{4} \tag{3.105}$$

and

$$\nu_{k+1}\left(\widetilde{A}^2\right) = \nu_{k+1}\left(\left(\frac{n}{s} \cdot (M^t S)^T \left(M^t S\right)\right)^2\right) \leq n^{-18} \tag{3.106}$$

By the bound on the $\nu_k(\widetilde{A}^2)$ and the inequality on $\|\widetilde{A}^2 - \widehat{A}^2\|_2$, we know that $\nu_k(\widehat{A}^2)$ is non-zero and so $\widehat{\Sigma}_{[k]}^{-2}$ exist. Recall that $\widetilde{A} = \widetilde{W}\widetilde{\Sigma}^2\widetilde{W}^T$. Observing that $\widetilde{A}$ is positive semi-definite, $\nu_{k+1}(\widetilde{A}^2) < \nu_k(\widetilde{A}^2)/4$, and $\|\widetilde{A}^2 - \widehat{A}^2\|_2 \leq \frac{1}{100} \cdot \nu_k(\widetilde{A}^2)$ we can apply Lemma 3.18 and we get

$$\begin{aligned}
\left\|\widetilde{W}_{[k]}\widetilde{\Sigma}_{[k]}^{-4}\widetilde{W}_{[k]}^T - \widehat{W}_{[k]}\widehat{\Sigma}_{[k]}^{-2}\widehat{W}_{[k]}^T\right\|_2 &\leq \frac{16 \cdot \|\widetilde{A}^2 - \widehat{A}^2\|_2 + 4 \cdot \nu_{k+1}(\widetilde{A}^2)}{\nu_k(\widetilde{A}^2)^2} \\
&\leq \frac{O\left(\frac{\xi \cdot n^{(-320\epsilon/\varphi^2)}}{c_3}\right) + 4 \cdot n^{-18}}{\frac{1}{16} \cdot n^{(-320\epsilon/\varphi^2)}} && \text{By (3.104) and (3.105)} \\
&\leq O\left(\frac{\xi}{c_3}\right) + 64 \cdot n^{-17} \\
&\leq \xi
\end{aligned}$$

The last inequality holds since $\xi \geq n^{-8}$ and by setting c_3 to a large enough constant to cancel the constant hidden in $O\left(\frac{\xi}{c_3}\right)$. □

3.5.5 Proof of Theorem 3.2

Theorem 3.2. *[Spectral Dot Product Oracle] Let $\epsilon, \varphi \in (0,1)$ with $\epsilon \leq \frac{\varphi^2}{10^5}$. Let $G = (V, E)$ be a d-regular graph that admits a (k, φ, ϵ)-clustering $C_1, \ldots, C_k$. Let $\frac{1}{n^5} < \xi < 1$. Then* INITIALIZEORACLE$(G, 1/2, \xi)$ *(Algorithm 4) computes in time $(\frac{k}{\xi})^{O(1)} \cdot n^{1/2+O(\epsilon/\varphi^2)} \cdot (\log n)^3 \cdot \frac{1}{\varphi^2}$ a sublinear space data structure $\mathscr{D}$ of size $(\frac{k}{\xi})^{O(1)} \cdot n^{1/2+O(\epsilon/\varphi^2)} \cdot (\log n)^3$ such that with probability at least $1 - n^{-100}$ the following property is satisfied:*

For every pair of vertices $x, y \in V$, SPECTRALDOTPRODUCT$(G, x, y, 1/2, \xi, \mathscr{D})$ *(Algorithm 5) com-*

putes an output value $\langle f_x, f_y\rangle_{apx}$ *such that with probability at least* $1-n^{-100}$

$$\left|\,\langle f_x, f_y\rangle_{apx} - \langle f_x, f_y\rangle\right| \le \frac{\xi}{n}.$$

The running time of $\textsc{SpectralDotProduct}(G,x,y,1/2,\xi,\mathscr{D})$ *is* $(\frac{k}{\xi})^{O(1)}\cdot n^{1/2+O(\epsilon/\varphi^2)}\cdot(\log n)^2\cdot\frac{1}{\varphi^2}$.

Furthermore, for any $0\le\delta\le 1/2$*, one can obtain the following trade-offs between preprocessing time and query time: Algorithm* $\textsc{SpectralDotProduct}(G,x,y,\delta,\xi,\mathscr{D})$ *requires* $(\frac{k}{\xi})^{O(1)}\cdot n^{\delta+O(\epsilon/\varphi^2)}\cdot(\log n)^2\cdot\frac{1}{\varphi^2}$ *per query when the prepressing time of Algorithm* $\textsc{InitializeOracle}(G,\delta,\xi)$ *is increased to* $(\frac{k}{\xi})^{O(1)}\cdot n^{1-\delta+O(\epsilon/\varphi^2)}\cdot(\log n)^3\cdot\frac{1}{\varphi^2}$.

To prove Theorem 3.2 we need to combine Lemma 3.19 from Section 3.5.3 with the following lemma.

Lemma 3.29. *Let* $G=(V,E)$ *be a* d*-regular and* (k,φ,ϵ)*-clusterable graph. Let* $0<\delta<1/2$*, and* $1/n^6<\xi<1$*. Let* $\mathscr{D}$ *denote the data structure constructed by Algorithm* $\textsc{InitializeOracle}(G,\delta,\xi)$ *(Algorithm 4). Let* $x,y\in V$*. Let* $\langle f_x, f_y\rangle_{apx}\in\mathbb{R}$ *denote the value returned by* $\textsc{SpectralDotProductOracle}(G,x,y,\delta,\xi,\mathscr{D}$ *(Algorithm 5). Let* $t\ge\frac{20\log n}{\varphi^2}$*. Let* $c>1$ *be a large enough constant and let* $s\ge c\cdot n^{240\cdot\epsilon/\varphi^2}\cdot\log n\cdot k^4$*. Let* $I_S=\{i_1,\dots,i_s\}$ *be a multiset of* s *indices chosen independently and uniformly at random from* $\{1,\dots,n\}$*. Let* S *be the* $n\times s$ *matrix whose* j*-th column equals* $\mathbb{1}_{i_j}$*. Let* M *be the random walk transition matrix of* G*. Let* $\sqrt{\frac{n}{s}}\cdot M^tS=\widetilde{U}\widetilde{\Sigma}\widetilde{W}^T$ *be an SVD of* $\sqrt{\frac{n}{s}}\cdot M^tS$ *where* $\widetilde{U}\in\mathbb{R}^{n\times n},\widetilde{\Sigma}\in\mathbb{R}^{n\times n},\widetilde{W}\in\mathbb{R}^{s\times n}$*. If* $\frac{\epsilon}{\varphi^2}\le\frac{1}{10^5}$*, and Algorithm 4 succeeds, then with probability at least* $1-n^{-100}$ *matrix* $\widetilde{\Sigma}_{[k]}^{-4}$ *exists and we have*

$$\left|\langle f_x, f_y\rangle_{apx}-(M^t\mathbb{1}_x)^T(M^tS)\left(\frac{n}{s}\cdot\widetilde{W}_{[k]}\widetilde{\Sigma}_{[k]}^{-4}\widetilde{W}_{[k]}^T\right)(M^tS)^T(M^t\mathbb{1}_y)\right|<\frac{\xi}{n}.$$

Proof. Note that as per line 7 of Algorithm 5 $\langle f_x, f_y\rangle_{apx}$ is defined as

$$\langle f_x, f_y\rangle_{apx}=\alpha_x^T\Psi\alpha_y.$$

where as per line 3 of Algorithm 4 we define matrix $\Psi\in\mathbb{R}^{s\times s}$ as

$$\Psi=\frac{n}{s}\cdot\widehat{W}_{[k]}\widehat{\Sigma}_{[k]}^{-2}\widehat{W}_{[k]}^T,$$

and $\alpha_x,\alpha_y\in\mathbb{R}^s$ are vectors obtained by taking entrywise median over all $(\widehat{Q}_i)^T(\widehat{m}_x^i)$ and $(\widehat{Q}_i)^T(\widehat{m}_y^i)$. (See line 5 and 6 of Algorithm 5). For any vertex $a\in V$ recall that m_a denote $m_a=M^t\mathbb{1}_a$. We then define

$$\mathbf{a}_x=m_x^T(M^tS),\quad A=\frac{n}{s}\cdot\widetilde{W}_{[k]}\widetilde{\Sigma}_{[k]}^{-4}\widetilde{W}_{[k]}^T,\quad \mathbf{a}_y=(M^tS)^Tm_y,\text{ and}$$

$$\mathbf{e}_x=\alpha_x^T-\mathbf{a}_x,\quad E=\Psi-A,\quad \mathbf{e}_y=\alpha_y-\mathbf{a}_y$$

Thus by triangle inequality we have

$$\begin{aligned}
&\left\|\alpha_x^T \Psi \alpha_y - m_x^T (M^t S) \left(\frac{n}{s} \cdot \widetilde{W}_{[k]} \widetilde{\Sigma}_{[k]}^{-4} \widetilde{W}_{[k]}^T\right) (M^t S)^T m_y\right\|_2 \\
&= \| \left(\mathbf{a}_x + \mathbf{e}_x\right) (A + E) \left(\mathbf{a}_y + \mathbf{e}_y\right) - \mathbf{a}_x A \mathbf{a}_y \|_2 \\
&\leq \|\mathbf{e}_x\|_2 \|A\|_2 \|\mathbf{a}_y\|_2 + \|\mathbf{a}_x\|_2 \|E\|_2 \|\mathbf{a}_y\|_2 + \|\mathbf{a}_x\|_2 \|A\|_2 \|\mathbf{e}_y\|_2 \\
&+ \|\mathbf{e}_x\|_2 \|E\|_2 \|\mathbf{a}_y\|_2 + \|\mathbf{a}_x\|_2 \|E\|_2 \|\mathbf{e}_y\|_2 + \|\mathbf{e}_x\|_2 \|A\|_2 \|\mathbf{e}_y\|_2 + \|\mathbf{e}_x\|_2 \|E\|_2 \|\mathbf{e}_y\|_2
\end{aligned}$$

Therefore we need to bound $\|\mathbf{e}_x\|_2$, $\|\mathbf{e}_y\|_2$, $\|E\|_2$, $\|\mathbf{a}_x\|_2$, $\|\mathbf{a}_y\|_2$ and $\|A\|_2$. Let $c' > 1$ be a constant we will define soon, and let $\xi' = \frac{\xi}{c' \cdot k^4 \cdot n^{80\epsilon/\varphi^2}}$. Let c_1 be a constant in front of s and let c_2 be a constant in front of R in Lemma 3.24. Thus for large enough c we have $s \geq c_1 \cdot n^{240\epsilon/\varphi^2} \cdot \log n \cdot k^4$ and $R_{\text{init}} = \Theta(n^{1-\delta+980 \cdot \epsilon/\varphi^2} \cdot k^{17}/\xi^2) \geq \frac{c_2 \cdot k^9 \cdot n^{1/2+820 \cdot \epsilon/\varphi^2}}{\xi'^2}$ as per line 2 of Algorithm 4, hence, by Lemma 3.24 applied with ξ' we have with probability at least $1 - n^{-100}$, $\widehat{W}_{[k]}^T-$ and $\widetilde{\Sigma}_{[k]}^{-4}$ exist and we have

$$\|E\|_2 = \frac{n}{s} \cdot \left\|\widehat{W}_{[k]} \widehat{\Sigma}_{[k]}^{-2} \widehat{W}_{[k]}^T - \widetilde{W}_{[k]} \widetilde{\Sigma}_{[k]}^{-4} \widetilde{W}_{[k]}^T\right\|_2 \leq \frac{n}{s} \cdot \xi' = \frac{\xi \cdot n}{c' \cdot k^4 \cdot n^{80\epsilon/\varphi^2} \cdot s}. \tag{3.107}$$

Recall that for a symmetric matrix A, we write $\nu_i(A)$ (resp. $\nu_{\max}(A), \nu_{\min}(A)$) to denote the i^{th} largest (resp. maximum, minimum) eigenvalue of A. We have

$$\|A\|_2 = \frac{n}{s} \cdot \|\widetilde{W}_{[k]} \widetilde{\Sigma}_{[k]}^{-4} \widetilde{W}_{[k]}^T\|_2 = \frac{n}{s} \cdot \nu_{\max}\left(\widetilde{W}_{[k]} \widetilde{\Sigma}_{[k]}^{-4} \widetilde{W}_{[k]}^T\right) = \frac{n}{s} \cdot \frac{1}{\nu_k\left(\widetilde{W}_{[k]} \widetilde{\Sigma}_{[k]}^{4} \widetilde{W}_{[k]}^T\right)}$$

Note that $\frac{n}{s} \cdot (M^t S)^T (M^t S) = \widetilde{W} \widetilde{\Sigma}^2 \widetilde{W}^T$. Thus by Lemma 3.28 item (1) we have

$$\nu_k\left(\widetilde{W}_{[k]} \widetilde{\Sigma}_{[k]}^{4} \widetilde{W}_{[k]}^T\right) = \nu_k\left(\widetilde{W} \widetilde{\Sigma}^4 \widetilde{W}^T\right) = \nu_k\left(\left(\frac{n}{s} \cdot (M^t S)^T (M^t S)\right)^2\right) \geq \frac{n^{-160\epsilon/\varphi^2}}{4}.$$

Therefore we have

$$\|A\|_2 \leq 4 \cdot \frac{n}{s} \cdot n^{160\epsilon/\varphi^2} = \frac{4 \cdot n^{1+160\epsilon/\varphi^2}}{s}. \tag{3.108}$$

Since G is (k, φ, ϵ)-clusterable by Lemma 3.22 for any vertex $x \in V$ we have

$$\|m_x\|_2^2 \leq O\left(k^2 \cdot n^{-1+(40\epsilon/\varphi^2)}\right). \tag{3.109}$$

Then we get

$$\begin{aligned}
\|\mathbf{a}_x\|_2 &= \|(m_x)^T(M^tS)\|_2 \\
&= \sqrt{\sum_{a\in I_S}\left((m_x)^T(m_a)\right)^2} \\
&\le \sqrt{\sum_{a\in I_S}\|m_x\|_2^2\|m_a\|_2^2} && \text{By Cauchy Schwarz} \\
&\le O\left(\sqrt{s\cdot\left(k^2\cdot n^{-1+(40\epsilon/\varphi^2)}\right)^2}\right) && \text{By (3.109)} \\
&= O\left(\sqrt{s}\cdot k^2\cdot n^{-1+(40\epsilon/\varphi^2)}\right) && (3.110)
\end{aligned}$$

By the same analysis we get

$$\|\mathbf{a}_y\|_2 \le O\left(\sqrt{s}\cdot k^2\cdot n^{-1+(40\epsilon/\varphi^2)}\right) \tag{3.111}$$

Now we left to bound $\|\mathbf{e}_x\|_2$ and $\|\mathbf{e}_y\|_2$. Recall that $\mathbf{e}_x = \alpha_x - (M^t\mathbb{1}_x)^T(M^tS)$ where $\alpha_x, \alpha_y \in \mathbb{R}^s$ are vectors obtained by taking entrywises median over all $(\widehat{Q}_i)^T(\widehat{m}_x^i)$ and $(\widehat{Q}_i)^T(\widehat{m}_y^i)$. (See line 5 and 6 of Algorithm 5). Also note that as per line 3 and line 4 of Algorithm 5, $\widehat{m}_x^i$ and $\widehat{m}_y^i$ are defined as the empirical probability distribution of running R_{query} random walks of length t starting from vertex x and y. Also note that $\widehat{Q}_i$s are generated by Algorithm 3 which runs R_{init} random walks from vertices in I_S. For any $z \in I_S$ any $i \in \{1,\dots,O(\log n)\}$ let $\mathbf{q}_z^i$ denote the column corresponding to vertex z in $\widehat{Q}_i$.

Let c_3 be a constant in front of R_1 and R_2 in Lemma 3.26. Let $\sigma_{\text{err}} = \frac{\xi}{c'\cdot k^2\cdot n^{(1+200\epsilon/\varphi^2)}}$. Thus by choice of $R_{\text{init}} = \Theta(n^{1-\delta+980\cdot\epsilon/\varphi^2}\cdot k^{17}/\xi^2)$ as per line 2 of Algorithm 4 and $R_{\text{query}} = \Theta(n^{\delta+500\cdot\epsilon/\varphi^2}\cdot k^9/\xi^2)$ as per line 1 of Algorithm 5, the prerequisites of Lemma 3.26 are satisfied:

$$\min(R_{\text{init}}, R_{\text{query}}) \ge \frac{c_3\cdot k^5\cdot n^{-2+(100\epsilon/\varphi^2)}}{\sigma_{\text{err}}^2}, \text{ and, } R_{\text{init}}\cdot R_{\text{query}} \ge \frac{c_3\cdot k^2\cdot n^{-1+(40\epsilon/\varphi^2)}}{\sigma_{\text{err}}^2}$$

Thus we can apply Lemma 3.26. Hence, for any $z \in I_S$ with probability at least 0.99 we have

$$|(\widehat{m}_x^i)^T\mathbf{q}_z^i - (m_x)^T(m_z)| \le \sigma_{\text{err}}$$

Note that as per line 5 and line 6 of Algorithm 5 we take entrywise median over all $(\widehat{Q}_i)^T(\widehat{m}_x^i)$ and $(\widehat{Q}_i)^T(\widehat{m}_y^i)$. Since we are running $O(\log n)$ copies of the same algorithm with success probability at least 0.99, thus by simple Chernoff bound with probability at least $1-n^{-100}$ for all $z \in I_S$ we have

$$|\alpha_x(z) - (m_x)^T(m_z)| \le \sigma_{\text{err}}$$

Therefore by choice of $\sigma_{\text{err}} = \frac{\xi}{c'\cdot k^2\cdot n^{1+200\cdot\epsilon/\varphi^2}}$ we get

$$\|\mathbf{e}_x\|_2 = \|\alpha_x - (m_x)^T(M^tS)\|_2 \le \sqrt{s}\cdot\sigma_{\text{err}} = \frac{\sqrt{s}\cdot\xi}{c'\cdot k^2\cdot n^{(1+200\epsilon/\varphi^2)}}. \tag{3.112}$$

By the same analysis we get

$$\|\mathbf{e}_y\|_2 \leq \frac{\sqrt{s}\cdot\xi}{c'\cdot k^2\cdot n^{(1+200\epsilon/\varphi^2)}}. \tag{3.113}$$

Putting (3.107), (3.108), (3.109), (3.110), (3.111), (3.112), and (3.113) and for large enough n we get:

$$\begin{aligned}
&\left|\left|\left\langle f_x, f_y\right\rangle_{apx} - \cdot m_x^T (M^t S)\left(\frac{n}{s}\cdot\widetilde{W}_{[k]}\widetilde{\Sigma}_{[k]}^{-4}\widetilde{W}_{[k]}^T\right)(M^t S)^T m_y\right|\right| \leq \\
&\|\mathbf{e}_x\|_2\|A\|_2\|\mathbf{a}_y\|_2 + \|\mathbf{a}_x\|_2\|E\|_2\|\mathbf{a}_y\|_2 + \|\mathbf{a}_x\|_2\|A\|_2\|\mathbf{e}_y\|_2 + \\
&\|\mathbf{e}_x\|_2\|E\|_2\|\mathbf{a}_y\|_2 + \|\mathbf{a}_x\|_2\|E\|_2\|\mathbf{e}_y\|_2 + \|\mathbf{e}_x\|_2\|A\|_2\|\mathbf{e}_y\|_2 + \|\mathbf{e}_x\|_2\|E\|_2\|\mathbf{e}_y\|_2 \\
&\leq 2\cdot\left(\frac{\sqrt{s}\cdot\xi}{c'\cdot k^2\cdot n^{(1+200\epsilon/\varphi^2)}}\right)\left(\frac{4\cdot n^{1+160\epsilon/\varphi^2}}{s}\right)\cdot O\left(\sqrt{s}\cdot k^2\cdot n^{-1+(40\epsilon/\varphi^2)}\right) \\
&+ 2\cdot\left(\frac{\sqrt{s}\cdot n^{(80\epsilon/\varphi^2)}k^2}{n}\right)\left(\frac{\xi\cdot n}{c'\cdot k^4\cdot n^{80\epsilon/\varphi^2}\cdot s}\right)\left(\frac{\xi}{c'\cdot\sqrt{s}\cdot n^{(1+20\epsilon/\varphi^2)}}\right) \\
&+ \left(\frac{\sqrt{s}\cdot\xi}{c'\cdot k^2\cdot n^{(1+200\epsilon/\varphi^2)}}\right)^2\left(\frac{4\cdot n^{1+160\epsilon/\varphi^2}}{s}\right) \\
&+ O\left(\sqrt{s}\cdot k^2\cdot n^{-1+(40\epsilon/\varphi^2)}\right)^2\left(\frac{\xi\cdot n}{c'\cdot k^4\cdot n^{80\epsilon/\varphi^2}\cdot s}\right) \\
&+ \left(\frac{\sqrt{s}\cdot\xi}{c'\cdot k^2\cdot n^{(1+200\epsilon/\varphi^2)}}\right)^2\left(\frac{\xi\cdot n}{c'\cdot k^4\cdot n^{80\epsilon/\varphi^2}\cdot s}\right) \\
&\leq O\left(\frac{\xi}{c'\cdot n}\right) \\
&\leq \frac{\xi}{n}.
\end{aligned}$$

The last inequality holds by setting c' to a large enough constant to cancel the hidden constant of $O\left(\frac{\xi}{c'\cdot n}\right)$. □

Now we are able to complete the proof of Theorem 3.2.

Theorem 3.2. *[Spectral Dot Product Oracle] Let $\epsilon, \varphi \in (0,1)$ with $\epsilon \leq \frac{\varphi^2}{10^5}$. Let $G = (V, E)$ be a d-regular graph that admits a (k, φ, ϵ)-clustering $C_1, \ldots, C_k$. Let $\frac{1}{n^5} < \xi < 1$. Then* INITIALIZEORACLE$(G, 1/2, \xi)$ *(Algorithm 4) computes in time $(\frac{k}{\xi})^{O(1)}\cdot n^{1/2+O(\epsilon/\varphi^2)}\cdot(\log n)^3\cdot\frac{1}{\varphi^2}$ a sublinear space data structure $\mathcal{D}$ of size $(\frac{k}{\xi})^{O(1)}\cdot n^{1/2+O(\epsilon/\varphi^2)}\cdot(\log n)^3$ such that with probability at least $1 - n^{-100}$ the following property is satisfied:*

For every pair of vertices $x, y \in V$, SPECTRALDOTPRODUCT$(G, x, y, 1/2, \xi, \mathcal{D})$ *(Algorithm 5) computes an output value $\left\langle f_x, f_y\right\rangle_{apx}$ such that with probability at least $1 - n^{-100}$*

$$\left|\left\langle f_x, f_y\right\rangle_{apx} - \langle f_x, f_y\rangle\right| \leq \frac{\xi}{n}.$$

The running time of SPECTRALDOTPRODUCT$(G,x,y,1/2,\xi,\mathcal{D})$ *is* $(\frac{k}{\xi})^{O(1)}\cdot n^{1/2+O(\epsilon/\varphi^2)}\cdot(\log n)^2\cdot\frac{1}{\varphi^2}$.

Furthermore, for any $0\le\delta\le 1/2$, *one can obtain the following trade-offs between preprocessing time and query time: Algorithm* SPECTRALDOTPRODUCT$(G,x,y,\delta,\xi,\mathcal{D})$ *requires* $(\frac{k}{\xi})^{O(1)}\cdot n^{\delta+O(\epsilon/\varphi^2)}\cdot(\log n)^2\cdot\frac{1}{\varphi^2}$ *per query when the prepressing time of Algorithm* INITIALIZEORACLE(G,δ,ξ) *is increased to* $(\frac{k}{\xi})^{O(1)}\cdot n^{1-\delta+O(\epsilon/\varphi^2)}\cdot(\log n)^3\cdot\frac{1}{\varphi^2}$.

Proof of Theorem 3.2. **Correctness:** Note that as per line 3 of Algorithm 4 we set $s=\Theta(n^{480\cdot\epsilon/\varphi^2}\cdot\log n\cdot k^8/\xi^2)$. Recall that $I_S=\{i_1,\dots,i_s\}$ is the multiset of s vertices each sampled uniformly at random (see line 4 of Algorithm 4). Let S be the $n\times s$ matrix whose j-th column equals $\mathbb{1}_{i_j}$. Recall that M is the random walk transition matrix of G. Let $\sqrt{\frac{n}{s}}\cdot M^tS=\widetilde{U}\widetilde{\Sigma}\widetilde{W}^T$ be the eigendecomposition of $\sqrt{\frac{n}{s}}\cdot M^tS$. We define

$$e_1=\left|(M^t\mathbb{1}_x)^T(M^tS)\left(\frac{n}{s}\cdot\widetilde{W}_{[k]}\widetilde{\Sigma}_{[k]}^{-4}\widetilde{W}_{[k]}^T\right)(M^tS)^T(M^t\mathbb{1}_y)-\mathbb{1}_x^TU_{[k]}U_{[k]}^T\mathbb{1}_y\right|$$

and

$$e_2=\left|\left\langle f_x,f_y\right\rangle_{apx}-(M^t\mathbb{1}_x)^T(M^tS)\left(\frac{n}{s}\cdot\widetilde{W}_{[k]}\widetilde{\Sigma}_{[k]}^{-4}\widetilde{W}_{[k]}^T\right)(M^tS)^T(M^t\mathbb{1}_y)\right|$$

By triangle inequality we have

$$\left|\left\langle f_x,f_y\right\rangle_{apx}-\langle f_x,f_y\rangle\right|=\left|\left\langle f_x,f_y\right\rangle_{apx}-\mathbb{1}_x^TU_{[k]}U_{[k]}^T\mathbb{1}_y\right|\le e_1+e_2.$$

Let $\xi'=\xi/2$. Let c be a constant in front of s in Lemma 3.19 and c' be a constant in front of s in Lemma 3.29. Note that as per line 3 of Algorithm 4 we set $s=\Theta(n^{480\cdot\epsilon/\varphi^2}\cdot\log n\cdot k^8/\xi^2)$. Since $\frac{\epsilon}{\varphi^2}\le\frac{1}{10^5}$ and $s\ge c\cdot n^{480\epsilon/\varphi^2}\cdot\log n\cdot k^8/\xi'^2$ by Lemma 3.19 with probability at least $1-n^{-100}$ we have $e_1\le\frac{\xi'}{n}=\frac{\xi}{2\cdot n}$. Since $s\ge c'\cdot n^{240\epsilon/\varphi^2}\cdot\log n\cdot k^4$, by Lemma 3.29 with probability at least $1-2\cdot n^{-100}$ we have $e_2\le\frac{\xi}{2\cdot n}$. Thus with probability at least $1-3\cdot n^{-100}$ we have

$$\left|\left\langle f_x,f_y\right\rangle_{apx}-\langle f_x,f_y\rangle\right|\le e_1+e_2\le\frac{\xi}{2\cdot n}+\frac{\xi}{2\cdot n}\le\frac{\xi}{n}.$$

Space and runtime of INITIALIZEORACLE: Algorithm INITIALIZEORACLE(G,δ,ξ) (Algorithm 4) samples a set I_S. Then as per line 6 of Algorithm 4 it estimates the empirical probability distribution of random walks starting from any vertex $x\in I_S$ for $O(\log n)$ times. To that end as per line 2 of Algorithm 3 it runs R_{init} random walks of length t from each vertex $x\in I_S$. So it takes $O(\log n\cdot s\cdot R_{\text{init}}\cdot t)$ time and requires $O(\log n\cdot s\cdot R_{\text{init}})$ space to store endpoints of random walks. Then as per line 7 of Algorithm 4 it estimates matrix $\mathcal{G}$ such that the entry corresponding to the x^{th} row and y^{th} column of $\mathcal{G}$ is an estimation of pairwise collision probability of random walks starting from $x,y\in I_S$. To compute $\mathcal{G}$ we call Algorithm ESTIMATECOLLISIONPROBABILITIES$(G,I_S,R_{\text{init}},t)$ (Algorithm 2) for $O(\log n)$ times. Algorithm 2 runs R_{init} random walks of length t from each vertex $x\in I_S$, hence, It takes $O(s\cdot R_{\text{init}}\cdot t\cdot\log n)$ time and it requires $O(s^2\cdot\log n)$

space to store matrix $\mathscr{G}$. Then as per line 8 of Algorithm 4 we compute the SVD of matrix $\mathscr{G}$ in time $O(s^3)$. Thus overall Algorithm 4 runs in time $O\left(\log n \cdot s \cdot R_{\text{init}} \cdot t + s^3\right)$. Thus, by choice of $t = \Theta\left(\frac{\log n}{\varphi^2}\right)$, $R_{\text{init}} = \Theta(n^{1-\delta+980\cdot\epsilon/\varphi^2} \cdot k^{17}/\xi^2)$ and $s = \Theta(n^{480\cdot\epsilon/\varphi^2} \cdot \log n \cdot k^8/\xi^2)$ as in Algorithm 4 we get that Algorithm 4 runs in time $O\left(\log n \cdot s \cdot R_{\text{init}} \cdot t + s^3\right) = (\frac{k}{\xi})^{O(1)} \cdot n^{1-\delta+O(\epsilon/\varphi^2)} \cdot \log^3 n \cdot \frac{1}{\varphi^2}$ and returns a data structure of size $O\left(s^2 + \log n \cdot s \cdot R_{\text{init}}\right) = (\frac{k}{\xi})^{O(1)} \cdot n^{1-\delta+O(\epsilon/\varphi^2)} \cdot \log^2 n$.

Space and runtime of SPECTRALDOTPRODUCTORACLE: Algorithm SPECTRALDOTPRODUCTORACLE$(G, x, y, \delta, \xi, \mathscr{D})$(Algorithm 5) repeats $O(\log n)$ copies of the following procedure: it runs R_{query} random walks of lenght t from vertex x and vertex y, then it computes $\widehat{m}_x \cdot \widehat{Q}_i$ and $\widehat{m}_y \cdot \widehat{Q}_i$. Since $\widehat{Q}_i \in \mathbb{R}^{n\times s}$ has s columns and since $\widehat{m}_x$ has at most R_{query} non-zero entries, thus one can compute $\widehat{m}_x \cdot \widehat{Q}_i$ in time $R_{\text{query}} \cdot s$. Finally Algorithm 5 take entry-wises median of computed vectors (see line 5 and line 6 of Algorithm 5), and returns value $\alpha_x \Psi \alpha_y$ (see line 7 of Algorithm 5). Since $\alpha_x, \alpha_y \in \mathbb{R}^s$ and $\Psi \in \mathbb{R}^{s\times s}$ one can compute $\alpha_x \Psi \alpha_y$ in time $O(s^2)$. Thus overall Algorithm 5 takes $O\left(t \cdot R_{\text{query}} \cdot \log n + s \cdot R_{\text{query}} \cdot \log n + s^2\right)$ time and $O\left(R_{\text{query}} \cdot \log n + s \cdot R_{\text{query}} \cdot \log n + s^2\right)$ space. Thus, by choice of $t = \Theta\left(\frac{\log n}{\varphi^2}\right)$, $R_{\text{query}} = \Theta(n^{\delta+500\cdot\epsilon/\varphi^2} \cdot k^9/\xi^2)$ and $s = \Theta(n^{480\cdot\epsilon/\varphi^2} \cdot \log n \cdot k^8/\xi^2)$ as in Algorithm 4 and Algorithm 5 we get that the Algorithm 5 runs in time $(\frac{k}{\xi})^{O(1)} \cdot n^{\delta+O(\epsilon/\varphi^2)} \cdot \frac{\log^2 n}{\varphi^2}$ and returns a data structure of size $(\frac{k}{\xi})^{O(1)} \cdot n^{\delta+O(\epsilon/\varphi^2)} \cdot \log^2 n$.

□

3.5.6 Computing approximate norms and spectral dot products (Proof of Theorem 3.6)

To design the clustering algorithm in Section 3.6, since we cannot evaluate the dot-product of the spectral embedding exactly in sublinear time, we prove that it is enough to have access to approximate dot-product of the spectral embedding. In Algorithm 7, Algorithm 9 and throughout the analysis of in Section 3.6 we will use $\langle\cdot,\cdot\rangle_{apx}$ to denote approximate spectral dot products and $\|\cdot\|_{apx}$ to denote the approximate norm of a vector. Let $r \in [k]$ and $B, B_1, \ldots, B_r \subseteq V$. Let $\widehat{\mu}, \widehat{\mu}_1, \ldots, \widehat{\mu}_r \in \mathbb{R}^k$ where $\widehat{\mu} = \frac{\sum_{z\in B} f_z}{|B|}$ and $\widehat{\mu}_i = \frac{\sum_{z\in B_i} f_z}{|B_i|}$. All dot products we will try to approximate in Section 3.6 will be of the form $\left\langle f_x, \widehat{\Pi}(\widehat{\mu})\right\rangle$ and all the norms that we approximate are of the form $\left\|\widehat{\Pi}(\widehat{\mu})\right\|_{apx}$, where $x \in V$ and $\widehat{\Pi}$ is defined as a orthogonal projection onto $span(\{\widehat{\mu}_1, \ldots, \widehat{\mu}_r\})^\perp$. To compute such dot products we call Algorithm 6 in the following way (see Corollary 3.1):

$$\left\langle f_x, \widehat{\Pi}\widehat{\mu}\right\rangle_{apx} := \frac{1}{|B|} \cdot \sum_{y\in B} \left\langle f_x, \widehat{\Pi} f_y\right\rangle_{apx}, \tag{3.114}$$

$$\left\|\widehat{\Pi}\widehat{\mu}\right\|^2_{apx} := \frac{1}{|B|} \cdot \sum_{x\in B} \left\langle f_x, \widehat{\Pi}\widehat{\mu}\right\rangle_{apx}. \tag{3.115}$$

```
Algorithm 6 DotProductOracleOnSubspace$(G, x, y, \delta, \xi, \mathcal{D}, B_1, \dots, B_r)$  ▷ Need: $\epsilon/\varphi^2 \le \frac{1}{10^5}$
                                                   ▷ $\mathcal{D} := \{\Psi, \widehat{Q}_1, \dots, \widehat{Q}_{O(\log n)}\}$
Let $X \in \mathbb{R}^{r \times r}, h_x \in \mathbb{R}^r, h_y \in \mathbb{R}^r$.
Let $\xi' := \Theta(\xi \cdot n^{(-80\epsilon/\varphi^2)} \cdot k^{-6})$
for $i, j$ in $[r]$ do
    $X(i,j) := \frac{1}{|B_i||B_j|} \cdot \sum_{z_i \in B_i} \sum_{z_j \in B_j}$ SpectralDotProduct$(G, z_i, z_j, \delta, \xi', \mathcal{D})$
                                                   ▷ $X(i,j) = \langle \widehat{\mu}_i, \widehat{\mu}_j \rangle_{apx}$
for $i$ in $[r]$ do
    $h_x(i) := \frac{1}{|B_i|} \cdot \sum_{z_i \in B_i}$ SpectralDotProduct$(G, z_i, x, \delta, \xi', \mathcal{D})$    ▷ $h_x(i) = \langle \widehat{\mu}_i, f_x \rangle_{apx}$
    $h_y(i) := \frac{1}{|B_i|} \cdot \sum_{z_i \in B_i}$ SpectralDotProduct$(G, z_i, y, \delta, \xi', \mathcal{D})$    ▷ $h_y(i) = \langle \widehat{\mu}_i, f_y \rangle_{apx}$
return $\langle f_x, \widehat{\Pi} f_y \rangle_{apx}$ := SpectralDotProduct$(G, x, y, \delta, \xi', \mathcal{D}) - h_x^T X^{-1} h_y$
```

The following Lemma is a generalization of Lemma 3.14 to the approximation of the cluster means (i.e, $\widehat{\mu}_1, \dots, \widehat{\mu}_k$), where $\widehat{\mu}_i \in \mathbb{R}^k$ is a vector that approximates the center of cluster C_i (i.e., μ_i) such that $||\widehat{\mu}_i - \mu_i||_2$ is small.

Lemma 3.30. *Let $k \ge 2$ be an integer, $\varphi \in (0,1)$, and $\epsilon \in (0,1)$. Let $G = (V, E)$ be a d-regular graph that admits a (k, φ, ϵ)-clustering $C_1, \dots, C_k$. Let $\mu_1, \dots, \mu_k$ denote the cluster means of $C_1, \dots, C_k$. Let $0 < \zeta < \frac{\sqrt{\epsilon}}{20 \cdot k \cdot \varphi}$. Let $\widehat{\mu}_1, \dots, \widehat{\mu}_k \in \mathbb{R}^k$ denote an approximation of the cluster means such that for each $i \in [k]$, $||\mu_i - \widehat{\mu}_i||_2 \le \zeta ||\mu_i||_2$. Let $S \subseteq \{\widehat{\mu}_1, \dots, \widehat{\mu}_k\}$. Let $|S| = r$ and $\widehat{H} \in \mathbb{R}^{k \times r}$ denote a matrix whose columns are the vectors in S. Let $\sigma : [r] \to [k]$ denote a mapping from the the columns of H to the corresponding cluster. Let $\widehat{W} \in \mathbb{R}^{r \times r}$ be a diagonal matrix such that $\widehat{W}(i,i) = \sqrt{|C_{\sigma(i)}|}$. Let $\widehat{Z} = \widehat{H}\widehat{W}$.Then for any vector $x \in \mathbb{R}^r$ with $||x||_2 = 1$ we have*

1. $|x^T(\widehat{Z}^T\widehat{Z} - I)x| \le \frac{5\sqrt{\epsilon}}{\varphi}$

2. $|x^T((\widehat{Z}^T\widehat{Z})^{-1} - I)x| \le \frac{5\sqrt{\epsilon}}{\varphi}$.

Proof. **Proof of item** (1) Let $Y \in \mathbb{R}^{k \times k}$ be a matrix, whose i-th column is equal to $\sqrt{C_i} \cdot \mu_i$. By Lemma 3.9 item (2) for any vector $\alpha \in \mathbb{R}^k$ with $||\alpha||_2 = 1$ we have

$$|\alpha^T(Y^TY - I)\alpha| \le \frac{4\sqrt{\epsilon}}{\varphi} \tag{3.116}$$

Let $\widehat{Y} \in \mathbb{R}^{k \times k}$ be a matrix, whose i-th column is equal to $\sqrt{C_i} \cdot \widehat{\mu}_i$. Note that for any $i, j \in [k]$ we have $(Y^TY)(i,j) = \sqrt{|C_i||C_j|}\langle \mu_i, \mu_j \rangle$ and $(\widehat{Y}^T\widehat{Y})(i,j) = \sqrt{|C_i||C_j|}\langle \widehat{\mu}_i, \widehat{\mu}_j \rangle$. Therefore for any

$i \in [k]$ we have

$$\begin{aligned}
\left|(Y^TY)(i,i) - (\widehat{Y}^T\widehat{Y})(i,i)\right| &= |C_i| \left| ||\mu_i||_2^2 - ||\widehat{\mu_i}||_2^2 \right| \\
&\leq |C_i| \cdot |(||\mu_i||_2 - ||\widehat{\mu_i}||_2)(||\mu_i||_2 + ||\widehat{\mu_i}||_2)| \\
&\leq |C_i| \cdot \left|(\zeta ||\mu_i||_2)(||\mu_i||_2 + (1+\zeta)||\mu_i||_2)\right| && \text{Since } ||\widehat{\mu_i}||_2 \leq (1+\zeta)||\mu_i||_2 \\
&\leq 3 \cdot \zeta |C_i| \cdot ||\mu_i||_2^2 && \text{Since } \zeta < 1 \\
&\leq 6 \cdot \zeta && \text{By Lemma 3.7 } ||\mu_i||_2^2 \leq \frac{2}{|C_i|}
\end{aligned}$$

Also for any $i \neq j \in [k]$ we have

$$\begin{aligned}
&\left|(Y^TY)(i,j) - (\widehat{Y}^T\widehat{Y})(i,j)\right| && (3.117) \\
&= \sqrt{|C_i||C_j|} \cdot \left|\langle \widehat{\mu}_i, \widehat{\mu}_j \rangle - \langle \mu_i, \mu_j \rangle\right| \\
&= \sqrt{|C_i||C_j|} \cdot \left|\langle \mu_i + (\widehat{\mu}_i - \mu_i), \mu_j + (\widehat{\mu}_j - \mu_j) \rangle - \langle \mu_i, \mu_j \rangle\right| \\
&\leq \sqrt{|C_i||C_j|} \cdot \left(|\langle \widehat{\mu}_i - \mu_i, \widehat{\mu}_j - \mu_j \rangle| + |\langle \widehat{\mu}_i - \mu_i, \mu_j \rangle| + |\langle \widehat{\mu}_j - \mu_j, \mu_i \rangle|\right) && \text{By triangle inequality} \\
&\leq \sqrt{|C_i||C_j|} \cdot \left(||\widehat{\mu}_i - \mu_i||_2 ||\widehat{\mu}_j - \mu_j||_2 + ||\widehat{\mu}_i - \mu_i||_2 ||\mu_j||_2 + ||\widehat{\mu}_j - \mu_j||_2 ||\mu_i||_2\right) && \text{By Cauchy-Schwarz} \\
&\leq \sqrt{|C_i||C_j|} \cdot (\zeta^2 + 2\zeta)\left(||\mu_i||_2 ||\mu_j||_2\right) && \text{Since } ||\widehat{\mu_i} - \mu_i||_2 \leq \zeta ||\mu_i||_2 \text{ for all } i \\
&\leq \sqrt{|C_i||C_j|} \cdot 6 \cdot \zeta \cdot \frac{1}{\sqrt{|C_i||C_j|}} && \text{By Lemma 3.7 } ||\mu_i||_2^2 \leq \frac{2}{|C_i|} \text{ for all} \\
&\leq 6 \cdot \zeta && (3.118)
\end{aligned}$$

Therefore we have

$$\begin{aligned}
||(Y^TY) - (\widehat{Y}^T\widehat{Y})||_2 &\leq ||(Y^TY) - (\widehat{Y}^T\widehat{Y})||_F \\
&\leq \sqrt{\sum_{i=1}^{k} \sum_{j=1}^{k} \left((Y^TY)(i,j) - (\widehat{Y}^T\widehat{Y})(i,j)\right)^2} \\
&\leq 6 \cdot k \cdot \zeta \\
&\leq \frac{\sqrt{\epsilon}}{2\varphi} && \text{Since } \zeta \leq \frac{\sqrt{\epsilon}}{20 \cdot k \cdot \varphi}
\end{aligned}$$

Thus for any $\alpha \in \mathbb{R}^k$ with $||\alpha||_2 = 1$ we have

$$\left|\alpha^T \left((Y^TY) - (\widehat{Y}^T\widehat{Y})\right)\alpha\right| \leq \frac{\sqrt{\epsilon}}{2\varphi} \tag{3.119}$$

Putting (3.119) and (3.116) together we get

$$\left|\alpha^T \left(\widehat{Y}^T\widehat{Y} - I\right)\alpha\right| \leq 4.5 \frac{\sqrt{\epsilon}}{\varphi}$$

Let $x \in \mathbb{R}^r$ be a vector with $||x||_2 = 1$, and let $\alpha \in \mathbb{R}^k$ be a vector that is $x_j = \alpha_j$ if $\widehat{\mu}_j \in S$ and otherwise $x_j = 0$. Thus we have $||\alpha||_2 = ||x||_2 = 1$ and $\widehat{Y}z = \widehat{Z}x$. Hence, we get

$$|x^T(\widehat{Z}^T\widehat{Z} - I)x| = |\alpha^T(\widehat{Y}^T\widehat{Y} - I)\alpha| \le \frac{4.5\sqrt{\epsilon}}{\varphi}$$

Proof of item (2) For any vector $x \in \mathbb{R}^r$ with $||x||_2 = 1$ we have

$$1 - \frac{4.5\sqrt{\epsilon}}{\varphi} \le x^T(\widehat{Z}^T\widehat{Z})x \le 1 + \frac{4.5\sqrt{\epsilon}}{\varphi} \tag{3.120}$$

Note that $\widehat{Z}^T\widehat{Z}$ is symmetric and positive semidefinit. Also note that $\widehat{Z}^T\widehat{Z}$ is spectrally close to I, hence, $\widehat{Z}^T\widehat{Z}$ is invertible. Thus by (3.120) and Lemma 3.13 for any vector $x \in \mathbb{R}^r$ we have

$$1 - \frac{5\sqrt{\epsilon}}{\varphi} \le x^T(\widehat{Z}^T\widehat{Z})^{-1}x \le 1 + \frac{5\sqrt{\epsilon}}{\varphi}$$

Therefore we get

$$|x^T((\widehat{Z}^T\widehat{Z})^{-1} - I)x| \le \frac{5\sqrt{\epsilon}}{\varphi}.$$

□

Theorem 3.6. *Let $G = (V,E)$ be a d-regular graph that admits a (k,φ,ϵ)-clustering $C_1,\ldots,C_k$. Let $k \ge 2$ be an integer, $\varphi \in (0,1)$, $\frac{1}{n^5} < \xi < 1$, and $\frac{\epsilon}{\varphi^2}$ be smaller than a positive absolute constant. Then there exists an event $\mathcal{E}$ such that $\mathcal{E}$ happens with probability $1 - n^{-48}$ and conditioned on $\mathcal{E}$ the following holds.*

Let $r \in [k]$. Let $\delta \in (0,1)$. Let $B_1,\ldots,B_r$ denote multisets of points. Let $b = \max_{i\in r}|B_i|$. Let $\sigma : [r] \to [k]$ denote a mapping from the set B to the cluster $C = \sigma(B)$. Suppose that for all $i \in [r]$, $B_i \subseteq \sigma(B_i)$ and for all $i \ne j \in [r]$, $\sigma(B_i) \ne \sigma(B_j)$. Let $\widehat{\mu}_i = \frac{1}{|B_i|}\cdot\sum_{z\in B_i} f_z$. Suppose that for each $i \in [r]$, $||\widehat{\mu}_i - \mu_{\sigma(i)}||_2 \le \frac{\sqrt{\epsilon}}{20\cdot k\cdot\varphi}||\mu_i||_2$. Let $\widehat{\Pi}$ is defined as a orthogonal projection onto then $span(\{\widehat{\mu}_1,\ldots,\widehat{\mu}_r\})^\perp$. Then for all $x,y \in V$ we have

$$\left|\left\langle f_x, \widehat{\Pi} f_y\right\rangle_{apx} - \left\langle f_x, \widehat{\Pi} f_y\right\rangle\right| \le \frac{\xi}{n},$$

where $\left\langle f_x, \widehat{\Pi} f_y\right\rangle_{apx} :=$ DotProductOracleOnSubspace$(G,x,y,\delta,\xi,\mathcal{D},B_1,\ldots,B_r)$. Algorithm 6 runs in time $b^2\cdot(\frac{k}{\xi})^{O(1)}\cdot n^{\delta+O(\epsilon/\varphi^2)}\cdot\frac{(\log n)^2}{\varphi^2}$.

Proof. **Runtime:** Note that Algorithm 6, first computes matrix $X \in \mathbb{R}^{r\times r}$, and vectors $h_x, h_y \in \mathbb{R}^k$. To compute $X(i,j)$ for any $i,j \in [r]$, as per line 4 of Algorithm 6, we run SpectralDotProduct$(G,z_i,z_j,\delta,\xi',\mathcal{D})$ for all $z_i \in B_i$ and $z_j \in B_j$, where $|B_i| \le b$ and $|B_j| \le b$.

Note that by Theorem 3.2, Algorithm SpectralDotProduct$(G,z_i,z_j,\delta,\xi',\mathcal{D})$ runs in time $(\frac{k}{\xi'})^{O(1)}\cdot n^{\delta+O(\epsilon/\varphi^2)}\cdot\frac{(\log n)^2}{\varphi^2}$. Thus one can compute the matrix X^{-1} in time $O(k^3 + k^2\cdot b^2\cdot$

$(\frac{k}{\xi'})^{O(1)} n^{\delta+O(\epsilon/\varphi^2)} \cdot \frac{(\log n)^2}{\varphi^2})$. Also, to compute $h_x(i)$ (respectively, $h_y(i)$) for any $i \in [r]$, as per line 7 and line 8 of Algorithm 6, we run SPECTRALDOTPRODUCT$(G, x, z, \delta, \xi', \mathscr{D})$ for all $z \in B_i$ (respectively, $z \in B_j$). Thus one can compute h_x and h_y in time $k \cdot b \cdot (\frac{k}{\xi'})^{O(1)} \cdot n^{\delta+O(\epsilon/\varphi^2)} \cdot \frac{(\log n)^2}{\varphi^2}$. As per line (2) of Algorithm 6 we set $\xi' := \Theta(\xi \cdot n^{(-80\epsilon/\varphi^2)} \cdot k^{-6})$. Therefore the runtime of the algoritm is $b^2 \cdot (\frac{k}{\xi})^{O(1)} \cdot n^{\delta+O(\epsilon/\varphi^2)} \cdot \frac{(\log n)^2}{\varphi^2}$.

Correctness: Let $x, y \in V$. Let $H \in \mathbb{R}^{k\times r}$ be a matrix whose columns are $\widehat{\mu}_1, \ldots, \widehat{\mu}_r$. Then we have $H\left(H^T H\right)^{-1} H^T$ is the orthogonal projection matrix onto $span(\{\widehat{\mu}_1, \ldots, \widehat{\mu}_r\})$. Let $W \in \mathbb{R}^{r\times r}$ denote a matrix such that for any $i \in [r]$, $W(i,i) = \sqrt{|C_{\sigma(i)}|}$. Note that

$$(HW)\left((HW)^T(HW)\right)^{-1}(HW)^T = HW\left(W^{-1}\left(H^T H\right)^{-1} W^{-1}\right) W H^T = H\left(H^T H\right)^{-1} H^T$$

Thus we have $(HW)\left((HW)^T(HW)\right)^{-1}(HW)^T$ is the orthogonal projection matrix onto $span(\{\widehat{\mu}_1, \ldots, \widehat{\mu}_r\})$ and we get

$$\widehat{\Pi} = I - HW\left(WH^T HW\right)^{-1} WH^T$$

Therefore, we have

$$\langle f_x, \widehat{\Pi} f_y \rangle = \langle f_x, f_y \rangle - f_x^T HW\left(WH^T HW\right)^{-1} WH^T f_y \tag{3.121}$$

Let $\langle f_x, f_y \rangle_{apx} :=$ SPECTRALDOTPRODUCT$(G, x, y, \delta, \xi', \mathscr{D})$. Then as per line 9 of Algorithm 6 we have

$$\langle f_x, \widehat{\Pi} f_y \rangle_{apx} := \langle f_x, f_y \rangle_{apx} - h_x^T X^{-1} h_y, \tag{3.122}$$

where as per line (4) of Algorithm 6 for any $i, j \in [r]$ we have $X(i,j) = \langle \widehat{\mu}_i, \widehat{\mu}_j \rangle_{apx}$, and as per line (7) and line (8) of Algorithm 6 for any $i \in [r]$ we have $h_x(i) = \langle \widehat{\mu}_i, f_x \rangle_{apx}$ and $h_y(i) = \langle \widehat{\mu}_i, f_y \rangle_{apx}$. Note that

$$h_x^T X^{-1} h_y = h_x^T W W^{-1} X^{-1} W^{-1} W h_y = h_x^T W (WXW)^{-1} W h_y$$

Therefore by (3.122), (3.121) and triangle inequality we have

$$\left|\langle f_x, \widehat{\Pi} f_y \rangle_{apx} - \langle f_x, \widehat{\Pi} f_y \rangle\right| \le |\langle f_x, f_y \rangle_{apx} - \langle f_x, f_y \rangle| + \left|h_x^T W (WXW)^{-1} W h_y - f_x^T HW\left(WH^T HW\right)^{-1} WH^T f_y\right|$$

Note that by Theorem 3.2 and by union bound over all pair of vertices with probability at least $1 - n^{-100} \cdot n^2$ for all $a, b \in V$ we have

$$|\langle f_a, f_b \rangle_{apx} - \langle f_a, f_b \rangle| \le \frac{\xi'}{n} \tag{3.123}$$

We define

$$\mathbf{a}_x = f_x^T HW, \quad A = (WH^T HW)^{-1}, \quad \mathbf{a}_y = WH^T f_y, \text{ and}$$

$$\mathbf{e}_x = h_x^T W - \mathbf{a}_x, \quad E = (WXW)^{-1} - A, \quad \mathbf{e}_y = W h_y - \mathbf{a}_y$$

Thus by triangle inequality we have

$$\begin{aligned}&\left|h_x^T W(WXW)^{-1}Wh_y - f_x^T HW(WH^T HW)^{-1}WH^T f_y\right| = \\ &\|\,(\mathbf{a}_x+\mathbf{e}_x)\,(A+E)\left(\mathbf{a}_y+\mathbf{e}_y\right)-\mathbf{a}_x A\mathbf{a}_y\|_2 \le \\ &\|\mathbf{e}_x\|_2\|A\|_2\|\mathbf{a}_y\|_2+\|\mathbf{a}_x\|_2\|E\|_2\|\mathbf{a}_y\|_2+\|\mathbf{a}_x\|_2\|A\|_2\|\mathbf{e}_y\|_2+ \\ &\|\mathbf{e}_x\|_2\|E\|_2\|\mathbf{a}_y\|_2+\|\mathbf{a}_x\|_2\|E\|_2\|\mathbf{e}_y\|_2+\|\mathbf{e}_x\|_2\|A\|_2\|\mathbf{e}_y\|_2+\|\mathbf{e}_x\|_2\|E\|_2\|\mathbf{e}_y\|_2\end{aligned} \tag{3.124}$$

Thus we need to bound $||\mathbf{a}_x||_2, ||\mathbf{a}_y||_2, ||\mathbf{e}_x||_2, ||\mathbf{e}_y||_2, ||A||_2, ||E||_2$. Note that $||\mathbf{a}_x||_2 = ||f_x^T HW||_2$, Thus we have $||\mathbf{a}_x||_2 \le ||f_x^T H||_2||W||_2$. Note that

$$||W||_2 \le \max_i W(i,i) = \max_i \sqrt{|C_i|} \le \sqrt{n} \tag{3.125}$$

Then we bound $||f_x^T H||_2$. Note that $||f_x^T H||_2 = \sqrt{\sum_{i=1}^r \left\langle f_x, \widehat{\mu}_i\right\rangle^2}$. We first bound $\left\langle f_x, \widehat{\mu}_i\right\rangle$.

$$\begin{aligned}\left\langle f_x, \widehat{\mu}_i\right\rangle &= \frac{1}{|B_i|}\cdot\sum_{z\in B_i}\left\langle f_x, f_z\right\rangle \\ &\le \frac{1}{|B_i|}\sum_{z\in B_i}||f_x||_2||f_z||_2 \\ &\le \frac{1}{|B_i|}\cdot\sum_{z\in B_i}\sqrt{k^2\cdot||f_x||_\infty^2||f_z||_\infty^2} \\ &\le \frac{1}{|B_i|}\cdot|B_i|\cdot k\cdot O\left(\frac{k\cdot n^{40\epsilon/\varphi^2}}{n}\right) && \text{By Lemma 3.5 and since } \min_{i\in k}|C_i| \ge \Omega\left(\frac{n}{k}\right) \\ &\le O(k^2\cdot n^{-1+40\epsilon/\varphi^2})\end{aligned}$$

Since, $r < k$, we get

$$||f_x^T H||_2 = \sqrt{\sum_{i=1}^r \left\langle f_x, \widehat{\mu}_i\right\rangle^2} \le \sqrt{k}\cdot O(k^2\cdot n^{-1+40\epsilon/\varphi^2}) \le O(k^{2.5}\cdot n^{-1+40\epsilon/\varphi^2}) \tag{3.126}$$

Thus we get

$$||\mathbf{a}_x||_2 = ||f_x^T HW||_2 \le ||f_x^T H||_2||W||_2 \le O\left(k^{2.5}\cdot n^{-1/2+40\epsilon/\varphi^2}\right) \tag{3.127}$$

By the same computation we also have

$$||\mathbf{a}_y||_2 \le O\left(k^{2.5}\cdot n^{-1/2+40\epsilon/\varphi^2}\right) \tag{3.128}$$

Next we bound $||\mathbf{e}_x||_2$. We have $\mathbf{e}_x = h_x^T W - f_x^T HW$. Thus we get $||\mathbf{e}_x||_2 \le ||h_x^T - f_x^T H||_2||W||_2$. By (3.125) we have a bound on $||W||_2$. Note that for any $i \in r$, we have $h_x(i) = \frac{1}{|B_i|}\sum_{z\in B_i}\left\langle f_x, f_z\right\rangle_{apx}$

and $(f_x^T H)(i) = \frac{1}{|B_i|}\sum_{z\in B_i}\langle f_x, f_z\rangle$. Therefore with probability at least $1-n^{-98}$ we have

$$\begin{aligned}
|h_x(i) - (f_x^T H)(i)| &= \left|\frac{1}{b}\sum_{z\in B_i}(\langle f_x, f_z\rangle_{apx} - \langle f_x, f_z\rangle)\right| \\
&\le \frac{1}{|B_i|}\sum_{z\in B_i}|\langle f_x, f_z\rangle_{apx} - \langle f_x, f_z\rangle| && \text{By triangle inequality} \\
&\le \frac{1}{|B_i|}\cdot|B_i|\cdot\frac{\xi'}{n} && \text{By (3.123)}
\end{aligned}$$

Since $r \le k$, we have

$$||h_x^T - f_x^T H||_2 = \sqrt{\sum_{i=1}^{r}(h_x(i) - \mathbf{a}_x(i))^2} \le \sqrt{k}\cdot\frac{\xi'}{n}$$

Therefore by (3.125) we have

$$||\mathbf{e}_x||_2 \le ||h_x^T - f_x^T H||_2||W||_2 \le \frac{\xi'\sqrt{k}}{\sqrt{n}} \tag{3.129}$$

By the same computation we also have

$$||\mathbf{e}_y||_2 \le \frac{\xi'\sqrt{k}}{\sqrt{n}} \tag{3.130}$$

Next we bound $||A||_2$. Note that $A = ((HW)^T(HW))^{-1}$. By Lemma 3.30 item (2) for any vector $x\in\mathbb{R}^r$ with $||x||_2 = 1$ we have

$$\left|x^T\left(\left((HW)^T(HW)\right)^{-1} - I\right)x\right| \le \frac{5\sqrt{\epsilon}}{\varphi}$$

Therefore

$$||A||_2 = ||((HW)^T(HW))^{-1}||_2 \le 1 + \frac{5\sqrt{\epsilon}}{\varphi} \le 2 \tag{3.131}$$

Now we bound $||E||_2 = ||(WXW)^{-1} - (WH^THW)^{-1}||_2$. For any $i,j\in[r]$ we have

$$(WXW)(i,j) = \sqrt{|C_{\sigma(B_i)}||C_{\sigma(B_j)}|}\cdot\frac{1}{|B_i|\cdot|B_j|}\cdot\sum_{z_i\in B_i, z_j\in B_j}\langle f_{z_i}, f_{z_j}\rangle_{apx}$$

and

$$(WH^THW)(i,j) = \sqrt{|C_{\sigma(B_i)}||C_{\sigma(B_j)}|}\cdot\frac{1}{|B_i|\cdot|B_j|}\cdot\sum_{z_i\in B_i, z_j\in B_j}\langle f_{z_i}, f_{z_j}\rangle$$

Therefore with probability at least $1 - n^{-98}$ we have

$$\begin{aligned}
&|(WXW)(i,j) - (WH^T HW)(i,j)| \\
&= \left| \sqrt{|C_{\sigma(B_i)}||C_{\sigma(B_j)}|} \cdot \frac{1}{|B_i| \cdot |B_j|} \sum_{z_i \in B_i, z_j \in B_j} (\hat{f}_{z_i z_j} - \langle f_{z_i}, f_{z_j} \rangle) \right| \\
&\le \sqrt{|C_{\sigma(B_i)}||C_{\sigma(B_j)}|} \cdot \frac{1}{|B_i| \cdot |B_j|} \sum_{z_i \in B_i, z_j \in B_j} |\hat{f}_{z_i z_j} - \langle f_{z_i}, f_{z_j} \rangle | && \text{By triangle inequality} \\
&\le n \cdot \frac{1}{|B_i| \cdot |B_j|} \cdot |B_i| \cdot |B_j| \cdot \frac{\xi'}{n} && \text{By (3.123) and since } |C| \le n
\end{aligned} \tag{3.132}$$

Since $r \le k$ and by (3.132) we get

$$\begin{aligned}
\big| ||WXW - WH^T HW||_2 \big| &\le ||WXW - WH^T HW||_F \\
&\le \sqrt{\sum_{i=1}^{r} \sum_{j=1}^{r} \big((WXW)(i,j) - (WH^T HW)(i,j) \big)^2} \\
&\le k \cdot \xi'
\end{aligned}$$

Thus for any vector $x \in \mathbb{R}^r$ with $||x||_2 = 1$ we have

$$x^T (WH^T HW)x - k \cdot \xi' \le x^T (WXW)x \le x^T (WH^T HW)x + k \cdot \xi' \tag{3.133}$$

By Lemma 3.30 item (1) for any vector $x \in \mathbb{R}^r$ with $||x||_2 = 1$ we have

$$|x^T \left((HW)^T (HW) - I \right) x| \le \frac{5\sqrt{\epsilon}}{\varphi}$$

Hence we have

$$x^T (HW)^T (HW)x \ge 1 - \frac{5\sqrt{\epsilon}}{\varphi} \ge \frac{1}{2} \tag{3.134}$$

Therfore by (3.133) and (3.134) we get for any vector $x \in \mathbb{R}^r$ with $||x||_2 = 1$ we have

$$(1 - 2 \cdot k \cdot \xi') \cdot x^T (WH^T HW)x \le x^T (WXW)x \le (1 + 2 \cdot k \cdot \xi') \cdot x^T (WH^T HW)x \tag{3.135}$$

Note that $WH^T HW$ is a symmetric matrix. Also note that by definition of X in line 4 of Algorithm 6, X is a symmetric matrix, hence, WXW is symmetric and positive semidefinit. Also note that WXW is spectrally close to $WH^T HW$ and I, hence, WXW is invertible. Thus by (3.135) and Lemma 3.13 we have

$$(1 - 4 \cdot k \cdot \xi') \cdot x^T (WH^T HW)^{-1} x \le x^T (WXW)^{-1} x \le (1 + 4 \cdot k \cdot \xi') \cdot x^T (WH^T HW)^{-1} x$$

Therefore by (3.131) we have

$$||E||_2 = ||(WH^THW)^{-1} - (WXW)^{-1}||_2 \leq 4\cdot k\cdot\xi'\cdot||(WH^THW)^{-1}||_2 = 8\cdot k\cdot\xi' \tag{3.136}$$

Putting (3.136), (3.131), (3.129), (3.130), (3.127), (3.128) and(3.124) together, with probability at least $1-n^{-50}$ we have

$$\begin{aligned}
&\left|h_x^TW(WXW)^{-1}Wh_y - f_x^THW(WH^THW)^{-1}WH^Tf_y\right| = \\
&\|\left(\mathbf{a}_x+\mathbf{e}_x\right)(A+E)\left(\mathbf{a}_y+\mathbf{e}_y\right) - \mathbf{a}_xA\mathbf{a}_y\|_2 \leq \\
&\|\mathbf{e}_x\|_2\|A\|_2\|\mathbf{a}_y\|_2 + \|\mathbf{a}_x\|_2\|E\|_2\|\mathbf{a}_y\|_2 + \|\mathbf{a}_x\|_2\|A\|_2\|\mathbf{e}_y\|_2 + \\
&\|\mathbf{e}_x\|_2\|E\|_2\|\mathbf{a}_y\|_2 + \|\mathbf{a}_x\|_2\|E\|_2\|\mathbf{e}_y\|_2 + \|\mathbf{e}_x\|_2\|A\|_2\|\mathbf{e}_y\|_2 + \|\mathbf{e}_x\|_2\|E\|_2\|\mathbf{e}_y\|_2 \\
&\leq O\left(\xi'\cdot\frac{\sqrt{k}}{\sqrt{n}}\cdot k^{2.5}\cdot n^{-1/2+40\epsilon/\varphi^2}\right) + O\left(k\cdot\xi'\cdot k^5\cdot n^{-1+80\epsilon/\varphi^2}\right) \\
&+ O\left(\xi'\cdot\frac{\sqrt{k}}{\sqrt{n}}\cdot k\cdot\xi'\cdot k^{2.5}\cdot n^{-1/2+40\epsilon/\varphi^2}\right) + O\left(\xi'^2\cdot\frac{k}{n}\right) + O\left(\xi'^2\cdot\frac{k}{n}\cdot k\cdot\xi'\right) \\
&\leq O\left(\frac{\xi'\cdot k^6\cdot n^{80\epsilon/\varphi^2}}{n}\right) \\
&\leq \frac{1}{2}\cdot\frac{\xi}{n}
\end{aligned} \tag{3.137}$$

The last inequality holds by setting $\xi' = \frac{\xi\cdot n^{(-80\epsilon/\varphi^2)}\cdot k^{-6}}{c}$ as per line of Algorithm 6 where c is a large enough constant to cancel the constant hidden in $O\left(\frac{\xi'\cdot k^6\cdot n^{80\epsilon/\varphi^2}}{n}\right)$.

Therefore with probability at least $1-n^{-98} \geq 1-n^{-50}$ we have

$$\begin{aligned}
&\left|\left\langle f_x, \widehat{\Pi}f_y\right\rangle_{apx} - \left\langle f_x, \widehat{\Pi}f_y\right\rangle\right| \\
&\leq |\widehat{f}_{xy} - \langle f_x, f_y\rangle| + \left|h_x^TW(WXW)^{-1}Wh_y - f_x^THW\left(WH^THW\right)^{-1}WH^Tf_y\right| \\
&\leq \frac{\xi'}{n} + \frac{1}{2}\cdot\frac{\xi}{n} \\
&\leq \frac{\xi}{n} && \text{By (3.123), (3.137), and since } \xi' <
\end{aligned} \tag{3.138}$$

Now let $\mathscr{E}$ be the event that for all $x,y\in V$ we have $|\left\langle f_x, \widehat{\Pi}f_y\right\rangle_{apx} - \left\langle f_x, \widehat{\Pi}f_y\right\rangle| \leq \frac{\xi}{n}$. Then by (3.138) and the union bound we get that $\mathscr{E}$ happens with probability at least $1-n^{-48}$ and it is the claimed high probability event from the statement.

□

Corollary 3.1. *Let $G=(V,E)$ be a d-regular graph that admits a (k,φ,ϵ)-clustering $C_1,\ldots,C_k$. Let $k\geq 2$ be an integer, $\varphi\in(0,1)$, $\delta\in(0,1)$, $\frac{1}{n^5}<\xi<1$, $\frac{\epsilon}{\varphi^2}$ be smaller than a positive absolute* *constant. Let $\mathcal{E}$ be the event that happens with probability $1-n^{-48}$ that is guaranteed by Theorem 3.6. Then conditioned on $\mathcal{E}$ the following conditions hold.*

Let $r \in [k]$. Let $B_1,\dots,B_r,B'$ denote multisets of points. Let $b = \max\{|B_1|,\dots,|B_r|,|B'|\}$. Let $\sigma : [r] \to [k]$ denote a mapping from the set B to the cluster $C = \sigma(B)$. Suppose that for all $i \in [r]$, $B_i \subseteq \sigma(B_i)$ and for all $i \neq j \in [r]$, $\sigma(B_i) \neq \sigma(B_j)$. Let $\widehat{\mu}_i = \frac{1}{|B_i|}\cdot\sum_{z\in B} f_z$ for all $i \in [r]$, and let $\widehat{\mu} = \frac{1}{|B'|}\cdot\sum_{z\in B_i} f_z$. Suppose that for each $i \in [r]$, $||\widehat{\mu}_i - \mu_{\sigma(i)}||_2 \le \frac{\sqrt{\epsilon}}{20\cdot k\cdot\varphi}||\mu_i||_2$. Let $\widehat{\Pi}$ is defined as a orthogonal projection onto then $span(\{\widehat{\mu}_1,\dots,\widehat{\mu}_r\})^{\perp}$. Then the following hold:

1. *There exits an algorithm that runs in time $b^3\cdot(\frac{k}{\xi})^{O(1)}\cdot n^{\delta+O(\epsilon/\varphi^2)}\cdot\frac{(\log n)^2}{\varphi^2}$ and for any $x \in V$ returns a value $\left\langle f_x,\widehat{\Pi}\widehat{\mu}\right\rangle_{apx}$ such that*
$$\left|\left\langle f_x,\widehat{\Pi}\widehat{\mu}\right\rangle_{apx} - \left\langle f_x,\widehat{\Pi}\widehat{\mu}\right\rangle\right| \le \frac{\xi}{n}.$$

2. *There exits an algorithm that runs in time $b^4\cdot(\frac{k}{\xi})^{O(1)}\cdot n^{\delta+O(\epsilon/\varphi^2)}\cdot\frac{(\log n)^2}{\varphi^2}$ and returns a value $\left\|\widehat{\Pi}\widehat{\mu}\right\|^2_{apx}$ such that $\left|\left\|\widehat{\Pi}\widehat{\mu}\right\|^2_{apx} - ||\widehat{\Pi}\widehat{\mu}||_2^2\right| \le \frac{\xi}{n}$.*

Proof. **Proof of 1:** To compute $\left\langle f_x,\widehat{\Pi}\widehat{\mu}\right\rangle_{apx}$ we call Algorithm 6, b times in the following way:

$$\left\langle f_x,\widehat{\Pi}\widehat{\mu}\right\rangle_{apx} := \frac{1}{|B|}\cdot\sum_{y\in B}\textsc{DotProductOracleOnSubspace}(G,x,y,\delta,\mathcal{D},\xi,B_1,\dots,B_r) \tag{3.139}$$

The runtime of Algorithm 6 is $b^2\cdot(\frac{k}{\xi})^{O(1)}\cdot n^{\delta+O(\epsilon/\varphi^2)}\cdot(\log n)^2\cdot\frac{1}{\varphi^2}$, thus the runtime of computation of $\left\langle f_x,\widehat{\Pi}\widehat{\mu}\right\rangle_{apx}$ is $b^3\cdot(\frac{k}{\xi})^{O(1)}\cdot n^{\delta+O(\epsilon/\varphi^2)}\cdot(\log n)^2\cdot\frac{1}{\varphi^2}$. Moreover by Theorem 3.6 and the assumption that $\mathcal{E}$ holds we have

$$\begin{aligned}
\left|\left\langle f_x,\widehat{\Pi}\widehat{\mu}\right\rangle_{apx} - \left\langle f_x,\widehat{\Pi}\widehat{\mu}\right\rangle\right| &= \left|\frac{1}{|B'|}\sum_{y\in B'}\left\langle f_x,\widehat{\Pi}y\right\rangle_{apx} - \left\langle f_x,\widehat{\Pi}\widehat{\mu}\right\rangle\right| \\
&\le \frac{1}{|B'|}\sum_{y\in B'}\left|\left\langle f_x,\widehat{\Pi}y\right\rangle_{apx} - \left\langle f_x,\widehat{\Pi}\widehat{\mu}\right\rangle\right| && \text{By triangle inequality} \\
&\le \frac{1}{|B'|}\cdot|B'|\cdot\frac{\xi}{n} && \text{By Theorem 3.6} \\
&\le \frac{\xi}{n}
\end{aligned}$$

Proof of 2: To compute $\left\|\widehat{\Pi}\widehat{\mu}\right\|^2_{apx}$ we call the procedure from item (1) b times in the following way:

$$\left\|\widehat{\Pi}\widehat{\mu}\right\|^2_{apx} := \frac{1}{|B|}\cdot\sum_{x\in B}\left\langle f_x,\widehat{\Pi}\widehat{\mu}\right\rangle_{apx}. \tag{3.140}$$

The runtime of the procedure from item (1) is $b^3\cdot(\frac{k}{\xi})^{O(1)}\cdot n^{\delta+O(\epsilon/\varphi^2)}\cdot(\log n)^2\cdot\frac{1}{\varphi^2}$, thus the runtime of computation of $\langle f_x, \widehat{\Pi}\widehat{\mu}\rangle_{apx}$ is $b^4 \cdot (\frac{k}{\xi})^{O(1)} \cdot n^{\delta+O(\epsilon/\varphi^2)} \cdot (\log n)^2 \cdot \frac{1}{\varphi^2}$. Moreover by item (1) we have

$$\begin{aligned}
\left|\|\widehat{\Pi}\widehat{\mu}\|_{apx}^2 - ||\widehat{\Pi}\widehat{\mu}||_2^2\right| &= \left|\langle\widehat{\mu}, \widehat{\Pi}\widehat{\mu}\rangle_{apx} - \langle\widehat{\mu}, \widehat{\Pi}\widehat{\mu}\rangle\right| \\
&= \left|\frac{1}{|B'|} \cdot \sum_{x\in B'} \langle f_x, \widehat{\Pi}\widehat{\mu}\rangle_{apx} - \frac{1}{|B'|} \cdot \sum_{x\in B'} \langle f_x, \widehat{\Pi}\widehat{\mu}\rangle\right| \\
&\leq \frac{1}{|B'|} \cdot \sum_{x\in B'} \left|\langle f_x, \widehat{\Pi}\widehat{\mu}\rangle_{apx} - \sum_{x\in B'} \langle f_x, \widehat{\Pi}\widehat{\mu}\rangle\right| && \text{By triangle inequality} \\
&\leq \frac{1}{|B'|} \cdot |B'| \cdot \frac{\xi}{n} && \text{By item (1)} \\
&\leq \frac{\xi}{n}.
\end{aligned}$$

□

3.6 The main algorithm and its analysis

In this section we show that, by having access to approximate spectral dot-products for a (k,φ,ϵ)-clusterable graph G, we can assign each vertex in G to a cluster in sublinear time so that the resulting collection of clusters is, with high probability, a good approximation of a (k,φ,ϵ)-clustering of G. In particular, we can show that the fraction of wrong assignments per cluster is at most $C\cdot\frac{\epsilon}{\varphi^3}\cdot\log(k)$, for some constant $C>0$. In the next subsection we describe our algorithm then in the remaining part of the section we present its analysis.

3.6.1 The Algorithm (Partitioning Scheme, Algorithm 7)

We first present an idealized version of the sublinear clustering scheme defined by Algorithm 7 and Algorithm 10. In this section to simplify presentation we assume φ to be constant.

The algorithm can be thought of as consisting of 3 parts. The first part, described in paragraph **Idealized Clustering Algorithm**, is a procedure that explicitly, in iterative fashion, produces a k-clustering of G. More precisely it recovers clusters in $O(\log(k))$ stages, where for every i after the i-th stage at most $k/2^i$ clusters are left unrecovered. The algorithm can be thought of as a version of carving of halfspaces in $\mathbb{R}^k$ and it relies on the knowledge of cluster means $\mu_1,\ldots,\mu_k$ (recall that $\mu_i=\frac{1}{|C_i|}\sum_{x\in C_i}f_x$). That is why in paragraph **Finding approximate centers** we show how to compute approximations of μ_i's. To find good approximation to μ_i's we need to test many candidate sets $\{\widehat{\mu}_1,\ldots,\widehat{\mu}_k\}$, which also means considering many candidate clusterings. This is a problem as we want our procedure to run in sublinear time but the idealized partitioning algorithm constructs clusterings explicitly! To solve this we explain in paragraph **Verifying a clustering** how to emulate the partitioning algorithm to test that, for a set of $\{\widehat{\mu}_1,\ldots,\widehat{\mu}_k\}$, it indeed induces a good clustering.

Idealized Clustering Algorithm. Assume that the we have access to cluster means $\{\mu_1,\ldots,\mu_k\}$ and dot product evaluations. The algorithm proceeds in $O(\log(k))$ stages, in the first stage it considers k candidate sets $\widehat{C}_i$, where $x\in\widehat{C}_i$ iff f_x has big correlation with μ_i but small correlation with all other μ_j's. More precisely $x\in\widehat{C}_i$ iff:

$$\langle f_x,\mu_i\rangle\geq 0.93||\mu_i||^2 \text{ and for all } j\neq i\,\langle f_x,\mu_j\rangle<0.93||\mu_j||^2.$$

Note that by definition all these clusters are disjoint. Moreover we are able to show (see Lemma 3.37) that at least $k/2$ out of $\widehat{C}_i$'s are good approximate clusters, that is for each one of them there exists j such that $|\widehat{C}_i\triangle C_j|\leq O(\epsilon)\cdot|C_j|$. At this point we return these good clusters, remove the corresponding vertices from the graph, remove the corresponding μ's from the set $\{\mu_1,\ldots,\mu_k\}$ of still alive centers and proceed to the next stage.

In the next stage we restrict our attention to a lower dimensional subspace Π of $\mathbb{R}^k$. Intuitively we want to project out all the directions corresponding to the removed cluster centers. Recall that μ_i's are close to being orthogonal (see Lemma 3.12 and 3.7) so projecting the returned directions out is almost equivalent to considering the subspace $\Pi := \text{span}(\{\mu_1, \dots, \mu_b\})$, where $\{\mu_1, \dots, \mu_b\}$ is the set of still alive μ's. Now the algorithm considers b candidate clusters where the condition for x being in a cluster i changes to:

$$\langle f_x, \Pi\mu_i \rangle \geq 0.93||\Pi\mu_i||^2 \text{ and for all } j \in [b], j \neq i \, \langle f_x, \Pi\mu_j \rangle < 0.93||\Pi\mu_j||^2.$$

We are still able to show (also Lemma 3.37) that at least $b/2$ out of them are good approximate clusters. That is for each i there exists j such that $|\widehat{C}_i \triangle C_j| \leq O(\epsilon) \cdot |C_j|$ but this time the constant hidden in the O notation is bigger than in the first stage. In general at any stage t the bound degrades to $O(\epsilon \cdot t)$. At the end of the stage we proceed in a similar fashion by returning the clusters, removing the corresponding vertices and μ's and considering a lower dimensional subspace of Π in the next stage.

The algorithm continues in such a fashion for $O(\log(k))$ steps, as we guarantee that in each stage at least half of the remaining cluster means is removed. Thus the final guarantee is: there exists a permutation π on k elements such that for every i:

$$|\widehat{C}_{\pi(i)} \triangle C_i| \leq O\big(\epsilon \log(k)\big) \cdot |C_i|.$$

The decreasing (in the inclusion sense) sequence of subspaces $(\Pi_1, \dots, \Pi_{\log(k)})$ corresponds to the subspaces constructed in Algorithm 7, while this offline algorithm as a whole corresponds to the sublinear Algorithm 10 that implicitly tries to construct a sequence of subspaces that (with respect to Algorithm 7) defines a good clustering.

Finding approximate centers. Note that cluster means are defined by the clustering, so it may seem that finding approximate means is a difficult operation. However, there is a relatively simple solution to this. In Algorithm 10 we find approximate cluster means by sampling $O(\frac{\varphi^2}{\epsilon} k^4 \log(k))$ points, guessing cluster memberships and considering the means of the samples as cluster centers. We use that the mean of a random sample of a cluster is typically close to the true mean of its cluster and so our sample means will provide a good estimation of the true means. We also remark that sampling a single vertex from each cluster does not seem to provide a sufficiently good estimate, i.e. we require to take the mean of a sample *set*.

Verifying a clustering. We also need a procedure that given an implicit sequence of subspaces $(\Pi_1, \dots, \Pi_{\log(k)})$ checks whether they indeed define (via Algorithm 7) a good clustering. In fact, for every guess of cluster centers and the corresponding (as implicitly created by Algorithm 10) sequence of Π's we need to be able to check efficiently if the resulting clustering is a good approximation of a (k, φ, ϵ)-clustering. Since we would like to do this in sublinear time as well, we need to do this verification by random sampling. Then we design a procedure that consists of two steps. In a first step, we check if the cluster sizes are not too small. This is only a technical step, which is needed to make sure that the later steps work. The main step is to test whether every cluster has small outer conductance (Algorithm 11). In order to do so, we sample vertices uniformly at random and check whether they are contained in the cluster that is currently checked. If this is the case, we sample a random edge incident to the sample vertex. This way, we obtain a random edge incident to a random vertex from the current cluster (this follows since the conditional distribution is uniform over the cluster). We use standard concentration bounds to prove that we get a good approximation.

In the partitioning scheme and in the analysis a useful definition are subsets of vertices called threshold sets. A threshold set of a point y is the set of vertices with dot products (or approximate dot product) with y being above a specific threshold, more formally:

Definition 3.8 (**Threshold sets**)**.** Let $G = (V, E)$ be a (k, φ, ϵ)-clusterable graph (*as in Definition 3.2*). Recall that $f_x = F\mathbb{1}_x$. For $y \in \mathbb{R}^k, \theta \in \mathbb{R}^+$ we define:

$$C_{y,\theta} := \{x \in V : \langle f_x, y \rangle \geq \theta ||y||^2\}$$

Definition 3.9 (**Approximate threshold sets**)**.** Let $G = (V, E)$ be a (k, φ, ϵ)-clusterable graph (*as in Definition 3.2*). Recall that $f_x = F\mathbb{1}_x$. For $\theta \in \mathbb{R}^+$ and $y \in \mathbb{R}^k$ such that $y = \widehat{\Pi}(\widehat{\mu})$, where $\widehat{\Pi}$ is the orthogonal projection onto $span(\{\widehat{\mu}_1, \ldots, \widehat{\mu}_b\})^\perp$ and each $\widehat{\mu}, \widehat{\mu}_1, \ldots, \widehat{\mu}_b$ is an average of a set of embedded vertices:

$$C^{apx}_{y,\theta} := \{x \in V : \langle f_x, y \rangle_{apx} \geq \theta \left\| y \right\|^2_{apx}\}. \tag{3.141}$$

Recall that a discussion of how $\langle \cdot, \cdot \rangle_{apx}$ and $\| \cdot \|_{apx}$ are computed is presented in Section 3.5.6.

Algorithm 7 HYPERPLANEPARTITIONING$(x, (T_1, T_2, \ldots, T_b))$
▷ T_i's are sets of $\widehat{\mu}_j$ where $\widehat{\mu}_j$'s are given as sets of points
▷ see Section 3.5.6 for the reason of such representation

```
for $i = 1$ to $b$ do
    Let $\Pi$ be the projection onto the span$(\bigcup_{j<i} T_j)^\perp$.
    Let $S_i = \bigcup_{j \geq i} T_j$
    for $\widehat{\mu} \in T_i$ do
        if $x \in C^{apx}_{\Pi\widehat{\mu},0.93} \setminus \bigcup_{\widehat{\mu}' \in S_i \setminus \{\widehat{\mu}\}} C^{apx}_{\Pi\widehat{\mu}',0.93}$ then      ▷ see (3.141) for definition of $C^{apx}_{y,\theta}$
            return $\widehat{\mu}$
```

HYPERPLANEPARTITIONING is the algorithm that, after preprocessing, is used to assign vertices to clusters. In the preprocessing step (see COMPUTEORDEREDPARTITION in Section 3.6.3) an ordered partition $(T_1, \ldots, T_b)$ of approximate cluster means $\{\widehat{\mu}_1, \ldots, \widehat{\mu}_k\}$ is computed. HYPERPLANEPARTITIONING invoked with this ordered partition as a parameter induces a collection of clusters as follows:

Definition 3.10 (**Implicit clustering**)**.** For an ordered partition $(T_1, \ldots, T_b)$ of approximate cluster means $\{\widehat{\mu}_1, \ldots, \widehat{\mu}_k\}$ we say that $(T_1, \ldots, T_b)$ **induces a collection of clusters** $\{\widehat{C}_{\widehat{\mu}_1}, \ldots, \widehat{C}_{\widehat{\mu}_k}\}$ if for all $i \in [k]$:

$$\widehat{C}_{\widehat{\mu}_i} = \left\{x \in V : \text{HYPERPLANEPARTITIONING}(x, (T_1, \ldots, T_b)) = \widehat{\mu}_i\right\}.$$

Remark 3.6. *Ordered partition $(T_1,\dots,T_b)$, precomputed in the preprocessing step (assuming access to $\{\mu_1,\dots,\mu_k\}$), will correspond to the **Idealized Clustering Algorithm** in the following sense. Number of sets in the partition (i.e. b) corresponds to the number of stages of **Idealized Clustering Algorithm** and for every $i \in [b]$ T_i contains exactly the μ's returned in stage i.*

In the rest of this section we explain how to compute an ordered partition $(T_1,\dots,T_b)$ of a set of approximate centers $(\widehat{\mu}_1,\widehat{\mu}_2,\dots,\widehat{\mu}_k)$ such that the induced clustering $\{\widehat{C}_{\widehat{\mu}_1},\dots,\widehat{C}_{\widehat{\mu}_k}\}$ satisfies that there exists a permutation π on k elements such that for all $i \in [k]$:

$$\left|\widehat{C}_{\widehat{\mu}_i} \triangle C_{\pi(i)}\right| \le O\left(\frac{\epsilon}{\varphi^3}\cdot\log(k)\right)|C_{\pi(i)}|.$$

We start, in Subsection 3.6.2, by studying geometric properties of our clustering instance. Recall, that we denote with μ_i the center of cluster C_i in the spectral embedding. We show that, for specific choices of θ, the threshold sets of μ_i have large intersection with the cluster C_i and small intersections with all other cluster C_j. This fact intuitively suggests that our partitioning algorithm works. Unfortunately, as discussed in the technical overview, this is not enough to prove a per cluster guarantee. For this reason in Subsection 3.6.3 we analyze the overlap structure of $\{C_{\mu_1,\theta},\dots,C_{\mu_k,\theta}\}$ more carefully and we give an algorithm (see COMPUTEORDEREDPARTITION) that given real centers $\{\mu_1,\dots,\mu_k\}$ and access to exact dot product evaluations computes an ordered partition of $\{\mu_1,\dots,\mu_k\}$ that induces a valid clustering. In Subsection 3.6.4 we present an algorithm that guesses the cluster memberships for a set of randomly selected nodes and, using those guesses, approximates cluster centers. Interestingly, we can show, in Subsection 3.6.4, that for the set of correct guesses the algorithm returns a good approximation of the cluster centers. Finally in Subsection 3.6.5 we show that we can find an ordered partition that induces a good clustering even if we have access only to approximate quantities. That is we show that even if we have access only to approximate means $\{\widehat{\mu}_1,\dots,\widehat{\mu}_k\}$ and the dot product evaluations are only approximately correct then we can find an ordered partition $(T_1,\dots,T_b)$ that induces a good collection of clusters. The last ingredient is to show that we are able to check if the clustering induced by a specific ordered partition is good. To solve this problem, we design an efficient and simple sampling algorithm which is also analyzed in Subsection 3.6.5.

3.6.2 Bounding intersections of $C_{\mu_i,\theta}$ with true clusters C_i

In this subsection we show that, for specific choices of θ, the threshold sets of μ_i (recall that μ_i's are cluster means in the spectral embedding) have large intersection with C_i and small intersections with other clusters. The main idea behind the proof is to use the bounds on dot product of cluster centers presented in Lemma 3.7. In particular, we use Lemma 3.6 to relate $\frac{\epsilon}{\varphi^2}$ with the directional variance of the spectral embedding in the direction of μ_i (i.e. $\sum_{x\in C_i}\langle f_x-\mu_i,\alpha\rangle^2$). Then we use the definition of threshold set to upper and lower bound $\langle f_x, \frac{\mu_i}{\|\mu_i\|}\rangle$ and Lemma 3.7 to upper and lower bound the dot product between cluster centers. By combining the bounds we obtain the following result:

Lemma 3.31. *Let $k \geq 2$, $\varphi \in (0,1)$ and $\frac{\epsilon}{\varphi^2}$ be smaller than a sufficiently small constant. Let $G = (V,E)$ be a d-regular graph that admits a (k,φ,ϵ)-clustering $\{C_1,\dots,C_k\}$. If μ_i's are cluster means then the following conditions hold. Let $S \subset \{\mu_1,\dots,\mu_k\}$. Let Π denote the orthogonal projection matrix on to the $span(S)^\perp$. Let $\mu \in \{\mu_1,\dots,\mu_k\} \setminus S$. Let C denote the cluster corresponding to the center μ. Let*

$$\widehat{C} := \{x \in V : \langle \Pi f_x, \Pi\mu\rangle \geq 0.96\|\Pi\mu\|_2^2\}$$

then we have:

$$\left|C \setminus \widehat{C}\right| \leq \frac{10^4\epsilon}{\varphi^2}|C|.$$

Proof. Let $x \in C \setminus \widehat{C}$. Then:

$$\begin{aligned}
\left|\left\langle \mu - f_x, \frac{\Pi\mu}{\|\Pi\mu\|_2}\right\rangle\right| &= \left|\left\langle \Pi(\mu - f_x), \frac{\Pi\mu}{\|\Pi\mu\|_2}\right\rangle\right| \\
&\geq 0.04 \cdot \|\Pi\mu\|_2 && \text{Since } \langle \Pi f_x, \Pi\mu\rangle < 0.96\|\Pi\mu\|_2^2 \\
&\geq 0.04 \cdot \left(1 - 24\frac{\sqrt{\epsilon}}{\varphi}\right)||\mu||_2 && \text{By Lemma 3.12} \\
&\geq 0.04 \cdot \left(1 - 40\frac{\sqrt{\epsilon}}{\varphi}\right)\sqrt{\frac{1}{|C|}} && \text{By Lemma 3.7} \\
&\geq 0.02 \cdot \sqrt{\frac{1}{|C|}} && \text{Since } \frac{\epsilon}{\varphi^2} \text{ is sufficiently small}
\end{aligned}$$

Then by Lemma 3.6 applied to direction $\alpha = \frac{\Pi\mu}{\|\Pi\mu\|_2}$ we have $\sum_{i=1}^k \sum_{x\in C_i} \langle f_x - \mu_i, \alpha\rangle^2 \leq \frac{4\epsilon}{\varphi^2}$. On the other hand

$$\frac{4\epsilon}{\varphi^2} \geq \sum_{i=1}^{k}\sum_{x\in C_i} \langle f_x - \mu_i, \alpha\rangle^2 \geq \sum_{x\in C\setminus\widehat{C}} \left\langle f_x - \mu, \frac{\Pi\mu}{\|\Pi\mu\|_2}\right\rangle^2 \geq 0.0004 \cdot \frac{|C \setminus \widehat{C}|}{|C|}.$$

Using the above we conclude with $|C \setminus \widehat{C}| \leq 10^4 \frac{\epsilon}{\varphi^2}|C|$. □

Remark 3.7. *Notice that the constants in Lemma 3.32 are different, they are equal* 0.96 *and* 0.9*. The reason is that the real tests for membership in Algorithm 7 are performed with constant* 0.93 *and the slacks are needed as we have access only to approximate dot products. See* (3.214) *for the formal reason.*

Lemma 3.32. *Let $k \geq 2$, $\varphi \in (0,1)$ and $\frac{\epsilon}{\varphi^2}$ be smaller than a sufficiently small constant. Let $G = (V,E)$ be a d-regular graph that admits a (k,φ,ϵ)-clustering $\{C_1,\dots,C_k\}$. If μ_i's are cluster means then the following conditions hold. Let $S \subset \{\mu_1,\dots,\mu_k\}$. Let Π denote the projection matrix on to $span(S)^\perp$. Let $\mu \in \{\mu_1,\dots,\mu_k\} \setminus S$. Let C denote the cluster corresponding to the center μ. Let*

$$\widehat{C} := \{x \in V : \langle \Pi f_x, \Pi\mu\rangle \geq 0.9\|\Pi\mu\|_2^2\}$$

then we have:

$$\left|\widehat{C}\cap(V\setminus C)\right|\leq 100\frac{\epsilon}{\varphi^2}|C|.$$

Proof. Let $x\in\widehat{C}\cap(V\setminus C)$. Then there exists cluster $C'\neq C$ such that $x\in C'$. Let μ' be the cluster mean of C'. Then:

$$\begin{aligned}\left|\left\langle f_x-\mu',\frac{\Pi\mu}{\|\Pi\mu\|_2}\right\rangle\right| &\geq \left|\left\langle \Pi f_x,\frac{\Pi\mu}{\|\Pi\mu\|_2}\right\rangle\right|-\left|\left\langle \Pi\mu',\frac{\Pi\mu}{\|\Pi\mu\|_2}\right\rangle\right| && \text{By triangle inequality}\\ &\geq 0.9\|\Pi\mu\|_2-\left|\left\langle \Pi\mu',\frac{\Pi\mu}{\|\Pi\mu\|_2}\right\rangle\right| && \text{As } x\in\widehat{C}\end{aligned}$$

Note that either $\mu'\in S$ and then $\Pi\mu'=0$ and in turn $|\langle\Pi\mu',\Pi\mu\rangle|=0$ or $\mu'\notin S$ and then $|\langle\Pi\mu',\Pi\mu\rangle|\leq\frac{60\sqrt{\epsilon}}{\varphi^2}\frac{1}{\sqrt{|C|\cdot|C'|}}$ by Lemma 3.12. Thus we have

$$\begin{aligned}\left|\left\langle f_x-\mu',\frac{\Pi\mu}{\|\Pi\mu\|_2}\right\rangle\right| &\geq 0.9\|\Pi\mu\|_2-\frac{60\sqrt{\epsilon}}{\varphi^2}\frac{1}{\sqrt{|C|\cdot|C'|}}\frac{1}{\|\Pi\mu\|_2}\\ &\geq 0.8\frac{1}{\sqrt{|C|}}-\frac{120\sqrt{\epsilon}}{\varphi^2}\frac{1}{\sqrt{|C|\cdot|C'|}}\cdot\sqrt{|C|} && \text{by Lemma 3.12 and Lemma 3.7, } \|\Pi\mu\|_2\geq\frac{1}{2\cdot\sqrt{|C|}}\\ &\geq 0.2\sqrt{\frac{1}{|C|}} && \text{Since } \frac{\epsilon}{\varphi^2} \text{ sufficiently small and } \frac{|C|}{|C'|} \text{ constant}\end{aligned}\tag{3.142}$$

Then by Lemma 3.6 applied to direction $\alpha=\frac{\Pi\mu}{\|\Pi\mu\|_2}$ we have $\sum_{i=1}^k\sum_{x\in C_i}\langle f_x-\mu_i,\alpha\rangle^2\leq\frac{4\epsilon}{\varphi^2}$. On the other hand using (3.142) we get

$$\frac{4\epsilon}{\varphi^2}\geq\sum_{i=1}^k\sum_{x\in C_i}\langle f_x-\mu_i,\alpha\rangle^2\geq\sum_{x\in\widehat{C}\cap(V\setminus C)}\left\langle f_x-\mu_x,\frac{\Pi\mu_x}{\|\Pi\mu_x\|_2}\right\rangle^2\geq 0.04\cdot\frac{|\widehat{C}\cap(V\setminus C)|}{|C|}.$$

Therefore we have $\left|\widehat{C}\cap(V\setminus C)\right|\leq 100\frac{\epsilon}{\varphi^2}|C|$. □

3.6.3 Partitioning scheme works with *exact* cluster means & dot products

The goal of this section is to present the main ideas behind the algorithms and the analysis. In this section we make a couple of simplifying assumptions. We assume that:

- We have access to real centers $\{\mu_1,\ldots,\mu_k\}$,
- Dot products computed by the algorithm are exact,
- A test, that relies on computing outer-conductance of candidate sets, for assessing the quality of clusters is perfect.

Whenever we use one (or more) of these assumptions we state them explicitly in the Lemmas.

Later in Section 3.6.5 we show that we can get rid of all of these assumptions.

In the previous section we showed geometric properties of the threshold sets. Recall that threshold sets are defined as follows:

$$C_{y,\theta} := \{x \in V : \langle f_x, y \rangle \geq \theta ||y||^2\}.$$

In this section, using these properties of threshold sets, we show an algorithm that given exact centers, access to real dot products and a perfect primitive for computing outer-conductance computes an ordered partition $(T_1, \ldots, T_b)$ of $\{\mu_1, \ldots, \mu_k\}$ such that $(T_1, \ldots, T_b)$ induces a good collection of clusters.

Algorithm 8 COMPUTEORDEREDPARTITION$(G, \widehat{\mu}_1, \widehat{\mu}_2, \ldots, \widehat{\mu}_k, s_1, s_2)$ ▷ $\widehat{\mu}_i$'s given as sets of points
▷ s_1 is # sampled points for size estimation
▷ s_2 is # of sampled points for conductance estimation

```
$S := \{\widehat{\mu}_1, \ldots, \widehat{\mu}_k\}$
for $i = 1$ to $\lceil \log(k) \rceil$ do
    $T_i := \emptyset$
    for $\widehat{\mu} \in S$ do
        $\psi :=$ OUTERCONDUCTANCE$\big(G, \widehat{\mu}, (T_1, T_2, \ldots, T_{i-1}), S, s_1, s_2\big)$    ▷ Algorithm 11
        if $\psi \leq O(\frac{\epsilon}{\varphi^2} \cdot \log(k))$ then
            $T_i := T_i \cup \{\widehat{\mu}\}$
    $S := S \setminus T_i$
    if $S = \emptyset$ then
        return (TRUE, $(T_1, \ldots, T_i)$)
return (FALSE, $\perp$)
```

To explain and analyze COMPUTEORDEREDPARTITION we first need to introduce another algorithm and some definitions.

Definition 3.11. For a set $\{a_1, \ldots, a_i\}$ we say a sequence $(S_1, \ldots, S_p)$ is an ordered partial partition of $\{a_1, \ldots, a_i\}$ if:

- $\bigcup_{j \in [p]} S_j \subseteq \{a_1, \ldots, a_i\}$,
- S_i's are pairwise disjoint.

Intuitively Algorithm ISINSIDE emulates CLASSIFYBYHYPERPLANEPARTITIONING on ordered partial partition $(T_1, \ldots, T_b)$. This intuition is made formal, after introducing Definition 3.12, in Remark 3.8. For this we need additional notation for clusters that are implicitly created by ISINSIDE. We define:

```
Algorithm 9 IsInside($x, \widehat{\mu}, (T_1, T_2, \dots, T_b), S$)
                                ▷ $T_i$'s are sets of $\widehat{\mu}_j$ where $\widehat{\mu}_j$'s are given as sets of points
                                ▷ see Section 3.5.6 for the reason of such representation
                                ▷ $S$ = set of not yet processed centers, $\widehat{\mu} \in S$
 1: for $i = 1$ to $b$ do
 2:     Let $\Pi$ be the projection onto the span$(\bigcup_{j<i} T_j)^{\perp}$.
 3:     Let $S_i = \left(\bigcup_{j \geq i} T_j\right) \cup S$
 4:     for $\widehat{\mu}_i \in T_i$ do
 5:         if $x \in C^{apx}_{\Pi\widehat{\mu}_i, 0.93} \setminus \bigcup_{\widehat{\mu}' \in S_i \setminus \{\widehat{\mu}_i\}} C^{apx}_{\Pi\widehat{\mu}', 0.93}$ then    ▷ see (3.141) for definition of $C^{apx}_{y,\theta}$
 6:             return False
 7: Let $\Pi$ be the projection onto the span$(\bigcup_{j \leq b} T_j)^{\perp}$.
 8: if $x \in C^{apx}_{\Pi\widehat{\mu}, 0.93} \setminus \bigcup_{\widehat{\mu}' \in S \setminus \{\widehat{\mu}\}} C^{apx}_{\Pi\widehat{\mu}', 0.93}$ then    ▷ see (3.141) for definition of $C^{apx}_{y,\theta}$
 9:     return True
10: return False
```

Definition 3.12 (**Candidate cluster**)**.** For an ordered partial partition $P = (T_1, \dots, T_p)$ of approximate cluster means $\{\widehat{\mu}_1, \dots, \widehat{\mu}_k\}$ and $\widehat{\mu} \in \{\widehat{\mu}_1, \dots, \widehat{\mu}_k\} \setminus \bigcup_{i \in [p]} T_i$ we say that $\widehat{C}^P_{\widehat{\mu}}$ is a **candidate cluster corresponding to $\widehat{\mu}$ with respect to** P if:

$$\widehat{C}^P_{\widehat{\mu}} = \left\{ x \in V : \textsc{IsInside}\left(x, \widehat{\mu}, P, \{\widehat{\mu}_1, \dots, \widehat{\mu}_k\} \setminus \bigcup_{i \in [p]} T_i \right) = \textsc{True} \right\}.$$

Furthermore we define: $V^P := V \setminus \bigcup_{j<p} \bigcup_{\widehat{\mu} \in T_j} \widehat{C}^{(T_1, \dots, T_{j-1})}_{\widehat{\mu}}$.

Algorithm IsInside receives a vertex x, the centre of a cluster $\widehat{\mu}$, and an ordered partial partition, then it tests if vertex x is not recovered by any of the previous stages (see line (5) of Algorithm 9) and can be recovered at the current stage using $\widehat{\mu}$. More formally, it can be recovered at the current stage if it only belongs to the candidate cluster corresponding to the center $\widehat{\mu}$ (see line (8) of Algorithm 9).

Remark 3.8. *Note that Definitions 3.10 and 3.12 are compatible in the following sense. For an ordered partition $(T_1, \dots, T_b)$ of approximate cluster means $\{\widehat{\mu}_1, \dots, \widehat{\mu}_k\}$ that induces a collection of clusters $\{\widehat{C}_{\widehat{\mu}_1}, \dots, \widehat{C}_{\widehat{\mu}_k}\}$ it is true that:*

$$\{\widehat{C}_{\widehat{\mu}_1}, \dots, \widehat{C}_{\widehat{\mu}_k}\} = \bigcup_{i \in [b]} \bigcup_{\widehat{\mu} \in T_i} \{\widehat{C}^{(T_1, \dots, T_{i-1})}_{\widehat{\mu}}\},$$

Equipped with Definition 3.12 we are ready to explain Algorithm ComputeOrderedPartition. The Algorithm proceeds in $O(\log(k))$ stages. It maintains a set S of approximate cluster means, that initially is equal to $\{\widehat{\mu}_1, \dots, \widehat{\mu}_k\}$, from which $\widehat{\mu}$'s are removed after every stage. At every stage i a collection of sets

$$\mathscr{C}_i := \bigcup_{\widehat{\mu} \in S} \{\widehat{C}^{(T_1, \dots, T_{i-1})}_{\widehat{\mu}}\},$$

is implicitly considered. In fact sets in this collection are, by definition, pairwise disjoint (see Defnition 3.12 and line: 8 of IsInside). $\widehat{C}_{\widehat{\mu}}^{(T_1,\ldots,T_{i-1})}$'s are defined as threshold sets (see Definition 3.8) that are made disjoint by removing intersections. The main idea behind the Algorithm is to use properties from Section 3.6.2 so that we can show that $\widehat{C}_{\widehat{\mu}}^{(T_1,\ldots,T_{i-1})}$'s match some C_j's well. Unfortunately after removing the intersections the above property might not hold for every cluster in $\mathscr{C}_i$. In the rest of this section we show however that it is true for a constant fraction of sets from $\mathscr{C}_i$. The Algorithm ComputeOrderedPartition proceeds by discarding, from set S, the $\widehat{\mu}$'s for which $\widehat{C}_{\widehat{\mu}}^{(T_1,\ldots,T_{i-1})}$ matches some C_j's well and implicitly removes the vertices of $\widehat{C}_{\widehat{\mu}}^{(T_1,\ldots,T_{i-1})}$ from consideration. Moreover it projects out the directions corresponding to the removed $\widehat{\mu}$'s and restricts its attention to a lower dimensional subspace Π of $\mathbb{R}^k$ (see **Idealized Clustering Algorithm** from Section 3.6.1 for comparison). The Algorithm doesn't know which sets from $\mathscr{C}_i$ are good as it runs in sublinear time. That is why we develop a simple sampling procedure that computes outer-conductance of candidate clusters (see Algorithm 11). Then the Algorithm removes the $\widehat{\mu}$'s for which the corresponding $\widehat{C}_{\widehat{\mu}}^{(T_1,\ldots,T_{i-1})}$ have small outer-conductance. We conclude using the robustness property of (k,φ,ϵ)-clusterable graphs (Lemma 3.16) that these tests are enough.

The rest of this subsection is devoted to showing that if ComputeOrderedPartition is called with $(\hat{\mu}_1,\ldots,\hat{\mu}_k)$ equal to $(\mu_1,\ldots,\mu_k)$ and the algorithm has access to real dot products then ComputeOrderedPartition returns True and an ordered partition $(T_1,\ldots,T_b)$ (of $\{\mu_1,\ldots,\mu_k\}$) that induces a collection of pairwise disjoint clusters $\{\widehat{C}_{\mu_1},\ldots,\widehat{C}_{\mu_k}\}$ such that for every i:

$$\phi\left(\widehat{C}_{\mu_i}\right) \le O\left(\frac{\epsilon}{\varphi^2}\cdot\log(k)\right). \tag{3.143}$$

Then using Lemma 3.16 we get that there exists a permutation π such that for all $i \in [k]$:

$$\left|\widehat{C}_{\mu_i}\triangle C_{\pi(i)}\right| \le O\left(\frac{\epsilon}{\varphi^3}\cdot\log(k)\right)|C_{\pi(i)}|. \tag{3.144}$$

The core of the argument is an averaging argument that, for every linear subspace of $\mathbb{R}^k$, bounds the average distance of embedded points to their centers in this subspace. What is important is that the bound depends linearly on the dimensionality of the subspace.

Lemma 3.33. *Let $k \ge 2$, $\varphi \in (0,1)$ and $\frac{\epsilon}{\varphi^2}$ be smaller than a sufficiently small constant. Let $G=(V,E)$ be a d-regular graph that admits a (k,φ,ϵ)-clustering $\{C_1,\ldots,C_k\}$. Then for all $L \subseteq \mathbb{R}^k$ - a linear subspace of $\mathbb{R}^k$, Π the orthogonal projection onto L we have:*

$$\sum_{x\in V}\|\Pi f_x - \Pi\mu_x\|_2^2 \le O\left(dim(L)\cdot\frac{\epsilon}{\varphi^2}\right)$$

Proof. Let $b := dim(L)$ and $\{w_1,\ldots,w_b\}$ be any orthonormal basis of L and recall that for $x \in V$

μ_x is the cluster mean of the cluster which x belongs to. Then

$$\begin{aligned}\sum_{x\in V} \|\Pi f_x - \Pi\mu_x\|_2^2 &= \sum_{x\in V}\sum_{i=1}^{b} \left\langle f_x - \mu_x, w_i\right\rangle^2 \\ &= \sum_{i=1}^{b}\sum_{x\in V} \left\langle f_x - \mu_x, w_i\right\rangle^2 \\ &\le b\cdot\frac{4\epsilon}{\varphi^2} && \text{By Lemma 3.6}\end{aligned}$$

□

In order to show (3.143) we need to show that a constant fraction of candidate sets $\widehat{C}_{\mu}^{(T_1,\dots,T_{i-1})}$'s match some C_j's well. To do that we argue that that sets of the form $C_{\Pi\widehat{\mu},0.9}$ (where Π is the orthogonal projection onto the span$(\bigcup_{j<i} T_j)^{\perp}$) don't overlap too much. We do this in two steps. First in Lemma 3.34 and Lemma 3.35 we show that points from the intersections are far from their centers. Then in Lemma 3.36 below we show that having too many such vertices would contradict Lemma 3.33.

Lemma 3.34. *Let $k \ge 2$, $\varphi \in (0,1)$ and $\frac{\epsilon}{\varphi^2}$ be smaller than a sufficiently small constant. Let $G=(V,E)$ be a d-regular graph that admits a (k,φ,ϵ)-clustering $\{C_1,\dots,C_k\}$. Let $\{v_1,\dots,v_k\}\in\mathbb{R}^k$ be a set of vectors satisfying:*

- $|\left\langle v_i, v_j\right\rangle| \le O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\frac{1}{\sqrt{|C_i||C_j|}}$
- $\left| ||v_i||^2 - \frac{1}{|C_i|}\right| \le O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\frac{1}{|C_i|}$

Then for every pair $i \neq j \in [k]$ for every $\theta \in (0,1)$ if $\alpha := \frac{v_i\frac{\|v_j\|}{\|v_i\|}+v_j\frac{\|v_i\|}{\|v_j\|}}{\sqrt{\|v_i\|^2+\|v_j\|^2}}$ and $I := C_{v_i,\theta}\cap C_{v_j,\theta} = \{x \in V : \langle f_x, v_i\rangle \ge \theta\|v_i\|^2 \wedge \langle f_x, v_j\rangle \ge \theta\|v_j\|^2\}$ then the following conditions hold:

1. *Correlation of vector v_p with the direction α is as follows:*
   - *for all $p \in [k]\setminus\{i,j\}$, $\left\langle \frac{\alpha}{\|\alpha\|}, v_p\right\rangle \le O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\cdot\frac{\|v_i\|\cdot\|v_j\|}{\sqrt{\|v_i\|^2+\|v_j\|^2}}$, for all $i \neq j \in [k]$*
   - *for all $p \in \{i,j\}$, $\left\langle \frac{\alpha}{\|\alpha\|}, v_p\right\rangle \le \left(1+O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\right)\cdot\frac{\|v_i\|\cdot\|v_j\|}{\sqrt{\|v_i\|^2+\|v_j\|^2}}$ for all $i \in [k]$*
2. *Spectral embeddings of vertices from set I have big correlation with direction α.*

$$\min_{x\in I}\left\langle \frac{\alpha}{\|\alpha\|}, f_x\right\rangle \ge \left(2\theta - O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\right)\cdot\frac{\|v_i\|\cdot\|v_j\|}{\sqrt{\|v_i\|^2+\|v_j\|^2}}$$

Proof. For all $p \in [k]$ let $\widetilde{v}_p := v_p/||v_p||$. Let $\gamma := \frac{||v_j||}{\sqrt{||v_i||^2+||v_j||^2}}$, $\alpha := \gamma\widetilde{v}_i + \sqrt{1-\gamma^2}\,\widetilde{v}_j$, and $\widetilde{\alpha} := \alpha/||\alpha||$. Fix $i \neq j \in [1,\dots,k]$. First we show that since v_i's are close to orthogonal we have

$||\alpha||^2 \approx 1$. More precisely we will upper bound $|||\alpha||^2-1|$

$$\begin{aligned}
\left|||\alpha||^2-1\right| &= \left|\gamma^2||\widetilde{v}_i||^2+(1-\gamma^2)||\widetilde{v}_j||^2+2\gamma\sqrt{1-\gamma^2}\left\langle \widetilde{v}_i,\widetilde{v}_j\right\rangle-1\right| \\
&= \frac{2\left\langle v_i,v_j\right\rangle}{||v_i||^2+||v_j||^2} && \text{as } ||\widetilde{v}_i||=||\widetilde{v}_j||=1 \\
&\le \frac{2\cdot O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\frac{1}{\sqrt{|C_i||C_j|}}}{\left(1-O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\right)\left(\frac{1}{|C_i|}+\frac{1}{|C_j|}\right)} && \text{By assumptions} \\
&\le O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\frac{\sqrt{|C_i||C_j|}}{|C_i|+|C_j|} \\
&\le O\left(\frac{\sqrt{\epsilon}}{\varphi}\right) && \text{as } \frac{\sqrt{|C_i||C_j|}}{\max(|C_i|,|C_j|)}\le 1
\end{aligned} \tag{3.145}$$

Observe the following fact:

$$\sqrt{1-\gamma^2}\cdot||v_j||=\gamma\cdot||v_i|| \tag{3.146}$$

Next notice the following:

$$\langle\alpha,v_i\rangle=\gamma||v_i||+\langle\widetilde{v}_i,\widetilde{v}_j\rangle\cdot\sqrt{1-\gamma^2}||v_i|| \tag{3.147}$$

$$\langle\alpha,v_j\rangle=\langle\widetilde{v}_i,\widetilde{v}_j\rangle\cdot\gamma||v_j||+\sqrt{1-\gamma^2}||v_j|| \tag{3.148}$$

For all $p\in\{1,2,\dots,k\}\setminus\{i,j\}$

$$\langle\alpha,v_p\rangle=\langle\widetilde{v}_i,\widetilde{v}_p\rangle\cdot\gamma||v_p||+\left\langle\widetilde{v}_j,\widetilde{v}_p\right\rangle\sqrt{1-\gamma^2}||v_p|| \tag{3.149}$$

Moreover for all $p\neq q\in[1,\dots,k]$ we have

$$\begin{aligned}
\left|\frac{1}{||v_p||^2}\cdot\left\langle v_q,v_p\right\rangle\right| &\le O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\frac{1}{\sqrt{|C_q||C_p|}}|C_p|\frac{1}{\left(1-O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\right)} && \text{By assumptions} \\
&\le O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\sqrt{\frac{|C_p|}{|C_q|}} && \text{for small enough } \frac{\epsilon}{\varphi^2} \\
&\le O\left(\frac{\sqrt{\epsilon}}{\varphi}\right) && \text{as } \frac{|C_p|}{|C_q|}=O(1)
\end{aligned} \tag{3.150}$$

Using the above we can prove:

$$\begin{aligned}\left|\langle\widetilde{v}_i,\widetilde{v}_j\rangle\cdot\sqrt{1-\gamma^2}||v_i||\right| &= \left|\sqrt{1-\gamma^2}\cdot||v_j||\cdot\frac{1}{||v_j||^2}\cdot\left\langle v_i,v_j\right\rangle\right| \\ &\leq \sqrt{1-\gamma^2}\cdot||v_j||\cdot O\left(\frac{\sqrt{\epsilon}}{\varphi}\right) && \text{By (3.150)} \\ &= O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\cdot\gamma\cdot||v_i|| && \text{By (3.146)} \qquad (3.151)\end{aligned}$$

And similarly we show:

$$\begin{aligned}\left|\langle\widetilde{v}_i,\widetilde{v}_j\rangle\cdot\gamma||v_j||\right| &= \left|\gamma\cdot||v_i||\cdot\frac{1}{||v_i||^2}\cdot\left\langle v_i,v_j\right\rangle\right| \\ &\leq O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\cdot\gamma\cdot||v_i|| && \text{By (3.150)} \qquad (3.152)\end{aligned}$$

For all $p\in\{1,2,\ldots,k\}\setminus\{i,j\}$ we get

$$\begin{aligned}\left|\left\langle\alpha,v_p\right\rangle\right| &\leq \left|\langle\widetilde{v}_i,\widetilde{v}_p\rangle\cdot\gamma||v_p||\right| + \left|\left\langle\widetilde{v}_j,\widetilde{v}_p\right\rangle\sqrt{1-\gamma^2}||v_p||\right| && \text{By (3.149)} \\ &= \left|\left\langle v_i,v_p\right\rangle\cdot\frac{1}{||v_i||^2}||v_i||\gamma\right| + \left|\left\langle v_j,v_p\right\rangle\frac{1}{||v_j||^2}||v_j||\sqrt{1-\gamma^2}\right| \\ &\leq O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\cdot\gamma\cdot||v_i|| && \text{By (3.150) and (3.146)} \\ & && (3.153)\end{aligned}$$

Combining (3.147), (3.148), (3.151), (3.152) and (3.153) we get that for all $p\in\{i,j\}$ we have

$$\left\langle\alpha,v_p\right\rangle\leq\left(1+O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\right)\cdot\gamma\cdot||v_i|| \qquad (3.154)$$

and for all $p\in\{1,\ldots,k\}\setminus\{i,j\}$

$$\left\langle\alpha,v_p\right\rangle\leq O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\cdot\gamma\cdot||v_i|| \qquad (3.155)$$

Now using (3.145) we get that for all $p\in\{i,j\}$

$$\left\langle\widetilde{\alpha},v_p\right\rangle\leq\frac{1}{\sqrt{1-O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)}}\left(1+O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\right)\cdot\gamma\cdot||v_i||\leq\left(1+O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\right)\cdot\frac{||v_i||||v_j||}{\sqrt{||v_i||^2+||v_j||^2}}$$

and for all $p\in\{1,\ldots,k\}\setminus\{i,j\}$

$$\left\langle\widetilde{\alpha},v_p\right\rangle\leq\frac{1}{\sqrt{1-O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)}}O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\cdot\gamma\cdot||v_i||\leq O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\cdot\frac{||v_i||||v_j||}{\sqrt{||v_i||^2+||v_j||^2}}$$

These two inequalities establish the first statement of the Claim.

Recall that

$$I = \{x \in V : \langle f_x, v_i \rangle \geq \theta ||v_i||^2 \wedge \langle f_x, v_j \rangle \geq \theta ||v_j||^2\}$$

Now let $x \in I$. Then observe

$$\begin{aligned} \left\langle \alpha, f_x \right\rangle &= \langle \gamma \cdot \widetilde{v}_i, f_x \rangle + \left\langle \sqrt{1-\gamma^2} \cdot \widetilde{v}_j, f_x \right\rangle \\ &\geq \gamma \cdot \theta \cdot ||v_i|| + \sqrt{1-\gamma^2} \cdot \theta \cdot ||v_j|| && \text{because } x \in I \\ &= 2\theta \cdot \gamma \cdot ||v_i|| && \text{by (3.146)} \end{aligned}$$

Hence

$$\begin{aligned} \left\langle \widetilde{\alpha}, f_x \right\rangle &\geq \frac{1}{\sqrt{1+O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)}} 2\theta \cdot \gamma \cdot ||v_i|| && \text{By (3.145)} \\ &\geq \left(2\theta - O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\right) \cdot \gamma \cdot ||v_i|| \end{aligned}$$

□

Now we use technical Lemma 3.34 to show that vertices from the intersections of $C_{\Pi\mu,0.9}$'s are far from their centers.

Lemma 3.35. *Let $k \geq 2$, $\varphi \in (0,1)$ and $\frac{\epsilon}{\varphi^2}$ be smaller than a sufficiently small constant. Let $G = (V,E)$ be a d-regular graph that admits a (k,φ,ϵ)-clustering $\{C_1,\ldots,C_k\}$. If μ_i's are cluster means then the following conditions hold. For all $S \subset \{\mu_1,\ldots,\mu_k\}$ if $L := span(S)^\perp$ and Π is the projection on L then if $x \in V$ is such that*

$$\left\langle \Pi f_x, \Pi\mu_i \right\rangle \geq 0.9 \|\Pi\mu_i\|_2^2 \wedge \left\langle \Pi f_x, \Pi\mu_j \right\rangle \geq 0.9 \|\Pi\mu_j\|_2^2$$

for some $\mu_i, \mu_j \in \{\mu_1,\ldots,\mu_k\} \setminus S, \mu_i \neq \mu_j$. Then:

$$\|\Pi f_x - \Pi\mu_x\| \geq 0.3\sqrt{\frac{1}{\max_{p\in[k]} |C_p|}}$$

Proof. Let $x \in V$ be such that $\left\langle \Pi f_x, \Pi\mu_i \right\rangle \geq 0.9\|\Pi\mu_i\|_2^2$ and $\left\langle \Pi f_x, \Pi\mu_j \right\rangle \geq 0.9\|\Pi\mu_j\|_2^2$. Note that by Lemma 3.12 set $\{\Pi\mu_1,\ldots,\Pi\mu_k\}$ satisfies assumptions of Lemma 3.34. So applying Lemma 3.34 for $\theta = 0.9$ we get that there exists $\alpha \in \text{span}\{\Pi\mu_i, \Pi\mu_j\}, ||\alpha|| = 1$ such that:

- $\left\langle \alpha, f_x \right\rangle = \left\langle \alpha, \Pi f_x \right\rangle \geq (1.8 - O(\frac{\sqrt{\epsilon}}{\varphi})) \cdot \frac{\|\Pi\mu_i\|\cdot\|\Pi\mu_j\|}{\sqrt{\|\Pi\mu_i\|^2+\|\Pi\mu_j\|^2}}$
- $\left\langle \alpha, \Pi\mu_p \right\rangle \leq (1 + O(\frac{\sqrt{\epsilon}}{\varphi})) \cdot \frac{\|\Pi\mu_i\|\cdot\|\Pi\mu_j\|}{\sqrt{\|\Pi\mu_i\|^2+\|\Pi\mu_j\|^2}}$, for all $p \in [k]$

Thus we get

$$\begin{aligned}||\Pi f_x - \Pi\mu_x|| &\geq |\langle \alpha, \Pi f_x\rangle - \langle \alpha, \Pi\mu_x\rangle| \\ &\geq \left(0.8 - O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\right) \cdot \frac{\|\Pi\mu_i\| \cdot \|\Pi\mu_j\|}{\sqrt{\|\Pi\mu_i\|^2 + \|\Pi\mu_j\|^2}} \\ &\geq 0.75 \cdot \frac{\|\Pi\mu_i\| \cdot \|\Pi\mu_j\|}{\sqrt{\|\Pi\mu_i\|^2 + \|\Pi\mu_j\|^2}} \qquad \text{By assumption that } \frac{\epsilon}{\varphi^2} \text{ small} \end{aligned} \tag{3.156}$$

without loss of generality we can assume $||\Pi\mu_i|| \geq ||\Pi\mu_j||$. Then we get:

$$\begin{aligned}\frac{\|\Pi\mu_i\| \cdot \|\Pi\mu_j\|}{\sqrt{\|\Pi\mu_i\|^2 + \|\Pi\mu_j\|^2}} &= \frac{\|\Pi\mu_j\|}{\sqrt{1 + \|\Pi\mu_j\|^2 / ||\Pi\mu_i||^2}} \\ &\geq \frac{1}{\sqrt{2}} ||\Pi\mu_j|| \\ &\geq \frac{1}{2\sqrt{\max_{p\in[k]}|C_p|}} \qquad \text{Lemma 3.12, assumption that } \frac{\epsilon}{\varphi^2} \text{ small}\end{aligned} \tag{3.157}$$

Combining (3.156) and (3.157) we get:

$$||\Pi f_x - \Pi\mu_x|| \geq 0.3 \cdot \frac{1}{\sqrt{\max_{p\in[k]}|C_p|}}$$

□

Combining Lemma 3.33 and Lemma 3.35 we show that sets $C_{\Pi\mu,0.9}$'s don't overlap much.

Lemma 3.36. *Let $k \geq 2$, $\varphi \in (0,1)$ and $\frac{\epsilon}{\varphi^2}$ be smaller than a sufficiently small constant. Let $G = (V, E)$ be a d-regular graph that admits a (k, φ, ϵ)-clustering $\{C_1, \ldots, C_k\}$. If μ_i's are cluster means then the following conditions hold. For all $S \subset \{\mu_1, \ldots, \mu_k\}$ if $L := span(S)^\perp$, $dim(L) = b$ and Π is projection on L then:*

$$\left| \bigcup_{\substack{\mu,\mu' \in \{\mu_1,\ldots,\mu_k\}\setminus S \\ \mu\neq\mu'}} C_{\Pi\mu,0.9} \cap C_{\Pi\mu',0.9} \right| \leq O\left(b \cdot \frac{\epsilon}{\varphi^2}\right) \cdot \frac{n}{k}.$$

Proof. Let $x \in V$ be such that $\langle \Pi f_x, \Pi\mu\rangle \geq 0.9\|\Pi\mu\|_2^2$ and $\langle \Pi f_x, \Pi\mu'\rangle \geq 0.9\|\Pi\mu'\|_2^2$ for some

$\mu, \mu' \in \{\mu_1, \dots, \mu_k\} \setminus S$. Then by Lemma 3.35 we get that

$$\|\Pi f_x - \Pi \mu_x\| \geq 0.3 \sqrt{\frac{1}{\max_{p \in [k]} |C_p|}}. \tag{3.158}$$

On the other hand Lemma 3.33 guarantees:

$$\sum_{x \in V} \|\Pi f_x - \Pi \mu_x\|_2^2 \leq O\left(dim(L) \cdot \frac{\epsilon}{\varphi^2} \right) \tag{3.159}$$

Combining (3.158), (3.159) and the fact that $\frac{\max_{p \in [k]} |C_p|}{\min_{p \in [k]} |C_p|} = O(1)$ we get

$$\left| \bigcup_{\mu, \mu' \in \{\mu_1, \dots, \mu_k\} \setminus S} C_{\Pi\mu, 0.93} \cap C_{\Pi\mu', 0.93} \right| \leq O\left(b \cdot \frac{\epsilon}{\varphi^2} \right) \cdot \frac{n}{k}$$

□

Our bounds above enable the following analysis. At every stage of the for loop from line 4 of Algorithm 8 at least half of the candidate clusters:

$$\mathscr{C}_i := \bigcup_{\widehat{\mu} \in S} \{\widehat{C}_{\widehat{\mu}}^{(T_1, \dots, T_{i-1})}\},$$

passes the test from line 6 of Algorithm 8, which means that they have small outer-conductance and satisfy condition (3.143).

Lemma 3.37. *Let $k \geq 2$, $\varphi \in (0,1)$ and $\frac{\epsilon}{\varphi^2} \cdot \log(k)$ be smaller than a sufficiently small constant. Let $G = (V, E)$ be a d-regular graph that admits a (k, φ, ϵ)-clustering $\{C_1, \dots, C_k\}$.*

If COMPUTEORDEREDPARTITION*$(G, \widehat{\mu}_1, \widehat{\mu}_2, \dots, \widehat{\mu}_k, s_1, s_2)$ is invoked with $(\widehat{\mu}_1, \dots, \widehat{\mu}_k) = (\mu_1, \dots, \mu_k)$ and we assume that all tests Algorithm 8 performs $\left(i.e. \langle f_x, \widehat{\Pi}\widehat{\mu} \rangle_{apx} \overset{?}{\geq} 0.93 \left\| \widehat{\Pi}\widehat{\mu} \right\|_{apx}^2 \right)$ are exact and* OUTERCONDUCTANCE *computes outer-conductance precisely then there exists an absolute constant Υ such that the following conditions hold.*

For any $i \in [0..\log(k)]$ assume that at the beginning of the i-th iteration of the for loop from line 4 of Algorithm 8 $|S| = b$ and, up to renaming of μ's, $S = \{\mu_1, \dots, \mu_b\}$, the corresponding clusters are $\mathscr{C} = \{C_1, \dots, C_b\}$ respectively and the ordered partial partition of μ's is equal to $(T_1, \dots, T_{i-1})$. Then if for every $C \in \mathscr{C}$ we have that $|V^{(T_1, \dots, T_{i-1})} \cap C| \geq \left(1 - \Upsilon \cdot i \cdot \frac{\epsilon}{\varphi^2}\right) |C|$ then at the beginning of $(i+1)$-th iteration:

1. *$|S| \leq b/2$ (that is at least half of the remaining cluster means were removed in i-th iteration),*
2. *for every $\mu \in S$ the corresponding cluster C satisfies $|V^{(T_1, \dots, T_i)} \cap C| \geq \left(1 - \Upsilon \cdot (i+1) \cdot \frac{\epsilon}{\varphi^2}\right) |C|$, where $(T_1, \dots, T_i)$ is the ordered partial partition of μ's created in the first i iterations.*

Proof. Let $i \in [0..\log(k)]$, without loss of generality we can assume that $S = \{\mu_1, \ldots, \mu_b\}$ (if not we can rename the μ's) at the beginning of the i-th iteration and the corresponding clusters be $\mathscr{C} = \{C_1, \ldots, C_b\}$ respectively. Assume that for every $C \in \mathscr{C}$ we have that $|V^{(T_1,\ldots,T_{i-1})} \cap C| \geq \left(1 - \Upsilon \cdot i \cdot \frac{\epsilon}{\varphi^2}\right)|C|$. We start by showing the first part of the Lemma.

At least half of the cluster means is removed from S:

Let $\mu \in S$, Π_i be the orthogonal projection onto the span$(\bigcup_{j<i} T_j)^{\perp}$, where $(T_1, \ldots, T_{i-1})$ is the ordered partial partition of $\{\mu_1, \ldots, \mu_k\}$ created before iteration i by COMPUTEORDEREDPARTITION. For brevity we will refer to $(T_1, \ldots, T_{i-1})$ as P in this proof. Let

$$I := \bigcup_{\mu', \mu'' \in \{\mu_1, \ldots, \mu_b\}} C_{\Pi_i \mu', 0.93} \cap C_{\Pi_i \mu'', 0.93}.$$

By Lemma 3.36 we have that

$$|I| \leq O\left(b \cdot \frac{\epsilon}{\varphi^2}\right) \cdot \frac{n}{k}$$

So by Markov inequality we get that there exists a subset of clusters $\mathscr{R} \subseteq \mathscr{C}$ such that $|\mathscr{R}| \geq b/2$ and for every $C \in \mathscr{R}$ we have that

$$|C \cap I| \leq 2 \cdot O\left(\frac{\epsilon}{\varphi^2}\right) \cdot \frac{n}{k} \tag{3.160}$$

We will argue that for any order of the for loop from line 4 of Algorithm 8 it is true that for every $C \in \mathscr{R}$ with corresponding mean μ the candidate cluster $\widehat{C}_\mu^P$ satisfies the if statement from line 6.

First note that behavior of the algorithm is independent of the order of the for loop from line 4 of Algorithm 8 as by definition $\widehat{C}_\mu^P$'s for $\mu \in S$ are pairwise disjoint. Now let $C \in \mathscr{R}$, μ be the corresponding mean to C and $\widehat{C}_\mu^P$ be the candidate cluster corresponding to μ with respect to $P = (T_1, \ldots, T_{i-1})$. By inductive assumption $|V^P \cap C| \geq \left(1 - \Upsilon \cdot i \cdot \frac{\epsilon}{\varphi^2}\right)|C|$ so by (3.160), Lemma 3.31 and the fact that $\frac{\max_{p \in [k]} |C_p|}{\min_{p \in [k]} |C_p|} = O(1)$ we get that:

$$\begin{aligned} |\widehat{C}_\mu^P \cap C| &\geq \left(1 - \Upsilon \cdot i \cdot \frac{\epsilon}{\varphi^2}\right)|C| - O\left(\frac{\epsilon}{\varphi^2}\right)\frac{n}{k} - O\left(\frac{\epsilon}{\varphi^2}\right)|C| \\ &\geq \left(1 - O\left(\frac{\epsilon}{\varphi^2} \cdot \log(k)\right)\right)|C| \end{aligned} \tag{3.161}$$

To prove that $\widehat{C}_\mu^P$ passes the outer-conductance test we also need to show that $\widehat{C}_\mu^P$ doesn't contain a lot of points from $V^P \setminus C$. By Lemma 3.32 we get that:

$$|\widehat{C}_\mu^P \cap (V^P \setminus C)| \leq |\widehat{C}_\mu^P \cap (V \setminus C)| \leq O\left(\frac{\epsilon}{\varphi^2}\right)|C|. \tag{3.162}$$

Combining (3.162) and (3.161) we get that:

$$|\widehat{C}^P_\mu \triangle C| \le O\left(\frac{\epsilon}{\varphi^2}\cdot\log(k)\right)|C| \tag{3.163}$$

Now we want to argue that $\widehat{C}^P_\mu$ passes the outerconductance test from line 6 of Algorithm 8. From the definition of outer conductance:

$$\begin{aligned}
\phi(\widehat{C}^P_\mu) &\le \frac{E(C, V\setminus C) + d|\widehat{C}^P_\mu \triangle C|}{d(|C| - |\widehat{C}^P_\mu \triangle C|)} \\
&\le \frac{E(C, V\setminus C) + d\cdot O\left(\frac{\epsilon}{\varphi^2}\cdot\log(k)\right)|C|}{d(|C| - O\left(\frac{\epsilon}{\varphi^2}\cdot\log(k)\right)|C|)} && \text{from (3.163)} \\
&\le \frac{O\left(\frac{\epsilon}{\varphi^2}\right) + O\left(\frac{\epsilon}{\varphi^2}\cdot\log(k)\right)}{1 - O\left(\frac{\epsilon}{\varphi^2}\cdot\log(k)\right)} && \text{because } \frac{E(C, V\setminus C)}{d|C|} \le O\left(\frac{\epsilon}{\varphi^2}\right) \\
&\le O\left(\frac{\epsilon}{\varphi^2}\cdot\log(k)\right) && \text{for sufficiently small } \frac{\epsilon}{\varphi^2}\cdot\log(k)
\end{aligned}$$

and it follows that

$$\phi(\widehat{C}^P_\mu) \le O\left(\frac{\epsilon}{\varphi^2}\cdot\log(k)\right),$$

which means that $\widehat{C}^P_\mu$ passes the test as we assumed that OUTERCONDUCTANCE computes outer-conductance precisely.

Clusters corresponding to unremoved μ's satisfy condition 2:

Now we prove that for every μ that was not removed from set S only small fraction of its corresponding cluster is removed.

Let $\mu \in S$ be such that it is not removed in the i-th step. Let Π_i be the orthogonal projection onto the span$(\bigcup_{j<i} T_j)^\perp$. Let $C \in \mathscr{C}$ be the cluster corresponding to μ. By assumption $|V^P \cap C| \ge \left(1 - \Upsilon\cdot i\cdot\frac{\epsilon}{\varphi^2}\right)|C|$. Now let $x \in V^{(T_1,\dots,T_{i-1})} \setminus V^{(T_1,\dots,T_i)}$, where $(T_1,\dots,T_i)$ is the partial partition of μ's created in the first i-th steps of the for loop. We get that there exists $\mu' \in \{\mu_1,\dots,\mu_b\}$ such that $x \in \widehat{C}^P_{\mu'}$ (recall that $\widehat{C}^P_{\mu'}$ is the candidate cluster corresponding to μ' with respect to $P = (T_1,\dots,T_{i-1})$). Recall (Definition 3.12) that $\widehat{C}^P_{\mu'}$ is defined as:

$$\widehat{C}^P_{\mu'} = \left\{x \in V : \text{ISINSIDE}\left(x, \mu', P, \{\mu_1,\dots,\mu_k\}\setminus \bigcup_{j\in[i-1]} T_j\right) = \text{TRUE}\right\}.$$

This in particular means (see line 8: of Algorithm ISINSIDE) that:

$$\widehat{C}^P_{\mu'} \subseteq C_{\Pi_i\mu',0.93} \setminus \bigcup_{\mu''\in S\setminus\{\mu'\}} C_{\Pi_i\mu'',0.93},$$

which, as $\mu \in S \setminus \{\mu'\}$, gives us that:

$$\widehat{C}^P_{\mu'} \cap C_{\Pi_i \mu, 0.93} = \emptyset,$$

and finally, using Definition 3.8, we have:

$$\langle f_x, \Pi_i \mu \rangle < 0.93 ||\Pi_i \mu||^2. \tag{3.164}$$

But by Lemma 3.31:

$$|\{x \in C : \langle \Pi_i f_x, \Pi_i \mu \rangle < 0.93 \|\Pi_i \mu\|_2^2\}| \le O\left(\frac{\epsilon}{\varphi^2}\right) \cdot |C| \tag{3.165}$$

Combining (3.164) and (3.165) we get that:

$$|C \cap (V^{(T_1,\dots,T_{i-1})} \setminus V^{(T_1,\dots,T_i)})| \le O\left(\frac{\epsilon}{\varphi^2}\right)|C|. \tag{3.166}$$

By assumption that $|V^{(T_1,\dots,T_{i-1})} \cap C| \ge \left(1 - \Upsilon \cdot i \cdot \frac{\epsilon}{\varphi^2}\right)|C|$ and (3.166) we get that:

$$|V^{(T_1,\dots,T_i)} \cap C| \ge \left(1 - \Upsilon \cdot (i+1) \cdot \frac{\epsilon}{\varphi^2}\right)|C|,$$

provided that Υ is bigger than the constant from O notation in (3.166), which is the same constant as the one in the statement of Lemma 3.31.

□

Remark 3.9. *Note that in this section we assume that the Algorithm has access to real centers $\{\mu_1,\dots,\mu_k\}$. If it was the case in the final algorithm we could in fact prove a stronger guarantee, i.e. "Algorithm 8 returns* TRUE *and an ordered partition $(T_1,\dots,T_b)$ (of $\{\mu_1,\dots,\mu_k\}$) that induces a collection of pairwise disjoint clusters $\{\widehat{C}_{\mu_1},\dots,\widehat{C}_{\mu_k}\}$ such that there exists a permutation π such that for all $i \in [k]$:*

$$\left|\widehat{C}_{\mu_i} \triangle C_{\pi(i)}\right| \le O\left(\frac{\epsilon}{\varphi^2} \cdot \log(k)\right)|C_{\pi(i)}|".$$

Compare the above statement with with (3.144) *and the main theorem of this section, Theorem 3.7. The reason we present it this way is the following.*

The final algorithm doesn't have access to μ's but instead tests many candidate sets $\{\widehat{\mu}_1,\dots,\widehat{\mu}_k\}$. Moreover Algorithm 8 returns an ordered partition $(T_1,\dots,T_b)$ that induces a collection of clusters $\{\widehat{C}_1,\dots,\widehat{C}_k\}$ whenever every set from this collection passes the test from line 6 of COMPUTEORDEREDPARTITION, *that is when for every $\widehat{C} \in \{\widehat{C}_1,\dots,\widehat{C}_k\}$:*

$$\phi\left(\widehat{C}\right) \le O\left(\frac{\epsilon}{\varphi^2} \cdot \log(k)\right).$$

This in particular means that Algorithm 8 may return TRUE *even for a set* $\{\widehat{\mu}_1,\ldots,\widehat{\mu}_k\}$ *that is* ***not*** *a good approximation to* $\{\mu_1,\ldots,\mu_k\}$.

Because of that, once we know that COMPUTEORDEREDPARTITION *invoked with* $\{\mu_1,\ldots,\mu_k\}$ *returns an ordered partition* $(T_1,\ldots,T_b)$ *that induces a collection of clusters* $\{\widehat{C}_1,\ldots,\widehat{C}_k\}$*, when proving the final result of this section (Theorem 3.7) the only thing we assume about* $\widehat{C}$*'s is that they passed the outer-conductance test. And that is why we use Lemma 3.16 and we "loose" a factor* $\frac{1}{\varphi}$ *in the final guarantee.*

Moreover structuring the argument in this way helps the presentation as later, in Section 3.6.5, the proof will follow a similar structure.

The following Theorem concludes this subsection by showing (3.144). It does so by induction using Lemma 3.37 as an inductive step. At the end it uses Lemma 3.16 to go from the guarantees for outer-conductance to guarantees for recovery.

Theorem 3.7. *Let* $k \geq 2$, $\varphi \in (0,1)$ *and* $\frac{\epsilon}{\varphi^2}\log(k)$ *be smaller than a sufficiently small constant. Let* $G=(V,E)$ *be a* d*-regular graph that admits a* (k,φ,ϵ)*-clustering* $\{C_1,\ldots,C_k\}$.

If COMPUTEORDEREDPARTITION$(G,\widehat{\mu}_1,\widehat{\mu}_2,\ldots,\widehat{\mu}_k,s_1,s_2)$ *is invoked with* $(\hat{\mu}_1,\ldots,\hat{\mu}_k)=(\mu_1,\ldots,\mu_k)$ *and we assume that all tests Algorithm 8 performs* $\left(i.e. \left\langle f_x, \widehat{\Pi}\widehat{\mu}\right\rangle_{apx} \overset{?}{\geq} 0.93 \left\|\widehat{\Pi}\widehat{\mu}\right\|_{apx}^2\right)$ *are exact and* OUTERCONDUCTANCE *computes outer-conductance precisely then the following conditions hold.*

COMPUTEORDEREDPARTITION *returns* (TRUE, $(T_1,\ldots,T_b)$) *such that* $(T_1,\ldots,T_b)$ *induces a collection of clusters* $\{\widehat{C}_{\mu_1},\ldots,\widehat{C}_{\mu_k}\}$ *such that there exists a permutation* π *on* k *elements such that for all* $i \in [k]$*:*

$$\left|\widehat{C}_{\mu_i} \triangle C_{\pi(i)}\right| \leq O\left(\frac{\epsilon}{\varphi^3}\cdot\log(k)\right)|C_{\pi(i)}|$$

and

$$\phi(\widehat{C}_{\mu_i}) \leq O\left(\frac{\epsilon}{\varphi^2}\cdot\log(k)\right).$$

Proof. Note that for $i=0$ in the for loop in line 2 of COMPUTEORDEREDPARTITION S and clusters $\{C_1,\ldots,C_k\}$ trivially satisfy assumptions of Lemma 3.37. So using Lemma 3.37 and induction we get that for every $i \in [0..\lceil\log(k)\rceil]$ at the beginning of the i-th iteration:

- $|S| \leq k/2^i$,
- for every $\mu \in S$ and the corresponding cluster C we have $|V^{(T_1,\ldots,T_{i-1})} \cap C| \geq \left(1-\Upsilon\cdot i\cdot\frac{\epsilon}{\varphi^2}\right)|C|$ (where Υ is the constant from the statement of Lemma 3.37).

In particular this means that after at most $\lceil\log(k)\rceil$ iterations set S becomes empty. This also means that COMPUTEORDEREDPARTITION returns in line 10, so it returns TRUE and the ordered partial partition $(T_1,\ldots,T_b)$ is in fact an ordered partition of $\{\mu_1,\ldots,\mu_k\}$.

Note that by definition (see Definition 3.10) all the approximate clusters $\{\widehat{C}_{\mu_1},\dots,\widehat{C}_{\mu_k}\}$ are pairwise disjoint and moreover for every constructed cluster $\widehat{C}\in\{\widehat{C}_{\mu_1},\dots,\widehat{C}_{\mu_k}\}$ we have:

$$\phi(\widehat{C})\le O\left(\frac{\epsilon}{\varphi^2}\cdot\log(k)\right),$$

as it passed the test in line 6 of ComputeOrderedPartition. So by Lemma 3.16 it means that there exists a permutation π on k elements such that for all $i\in[k]$:

$$\left|\widehat{C}_{\mu_i}\triangle C_{\pi(i)}\right|\le O\left(\frac{\epsilon}{\varphi^3}\cdot\log(k)\right)|C_{\pi(i)}|.$$

□

3.6.4 Finding the cluster means

In the previous subsection we showed that ComputeOrderedPartition succeeds if we have access to real cluster centers (i.e. μ_i's). In this section we present a search procedure for finding the centers.

The main idea behind our algorithm is to guess the clustering assignment of few random nodes and use this assignment to compute the approximate cluster means. More precisely, the first step of our algorithm is to learn the spectral embedding as described in Section 3.5. Then we sample $s=\Omega(\frac{\varphi^2}{\epsilon}\cdot k^4\log(k))$ random nodes and we consider all the possible clustering assignments for them. For each assignment, we implicitly define the cluster center for a specific cluster as $\widehat{\mu}_i := \frac{1}{|P_i|}\sum_{x\in P_i} f_x$.

Remark 3.10. *We note that in* FindCenters *we don't necessarily find $\mu_1,\dots,\mu_k$ exactly but we are able to show (see Section 3.6.4) that it finds a good approximation to μ_i's. Then in Section 3.6.5 we show that such approximation is sufficient for the partitioning scheme to work.*

Algorithm 10 FindCenters(G,η,δ)

```
InitializeOracle$(G,\delta)$
for $t\in[1\dots\log(2/\eta)]$ do
    $S :=$ Random sample of vertices of $V$ of size $s=\Theta(\frac{\varphi^2}{\epsilon}k^4\log(k))$
    for $(P_1,P_2,\dots,P_k)\in$ Partitions$(S)$ do
        for $i=1$ to $k$ do
            $\widehat{\mu}_i := \frac{1}{|P_i|}\sum_{x\in P_i} f_x$        ▷ Note that we compute the centers only implicitly.
        $(r,C) :=$
        ComputeOrderedPartition$\left(G,(\widehat{\mu}_1,\widehat{\mu}_2,\dots,\widehat{\mu}_k),\Theta\left(\frac{\varphi^2}{\epsilon}k^5\log^2(k)\log(1/\eta)\right),\Theta\left(\frac{\varphi^4}{\epsilon^2}k^5\log^2(k)\log(1/\eta)\right)\right)$
        if $r=$ True then
            return $C$
```

Quality of cluster means approximation

In the previous Section 3.6.3 we showed that the partitioning scheme works if we can find $\mu_1,\dots,\mu_k$ exactly. In this section we show that it is possible to estimate the cluster means with a small error factor (i.e $\mu_i \approx \widehat{\mu}_i$). Later in Section 3.6.5 we show that such an approximation to μ_i's is enough for the partitioning scheme to work.

In the rest of this section we show that if PARTITIONS(S) (see Algorithm 10) computes a correct guess of cluster assignments then the cluster means computed in line (6) are close to the real cluster means with constant probability. Then we repeat the procedure $O(\log(1/\eta))$ times to achieve success probability of at least $1-\eta$.

In particular, in Lemma 3.39 we show using Matrix Bernstein that if we have enough samples in a cluster i then $\|\mu_i - \widehat{\mu}_i\|_2 \le \zeta\cdot\|\mu_i\|_2$. Then we prove that if we sample enough random nodes we have enough samples in every cluster.

Before proving Lemma 3.39 we show a tail bound for the spectral projection of a node that will be useful to apply Matrix Bernstein.

Lemma 3.38. *Let $k \ge 2$, $\varphi \in (0,1)$ and $\frac{\epsilon}{\varphi^2}\log(k)$ be smaller than a sufficiently small constant. Let $G=(V,E)$ be a d-regular and a (k,φ,ϵ)-clusterable graph. Let $\beta > 1$ .Let*

$$T = \left\{ x \in V : ||f_x||_\infty \ge \beta\cdot\sqrt{\frac{10}{\min_{i\in[k]}|C_i|}} \right\}.$$

Then we have $|T| \le k\cdot\left(\frac{\beta}{2}\right)^{-\varphi^2/20\cdot\epsilon}\cdot(\min_{i\in[k]}|C_i|)$.

Proof. Recall that $f_x = U_{[k]}^T \mathbb{1}_x$, and u_i denote the i^{th} column of $U_{[k]}$. Thus we have $\|f_x\|_\infty = \max_{i\in[k]}\{u_i(x)\}$. Let $s_{\min} = \min_{i\in k}|C_i|$. We define

$$T_i = \left\{ x \in V : |u_i(x)| \ge \beta\cdot\sqrt{\frac{10}{s_{\min}}} \right\}$$

Therefore, by Lemma 3.4 we have $|T_i| \le \left(\frac{\beta}{2}\right)^{-\varphi^2/20\cdot\epsilon}\cdot s_{\min}$. Note that $T=\bigcup_{i=1}^k T_i$. Therefore we have

$$|T| \le k\cdot\left(\frac{\beta}{2}\right)^{-\varphi^2/20\cdot\epsilon}\cdot s_{\min}$$

□

Now we are ready to derive a bound on the difference between μ_i and $\widehat{\mu}_i$.

Lemma 3.39. *Let $\zeta,\delta \in (0,1)$, $k \ge 2$, $\varphi \in (0,1)$, $\frac{\epsilon\log k}{\varphi^2}$ be smaller than a positive sufficiently small constant. Let $G=(V,E)$ be a d-regular graph that admits a (k,φ,ϵ)-clustering $C_1,\dots,C_k$. Let $s \ge$*

$c\cdot\left(k\cdot\log\left(\frac{k}{\delta}\right)\cdot\left(\frac{1}{\delta}\right)^{(80\cdot\epsilon/\varphi^2)}\cdot\left(\frac{1}{\zeta}\right)^2\right)^{1/(1-(80\cdot\epsilon/\varphi^2))}$ *for large enough constant c. Let* $S=\{x_1,x_2,\ldots,x_s\}$ *be the multiset with s vertices sampled uniformly at random from cluster C. Let* $\mu=\frac{1}{|C|}\sum_{x\in C}f_x$ *denote the cluster mean, and let* $\widehat{\mu}=\frac{1}{s}\sum_{i=1}^{s}f_x$ *denote the empirical cluster mean. Then with probability at least* $1-\delta$ *we have*

$$\|\mu-\widehat{\mu}\|_2\le\zeta\cdot\|\mu\|_2$$

Proof. Let $s_{\min}:=\min_{i\in[k]}|C_i|$. We define

$$C'=\left\{x\in C:||f_x||_\infty\le 2\cdot\left(\frac{s\cdot k}{\delta}\right)^{(40\cdot\epsilon/\varphi^2)}\cdot\sqrt{\frac{10}{s_{\min}}}\right\}$$

Note that by Lemma 3.38 and by choice of $\beta=2\cdot\left(\frac{s\cdot k}{\delta}\right)^{(40\cdot\epsilon/\varphi^2)}$ we have

$$|C\setminus C'|\le k\cdot\left(\frac{\beta}{2}\right)^{-\varphi^2/(20\cdot\epsilon)}\cdot s_{\min}\le k\cdot\left(\frac{s\cdot k}{\delta}\right)^{-2}\cdot|C|=(k^{-1}\cdot s^{-2}\cdot\delta^2)\cdot|C|$$

Thus we have

$$|C'|\ge\left(1-(k^{-1}\cdot s^{-2}\cdot\delta^2)\right)|C| \tag{3.167}$$

Let $\mu'=\frac{1}{|C'|}\sum_{x\in C'}f_x$. By triangle inequality we have

$$||\widehat{\mu}-\mu||_2\le||\widehat{\mu}-\mu'||_2+||\mu'-\mu||_2 \tag{3.168}$$

In the rest of the proof we will upper bound both of these terms by $\frac{\zeta}{2}\cdot||\mu||_2$.

Step 1: We first prove $||\widehat{\mu}-\mu'||_2\le\frac{\zeta}{2}\cdot||\mu||_2$. By the assumption of the lemma for sufficiently small $\frac{\epsilon\log k}{\varphi^2}$ we have $k^{(40\cdot\epsilon/\varphi^2)}\le 2$. Thus for any $x\in C'$ we have $||f_x||_\infty\le\left(\frac{s}{\delta}\right)^{(40\cdot\epsilon/\varphi^2)}\cdot\sqrt{\frac{160}{s_{\min}}}$. Therefore by triangle inequality we have

$$||\mu'||_2=\left\|\frac{1}{|C'|}\cdot\sum_{x\in C'}f_x\right\|\le\frac{1}{|C'|}\cdot\sum_{x\in C'}||f_x||_2\le\frac{\sqrt{k}}{|C'|}\cdot\sum_{x\in C'}||f_x||_\infty\le\left(\frac{s}{\delta}\right)^{(40\cdot\epsilon/\varphi^2)}\cdot\sqrt{\frac{160\cdot k}{s_{\min}}}. \tag{3.169}$$

By (3.167) and by union bound over all samples in S with probability at least $1-s\cdot(k^{-1}\cdot s^{-2}\cdot\delta^2)=1-s^{-1}\cdot k^{-1}\cdot\delta^2\ge1-\frac{\delta}{2}$ for all $x_i\in S$ we have $x_i\in C'$, hence, $||f_x||_\infty\le\left(\frac{s}{\delta}\right)^{(40\cdot\epsilon/\varphi^2)}\cdot\sqrt{\frac{160}{s_{\min}}}$. Thus with probability at least $1-\frac{\delta}{2}$, S is chosen uniformly at random from C' so for all $x_i\in S$ we have

$$|f_x||_\infty\le\left(\frac{s}{\delta}\right)^{(40\cdot\epsilon/\varphi^2)}\cdot\sqrt{\frac{160}{s_{\min}}} \tag{3.170}$$

In the rest of the proof of step 1 we assume $S\subseteq C'$ which holds with probability at least $1-\frac{\delta}{2}$.

Therefore conditioned on $S \subseteq C'$ we have $\mathbb{E}[f_{x_i}] = \mu'$.

$$\|\widehat{\mu} - \mu'\|_2 = \left\| \sum_{i=1}^{s} \left(\frac{f_{x_i}}{s} - \mu' \right) \right\|_2 .$$

We define $\mathbf{z}_i = \frac{f_{x_i}}{s} - \frac{\mu'}{s}$, so $\|\widehat{\mu} - \mu'\|_2 = \|\sum_{i=1}^{s} \mathbf{z}_i\|_2$. Observe that $\mathbb{E}[\mathbf{z}_i] = \mathbb{E}\left[\frac{f_{x_i}}{s}\right] - \frac{\mu'}{s} = 0$, thus we can apply Lemma 3.20. Therefore we get

$$\mathbb{P}\left[\left|\left|\widehat{\mu} - \mu'\right|\right|_2 > q\right] = \mathbb{P}\left[\|\sum_{i=1}^{s} \mathbf{z}_i\|_2 > q\right] \leq (k+1) \cdot \exp\left(\frac{\frac{-q^2}{2}}{\sigma^2 + \frac{bq}{3}} \right), \tag{3.171}$$

where $\sigma^2 = \max\{\|\sum_{i=1}^{s} \mathbb{E}[\mathbf{z}_i \mathbf{z}_i^T]\|_2, \|\sum_{i=1}^{s} \mathbb{E}[\mathbf{z}_i^T \mathbf{z}_i]\|_2\}$ and b is an upper bound on $\|\mathbf{z}_i\|_2$ for all random variables $\mathbf{z}_i$. Therefore we need to upperbound $\|\mathbf{z}_i\|_2$ and σ^2. Note that

$$\|\mathbf{z}_i\|_2 = \left\| \frac{f_{x_i}}{s} - \frac{\mu'}{s} \right\|_2 \leq \left\| \frac{f_{x_i}}{s} \right\|_2 + \left\| \frac{\mu'}{s} \right\|_2 \leq \frac{\sqrt{k}}{s} \cdot \|f_{x_i}\|_\infty + \frac{1}{s} \cdot ||\mu'||_2 \tag{3.172}$$

Therefore by (3.169), (3.170) and (3.172) we have

$$\|\mathbf{z}_i\|_2 \leq \frac{\sqrt{k}}{s} \cdot \|f_{x_i}\|_\infty + \frac{1}{s} \cdot ||\mu'||_2 \leq \frac{2}{s} \cdot \left(\frac{s}{\delta}\right)^{(40 \cdot \epsilon/\varphi^2)} \cdot \sqrt{\frac{160 \cdot k}{s_{\min}}}, \tag{3.173}$$

Thus $b \leq \frac{2}{s} \cdot \left(\frac{s}{\delta}\right)^{(40 \cdot \epsilon/\varphi^2)} \cdot \sqrt{\frac{160 \cdot k}{s_{\min}}}$. We also need to upper bound σ^2. By (3.173) we get

$$\sigma^2 = \max\{\|\sum_{i=1}^{s} \mathbb{E}[\mathbf{z}_i \mathbf{z}_i^T]\|_2, \|\sum_{i=1}^{s} \mathbb{E}[\mathbf{z}_i^T \mathbf{z}_i]\|_2\} = s \cdot \mathbb{E}\left[\|\mathbf{z}_i\|_2^2\right] \leq s \cdot \frac{4}{s^2} \cdot \left(\frac{s}{\delta}\right)^{(80 \cdot \epsilon/\varphi^2)} \cdot \frac{160 \cdot k}{s_{\min}}. \tag{3.174}$$

We set $q = \frac{\zeta}{2} \cdot ||\mu||_2$. Having upper bound for σ^2 by (3.174) and on b by (3.173) we can apply Lemma 3.20 and we get

$$\begin{aligned} \mathbb{P}\left[\left|\left|\widehat{\mu} - \mu'\right|\right|_2 > \frac{\zeta}{2} \cdot ||\mu||_2\right] &\leq (k+1) \cdot \exp\left(\frac{\frac{-q^2}{2}}{\sigma^2 + \frac{bq}{3}} \right) \\ &\leq (k+1) \cdot \exp\left(\frac{\frac{-\zeta^2 \cdot ||\mu||_2^2}{8}}{\frac{640 \cdot k \cdot \left(\frac{s}{\delta}\right)^{(80 \cdot \epsilon/\varphi^2)}}{s \cdot s_{\min}} + \frac{\zeta}{2} \cdot ||\mu||_2 \cdot \frac{2 \cdot \left(\frac{s}{\delta}\right)^{(40 \cdot \epsilon/\varphi^2)}}{3 \cdot s} \sqrt{\frac{160 \cdot k}{s_{\min}}}} \right) \end{aligned} \tag{3.175}$$

By Lemma 3.7 for small enough $\frac{\epsilon}{\varphi^2}$ we have $\|\mu\|_2^2 \geq \frac{1}{2 \cdot |C|}$ and since $\min_{i,j} \frac{|C_i|}{|C_j|} \geq \Omega(1)$. Thus for a small enough constant c' we have

$$s_{\min} \cdot ||\mu||_2^2 \geq \frac{s_{\min}}{2 \cdot |C|} \geq c', \tag{3.176}$$

Thus by (3.176) and by choice of $s^{(1 - 80 \cdot \epsilon/\varphi^2)} \geq \frac{10^6}{c'} \cdot k \cdot \log\left(\frac{k}{\delta}\right) \cdot \left(\frac{1}{\delta}\right)^{(80 \cdot \epsilon/\varphi^2)} \cdot \left(\frac{1}{\zeta}\right)^2 \geq \frac{10^6 \cdot k \cdot \log\left(\frac{k}{\delta}\right) \cdot \left(\frac{1}{\delta}\right)^{(80 \cdot \epsilon/\varphi^2)} \cdot \left(\frac{1}{\zeta}\right)^2}{s_{\min} \cdot ||\mu||_2^2}$

we get

$$\frac{\zeta^2\cdot||\mu||_2^2}{8}\geq 400\cdot\log\left(\frac{k}{\delta}\right)\cdot\left(\frac{640\cdot k\cdot\left(\frac{s}{\delta}\right)^{(80\cdot\epsilon/\varphi^2)}}{s\cdot s_{\min}}\right) \tag{3.177}$$

and

$$\frac{\zeta^2\cdot||\mu||_2^2}{8}\geq 400\cdot\log\left(\frac{k}{\delta}\right)\left(\frac{\zeta}{2}\cdot||\mu||_2\cdot\frac{2\cdot\left(\frac{s}{\delta}\right)^{(80\cdot\epsilon/\varphi^2)}}{3\cdot s}\sqrt{\frac{160\cdot k}{s_{\min}}}\right) \tag{3.178}$$

Therefore since $s\geq c\cdot\left(k\cdot\log\left(\frac{k}{\delta}\right)\cdot\left(\frac{1}{\delta}\right)^{(80\cdot\epsilon/\varphi^2)}\cdot\left(\frac{1}{\zeta}\right)^2\right)^{1/(1-(80\cdot\epsilon/\varphi^2))}$ for large enough constant c, and putting (3.175), (3.177) and (3.178) together we get

$$\mathbb{P}\left[\left|\left|\widehat{\mu}-\mu'\right|\right|_2>\frac{\zeta}{2}\cdot||\mu||_2\right]\leq(k+1)\cdot e^{-200\cdot\log\left(\frac{k}{\delta}\right)}\leq\left(\frac{\delta}{k}\right)^{100}$$

Thus with probability at least $1-\frac{\delta}{2}-\left(\frac{\delta}{k}\right)^{100}\geq 1-\delta$ we have

$$\|\widehat{\mu}-\mu'\|_2\leq\frac{\zeta}{2}\cdot\|\mu\|_2. \tag{3.179}$$

Step 2: Next we want to bound $\|\mu-\mu'\|_2$. We have

$$\begin{aligned}\|\mu'-\mu\|_2&=\left|\left|\frac{1}{|C'|}\sum_{x\in C'}f_x-\frac{1}{|C|}\sum_{x\in C}f_x\right|\right|_2\\&\leq\left|\left|\frac{1}{|C'|}\sum_{x\in C}f_x-\frac{1}{|C|}\sum_{x\in C}f_x\right|\right|_2+\left|\left|\frac{1}{|C'|}\sum_{x\in C\setminus C'}f_x\right|\right|_2 &&\text{By triangle inequality}\\&\leq\left(\frac{1}{1-(k^{-1}\cdot s^{-2}\cdot\delta^2)}-1\right)||\mu||_2+\left|\left|\frac{1}{|C'|}\sum_{x\in C\setminus C'}f_x\right|\right|_2 &&\text{Since }|C'|\geq\left(1-(k^{-1}\cdot s^{-2}\cdot\delta^2)\right)|C|\text{ by (3.167)}\\&\leq 2\cdot(k^{-1}\cdot s^{-2}\cdot\delta^2)\cdot\|\mu\|_2+\left|\left|\frac{1}{|C'|}\sum_{x\in C\setminus C'}f_x\right|\right|_2 &&(3.180)\end{aligned}$$

It thus remains to upper bound the second term. We now note that

$$\left|\left|\frac{1}{|C'|}\sum_{x\in C\setminus C'}f_x\right|\right|_2\leq\frac{1}{|C'|}\sum_{x\in C\setminus C'}||f_x||_2\leq\frac{\sqrt{k}}{|C'|}\sum_{x\in C\setminus C'}\|f_x\|_\infty \tag{3.181}$$

For any $y\geq 1$ we define

$$T(y)=\left\{x\in V:||f_x||_\infty\geq 2\cdot y\cdot\left(\frac{s\cdot k}{\delta}\right)^{(40\cdot\epsilon/\varphi^2)}\cdot\sqrt{\frac{10}{s_{\min}}}\right\}$$

Therefore, by Lemma 3.38 we have

$$|T(y)| \leq k \cdot \left(\frac{2 \cdot y \cdot \left(\frac{s \cdot k}{\delta}\right)^{(40 \cdot \epsilon/\varphi^2)}}{2} \right)^{-\varphi^2/(20\cdot\epsilon)} \cdot s_{\min} = \left(\frac{s \cdot k}{\delta}\right)^{-2} \cdot y^{-\varphi^2/(20\cdot\epsilon)} \cdot s_{\min}. \tag{3.182}$$

Using the bound on $|T(y)|$ above, we now get

$$\begin{aligned}
&\sum_{x \in C \setminus C'} \|f_x\|_\infty && (3.183)\\
&\leq \int_1^\infty \left(y \cdot \left(\frac{s \cdot k}{\delta}\right)^{(40\cdot\epsilon/\varphi^2)} \cdot \sqrt{\frac{40}{s_{\min}}} \right) \cdot |T(y)| \cdot dy && \text{By definition of } T(y) \text{ and } C'\\
&\leq \sqrt{\frac{160}{s_{\min}}} \cdot \left(\frac{s}{\delta}\right)^{(40\cdot\epsilon/\varphi^2)} \cdot \int_1^\infty y \cdot |T(y)| \cdot dy && \text{Since } k^{(40\cdot\epsilon/\varphi^2)} \leq 2 \text{ for small enough } \frac{\epsilon \cdot \log k}{\varphi^2}\\
&\leq \sqrt{\frac{160}{s_{\min}}} \cdot \left(\frac{s}{\delta}\right)^{(40\cdot\epsilon/\varphi^2)} \cdot \int_1^\infty \left(\frac{s \cdot k}{\delta}\right)^{-2} \cdot y^{(1-\varphi^2/(20\cdot\epsilon))} \cdot s_{\min} \cdot dy && \text{By (3.182)}\\
&\leq \sqrt{\frac{160}{s_{\min}}} \cdot s_{\min} \cdot \left(\frac{s}{\delta}\right)^{(40\cdot\epsilon/\varphi^2)} \cdot \left(\frac{s \cdot k}{\delta}\right)^{-2} \frac{1}{\varphi^2/(20\cdot\epsilon) - 2} && \text{Since for any } c<0, \int_1^\infty y^c dy = \frac{-1}{c+1}\\
&\leq k^{-2} \cdot s^{-1} \cdot \sqrt{s_{\min}} && \text{For small enough } \frac{\epsilon}{\varphi^2} \quad (3.184)
\end{aligned}$$

Therefore we get

$$\begin{aligned}
\left\| \frac{1}{|C'|} \sum_{x \in C \setminus C'} f_x \right\|_2 &\leq \frac{\sqrt{k}}{|C'|} \sum_{x \in C \setminus C'} \|f_x\|_\infty && \text{By (3.181)}\\
&\leq \frac{\sqrt{k} \cdot k^{-2} \cdot s^{-1} \cdot \sqrt{s_{\min}}}{|C'|} && \text{By (3.184)}\\
&\leq \frac{2 \cdot k^{-1} \cdot s^{-1}}{\sqrt{|C|}} \cdot \frac{\sqrt{s_{\min}}}{\sqrt{|C|}} && \text{By (3.167)}\\
&\leq \frac{k^{-1} \cdot s^{-1}}{\sqrt{|C|}} && \text{Since } |C| \geq s_{\min}\\
&\leq 2 \cdot k^{-1} \cdot s^{-1} \cdot ||\mu||_2 && \text{By Lemma 3.7 } ||\mu||_2 \geq \frac{1}{2 \cdot \sqrt{|C|}}
\end{aligned}$$

Therefore by (3.180) we have

$$\|\mu' - \mu\|_2 \leq 2 \cdot (k^{-1} \cdot s^{-2} \cdot \delta^2) \|\mu\|_2 + \left\| \frac{1}{|C'|} \sum_{x \in C \setminus C'} f_x \right\|_2 \leq 2 \left(k^{-1} \cdot s^{-2} \cdot \delta^2 + \cdot k^{-1} \cdot s^{-1} \right) ||\mu||_2 \leq \frac{\zeta}{2} \cdot ||\mu||_2 \tag{3.185}$$

The last inequality holds since $s \geq 8 \cdot \left(\frac{1}{\zeta}\right)^2$, hence, $2\left(k^{-1} \cdot s^{-2} \cdot \delta^2 + \cdot k^{-1} \cdot s^{-1}\right) \leq \frac{\zeta}{2}$. Putting (3.168), (3.179) and (3.185) together with probability at least $1-\delta$ we get

$$||\widehat{\mu} - \mu||_2 \leq ||\widehat{\mu} - \mu'||_2 + ||\mu' - \mu||_2 \leq \frac{\zeta}{2} \cdot ||\mu||_2 + \frac{\zeta}{2} \cdot ||\mu||_2 \leq \zeta \cdot ||\mu||_2$$

□

To conclude our argument we show that if we sample enough nodes, we have a large number of samples in each cluster.

Lemma 3.40. *Let $k \geq 2$, $\varphi \in (0,1)$, $\frac{\epsilon \log k}{\varphi^2}$ be smaller than a positive sufficiently small constant. Let $G = (V, E)$ be a d-regular graph that admits a (k, φ, ϵ)-clustering $C_1, \dots, C_k$. Let S be the multiset of $s \in \Omega(k \log k)$ vertices each sampled independently at random from V. Then with probability at least $\frac{9}{10}$, for every $i \in [k]$,*

$$|S \cap C_i| \geq \frac{0.9 \cdot s}{k} \cdot \min_{p,q \in [k]} \frac{|C_p|}{|C_q|}.$$

Proof. For $i \in [k]$, and $1 \leq r \leq s$, let X_i^r be a random variable which is 1 if the r-th sampled vertex is in C_i, and 0 otherwise. Thus $\mathbb{E}[X_i^r] = \frac{|C_i|}{n}$. Observe that $|S \cap C_i|$ is a random variable defined as $\sum_{r=1}^{s} X_i^r$, where its expectation is given by

$$\mathbb{E}[|S \cap C_i|] = \sum_{r=1}^{s} \mathbb{E}[X_i^r] = s \cdot \frac{|C_i|}{n} \geq \frac{s \cdot s_{\min}}{k \cdot s_{\max}}.$$

Notice that random variables X_i^r are independent, Therefore, by Chernoff bound,

$$\Pr\left[|S \cap C_i| < \frac{9s}{10} \cdot \frac{|C_i|}{n}\right] \leq \exp\left(-\frac{1}{200} \cdot \frac{s \cdot s_{\min}}{k \cdot s_{\max}}\right).$$

By union bound and since $s = 500 \cdot k \cdot \log k \cdot \frac{s_{\max}}{s_{\min}}$ we have

$$\Pr\left[\exists i \colon |S \cap C_i| < \frac{9s}{10} \cdot \frac{|C_i|}{n}\right] \leq k \cdot \exp\left(-\frac{1}{200} \cdot \frac{s \cdot s_{\min}}{k \cdot s_{\max}}\right) \leq \frac{1}{10}.$$

Therefore with probability at least $\frac{9}{10}$ for all $i \in [k]$ we have

$$|S \cap C_i| \geq \frac{9 \cdot s}{10} \cdot \frac{|C_i|}{n} \geq \frac{0.9 \cdot s}{k} \cdot \frac{s_{\min}}{s_{\max}}$$

□

Approximate Centers are strongly orthogonal

The main result of this section is Lemma 3.41 that generalizes Lemma 3.12 to the approximate of cluster means.

Lemma 3.41. *Let $k \geq 2$ be an integer, $\varphi \in (0,1)$, and $\epsilon \in (0,1)$. Let $G = (V, E)$ be a d-regular graph that admits a (k, φ, ϵ)-clustering $C_1, \dots, C_k$. Let $0 < \zeta < \frac{\sqrt{\epsilon}}{20 \cdot k \cdot \varphi}$. Let $\mu_1, \dots, \mu_k$ denote the cluster means of $C_1, \dots, C_k$. Let $\widehat{\mu}_1, \dots, \widehat{\mu}_k \in \mathbb{R}^k$ denote an approximation of the cluster means such that* *for each $i \in [k]$, $||\mu_i - \widehat{\mu}_i||_2 \le \zeta ||\mu_i||_2$. Let $S \subset \{\widehat{\mu}_1, \dots, \widehat{\mu}_k\}$ denote a subset of cluster means. Let $\widehat{\Pi} \in \mathbb{R}^{k \times k}$ denote the orthogonal projection matrix into the $span(S)^{\perp}$. Then the following holds:*

1. *For all $\widehat{\mu}_i \in \{\widehat{\mu}_1, \dots, \widehat{\mu}_k\} \setminus S$ we have $\left| \|\widehat{\Pi}\widehat{\mu}_i\|_2^2 - ||\widehat{\mu_i}||_2^2 \right| \le \frac{20\sqrt{\epsilon}}{\varphi} \cdot ||\widehat{\mu_i}||_2^2$.*

2. *For all $\widehat{\mu}_i \ne \widehat{\mu}_j \in \{\widehat{\mu}_1, \dots, \widehat{\mu}_k\} \setminus S$ we have $|\langle \widehat{\Pi}\widehat{\mu}_i, \widehat{\Pi}\widehat{\mu}_j \rangle| \le \frac{50\sqrt{\epsilon}}{\varphi} \cdot \frac{1}{\sqrt{|C_i| \cdot |C_j|}}$.*

To prove Lemma 3.41 we use Lemma 3.30 from Section 3.4 and we prove Lemma 3.42.

Lemma 3.42. *Let $k \ge 2$ be an integer, $\varphi \in (0,1)$, and $\epsilon \in (0,1)$. Let $G = (V, E)$ be a d-regular graph that admits a (k, φ, ϵ)-clustering $C_1, \dots, C_k$. Let $0 < \zeta < \frac{\sqrt{\epsilon}}{20 \cdot k \cdot \varphi}$. Let $\widehat{\mu}_1, \dots, \widehat{\mu}_k \in \mathbb{R}^k$ denote an approximation of the cluster means such that for each $i \in [k]$, $||\mu_i - \widehat{\mu}_i||_2 \le \zeta ||\mu_i||_2$. Let $S = \{\widehat{\mu}_1, \dots, \widehat{\mu}_k\} \setminus \{\widehat{\mu}_i\}$. Let $\widehat{H} = [\widehat{\mu}_1, \widehat{\mu}_2, \dots, \widehat{\mu}_{i-1}, \widehat{\mu}_{i+1}, \dots, \widehat{\mu}_k]$ denote a matrix such that its columns are the vectors in S. Let $\widehat{W} \in \mathbb{R}^{(k-1)\times(k-1)}$ denote a diagonal matrix such that for all $j < i$ we have $\widehat{W}(j,j) = \sqrt{|C_j|}$ and for all $j \ge i$ we have $\widehat{W}(j,j) = \sqrt{|C_{j+1}|}$. Let $\widehat{Z} = \widehat{H}\widehat{W}$. Then we have*

$$\widehat{\mu}_i^T \widehat{Z} \widehat{Z}^T \widehat{\mu}_i \le \frac{10\sqrt{\epsilon}}{\varphi} \cdot ||\widehat{\mu}_i||_2^2.$$

Proof. Note that $\widehat{Z}\widehat{Z}^T = (\sum_{j=1}^k |C_j| \widehat{\mu}_j \widehat{\mu}_j^T) - |C_i| \widehat{\mu}_i \widehat{\mu}_i^T$. Thus we have

$$\widehat{\mu}_i^T \widehat{Z} \widehat{Z}^T \widehat{\mu}_i = \widehat{\mu}_i^T \left(\sum_{j=1}^k |C_j| \widehat{\mu}_j \widehat{\mu}_j^T \right) \widehat{\mu}_i - |C_i| \cdot ||\widehat{\mu}_i||_2^4. \tag{3.186}$$

By Lemma 3.9 for any vector x with $||x||_2 = 1$ we have

$$x^T \left(\sum_{j=1}^k |C_j| \mu_j \mu_j^T - I \right) x \le \frac{4\sqrt{\epsilon}}{\varphi} \tag{3.187}$$

Note that

$$\begin{aligned}
&||\sum_{j=1}^{k}|C_j|\widehat{\mu}_j\widehat{\mu}_j^T-\sum_{j=1}^{k}|C_j|\mu_j\mu_j^T||_2 && \\
&\le \sum_{j=1}^{k}|C_j|\cdot||\widehat{\mu}_j\widehat{\mu}_j^T-\mu_j\mu_j^T||_2 && \text{By triangle inequality}\\
&= \sum_{j=1}^{k}|C_j|\left(||\left(\mu_j+(\widehat{\mu}_j-\mu_j)\right)\left(\mu_j+(\widehat{\mu}_j-\mu_j)\right)^T-\mu_j\mu_j^T||_2\right) && \\
&\le \sum_{j=1}^{k}|C_j|\left(||\left(\widehat{\mu}_j-\mu_j\right)\left(\widehat{\mu}_j-\mu_j\right)^T||_2+||\mu_j\left(\widehat{\mu}_j-\mu_j\right)^T||_2+||\left(\widehat{\mu}_j-\mu_j\right)\mu_j^T||_2\right) && \text{By triangle inequality}\\
&\le \sum_{j=1}^{k}|C_j|\cdot(\zeta^2+2\zeta)\cdot||\mu_j||_2^2 && \text{Since } ||\widehat{\mu}_j-\mu_j||_2\le\zeta||\mu_j||_2\\
&\le \sum_{j=1}^{k}|C_j|\cdot 6\cdot\zeta\cdot\frac{1}{|C_j|} && \text{By Lemma 3.7 } ||\mu_j||_2^2\le\frac{2}{|C_i|}\\
&\le 6\cdot\zeta\cdot k && \\
&\le \frac{\sqrt{\epsilon}}{2\varphi} && \text{Since } \zeta\le\frac{\sqrt{\epsilon}}{20\cdot k\cdot\varphi}
\end{aligned}$$

Thus for any vector x with $||x||_2 = 1$ we have

$$x^T\left(\sum_{j=1}^{k}|C_j|\widehat{\mu}_j\widehat{\mu}_j^T-\sum_{j=1}^{k}|C_j|\mu_j\mu_j^T\right)x\le\frac{\sqrt{\epsilon}}{2\varphi} \tag{3.188}$$

Putting (3.188) and (3.187) for any vector any vector x with $||x||_2 = 1$ we have that

$$x^T\left(\sum_{j=1}^{k}|C_j|\widehat{\mu}_j\widehat{\mu}_j^T-I\right)x\le\frac{5\sqrt{\epsilon}}{\varphi}$$

Hence we can write

$$\widehat{\mu}_i^T\left(\sum_{j=1}^{k}|C_j|\widehat{\mu}_j\widehat{\mu}_j^T\right)\widehat{\mu}_i=\widehat{\mu}_i^T\left(\sum_{j=1}^{k}|C_j|\widehat{\mu}_j\widehat{\mu}_j^T-I\right)\widehat{\mu}_i+\widehat{\mu}_i^T\widehat{\mu}_i\le\left(1+\frac{5\sqrt{\epsilon}}{\varphi}\right)||\widehat{\mu}_i||_2^2$$

Therefore by (3.186) we get

$$\widehat{\mu}_i^T\widehat{Z}\widehat{Z}^T\widehat{\mu}_i=\widehat{\mu}_i^T\left(\sum_{j=1}^{k}|C_j|\widehat{\mu}_j\widehat{\mu}_j^T\right)\widehat{\mu}_i-|C_i|\cdot||\widehat{\mu}_i||_2^4\le\left(1+\frac{5\sqrt{\epsilon}}{\varphi}-|C_i|\cdot||\widehat{\mu}_i||_2^2\right)||\widehat{\mu}_i||_2^2$$

By Lemma 3.7, and since $||\widehat{\mu_i}||\ge(1-\zeta)||\mu_i||_2$ and $\zeta\le\frac{\sqrt{\epsilon}}{20\cdot k\cdot\varphi}$ we have that

$$|C_i|\cdot||\widehat{\mu}_i||_2^2\ge\left(1-\frac{4\sqrt{\epsilon}}{\varphi}\right)(1-\zeta)^2\ge1-\frac{5\sqrt{\epsilon}}{\varphi}$$

Thus we get

$$\widehat{\mu}_i^T \widehat{Z}\widehat{Z}^T \widehat{\mu}_i \leq \left(1 + \frac{5\sqrt{\epsilon}}{\varphi} - |C_i| \cdot ||\widehat{\mu}_i||_2^2\right) ||\widehat{\mu}_i||_2^2 \leq \left(1 + \frac{5\sqrt{\epsilon}}{\varphi} - 1 + \frac{5\sqrt{\epsilon}}{\varphi}\right) ||\widehat{\mu}_i||_2^2 \leq \frac{10\sqrt{\epsilon}}{\varphi} \cdot ||\widehat{\mu}_i||_2^2$$

□

We now prove the main result of this section (Lemma 3.41).

Lemma 3.41. *Let $k \geq 2$ be an integer, $\varphi \in (0,1)$, and $\epsilon \in (0,1)$. Let $G = (V, E)$ be a d-regular graph that admits a (k, φ, ϵ)-clustering $C_1, \ldots, C_k$. Let $0 < \zeta < \frac{\sqrt{\epsilon}}{20 \cdot k \cdot \varphi}$. Let $\mu_1, \ldots, \mu_k$ denote the cluster means of $C_1, \ldots, C_k$. Let $\widehat{\mu}_1, \ldots, \widehat{\mu}_k \in \mathbb{R}^k$ denote an approximation of the cluster means such that for each $i \in [k]$, $||\mu_i - \widehat{\mu}_i||_2 \leq \zeta ||\mu_i||_2$. Let $S \subset \{\widehat{\mu}_1, \ldots, \widehat{\mu}_k\}$ denote a subset of cluster means. Let $\widehat{\Pi} \in \mathbb{R}^{k \times k}$ denote the orthogonal projection matrix into the $span(S)^\perp$. Then the following holds:*

1. *For all $\widehat{\mu}_i \in \{\widehat{\mu}_1, \ldots, \widehat{\mu}_k\} \setminus S$ we have $\left| ||\widehat{\Pi}\widehat{\mu}_i||_2^2 - ||\widehat{\mu_i}||_2^2 \right| \leq \frac{20\sqrt{\epsilon}}{\varphi} \cdot ||\widehat{\mu_i}||_2^2$.*

2. *For all $\widehat{\mu}_i \neq \widehat{\mu}_j \in \{\widehat{\mu}_1, \ldots, \widehat{\mu}_k\} \setminus S$ we have $|\langle \widehat{\Pi}\widehat{\mu}_i, \widehat{\Pi}\widehat{\mu}_j \rangle| \leq \frac{50\sqrt{\epsilon}}{\varphi} \cdot \frac{1}{\sqrt{|C_i| \cdot |C_j|}}$.*

Proof. **Proof of item** (1)**:** Since $\widehat{\Pi}$ is a orthogonal projection matrix we have $||\widehat{\Pi}||_2 = 1$. Hence, we have

$$||\widehat{\Pi}\widehat{\mu}_i||_2^2 \leq ||\widehat{\mu}_i||_2^2 \leq \left(1 + \frac{20\sqrt{\epsilon}}{\varphi}\right) ||\widehat{\mu}_i||_2^2.$$

Thus it's left to prove $||\widehat{\Pi}\widehat{\mu}_i||_2^2 \geq \left(1 - \frac{20\sqrt{\epsilon}}{\varphi}\right) ||\widehat{\mu}_i||_2^2$. Note that by Pythagoras $||\widehat{\Pi}\widehat{\mu}_i||_2^2 = ||\widehat{\mu}_i||_2^2 - ||(I - \widehat{\Pi})\widehat{\mu}_i||_2^2$. We will prove $||(I - \widehat{\Pi})\widehat{\mu}_i||_2^2 \leq \frac{20\sqrt{\epsilon}}{\varphi} ||\widehat{\mu}_i||_2^2$ which implies

$$||\widehat{\Pi}\widehat{\mu}_i||_2^2 \geq \left(1 - 20\frac{\sqrt{\epsilon}}{\varphi}\right) ||\widehat{\mu}_i||_2^2.$$

Thus in order to complete the proof we need to show $||(I - \widehat{\Pi})\widehat{\mu}_i||_2^2 \leq \frac{20\sqrt{\epsilon}}{\varphi} ||\widehat{\mu}_i||_2^2$. Let $S' = \{\widehat{\mu}_1, \ldots, \widehat{\mu}_k\} \setminus \{\widehat{\mu}_i\}$. Let $\widehat{\Pi}'$ denote the orthogonal projection matrix into $span(S')^\perp$. Note that $S \subseteq S'$, hence $span(S)$ is a subspace of $span(S')$, therefore we have $||(I - \widehat{\Pi})\widehat{\mu}_i||_2^2 \leq ||(I - \widehat{\Pi}')\widehat{\mu}_i||_2^2$. Thus it suffices to prove $||(I - \widehat{\Pi}')\widehat{\mu}_i||_2^2 \leq \frac{20\sqrt{\epsilon}}{\varphi} ||\widehat{\mu}_i||_2^2$. Let $\widehat{H} = [\widehat{\mu}_1, \widehat{\mu}_2, \ldots, \widehat{\mu}_{i-1}, \widehat{\mu}_{i+1}, \ldots, \widehat{\mu}_k]$ denote a matrix such that its columns are the vectors in S'. Let $\widehat{W} \in \mathbb{R}^{(k-1) \times (k-1)}$ denote a diagonal matrix such that for all $j < i$ we have $\widehat{W}(j, j) = \sqrt{|C_j|}$ and for all $j \geq i$ we have $\widehat{W}(j, j) = \sqrt{|C_{j+1}|}$. Let $\widehat{Z} = \widehat{H}\widehat{W}$. Then the orthogonal projection matrix onto the span of S' is defined as $(I - \widehat{\Pi}') = \widehat{Z}(\widehat{Z}^T\widehat{Z})^{-1}\widehat{Z}^T$. By Lemma 3.30 item (2), $(\widehat{Z}^T\widehat{Z})^{-1}$ is spectrally close to I, hence, $(\widehat{Z}^T\widehat{Z})^{-1}$ exists. Therefore we have

$$\begin{aligned} ||(I - \widehat{\Pi}')\widehat{\mu}_i||_2^2 &= \widehat{\mu}_i^T \widehat{Z}(\widehat{Z}^T\widehat{Z})^{-1}\widehat{Z}^T\widehat{\mu}_i \\ &= \widehat{\mu}_i^T \widehat{Z}((\widehat{Z}^T\widehat{Z})^{-1} - I)\widehat{Z}^T\widehat{\mu}_i + \widehat{\mu}_i^T \widehat{Z}\widehat{Z}^T\widehat{\mu}_i \end{aligned} \tag{3.189}$$

By Lemma 3.30 item (2) we have

$$\left|\widehat{\mu}_i^T \widehat{Z}\left((\widehat{Z}^T\widehat{Z})^{-1}-I\right)\widehat{Z}^T\widehat{\mu}_i\right| \le \frac{5\sqrt{\epsilon}}{\varphi}||\widehat{Z}^T\widehat{\mu}_i||_2^2 \tag{3.190}$$

Thus we get

$$\begin{aligned}
||(I-\widehat{\Pi}')\widehat{\mu}_i||_2^2 &\le \widehat{\mu}_i^T\widehat{Z}((\widehat{Z}^T\widehat{Z})^{-1}-I)\widehat{Z}^T\widehat{\mu}_i + \widehat{\mu}_i^T\widehat{Z}\widehat{Z}^T\widehat{\mu}_i && \text{By (3.189)}\\
&\le \left(\frac{5\sqrt{\epsilon}}{\varphi}+1\right)||\widehat{Z}^T\widehat{\mu}_i||_2^2 && \text{By (3.190)}\\
&\le 2\cdot||\widehat{Z}^T\widehat{\mu}_i||_2^2 && \text{For small enough } \frac{\epsilon}{\varphi^2}
\end{aligned}$$

By Lemma 3.42 we have

$$||\widehat{Z}^T\widehat{\mu}_i||_2^2 = \widehat{\mu}_i^T\widehat{Z}\widehat{Z}^T\widehat{\mu}_i \le \frac{10\sqrt{\epsilon}}{\varphi}\cdot||\widehat{\mu}_i||_2^2$$

Therefore we get

$$||(I-\widehat{\Pi})\widehat{\mu}_i||_2^2 \le ||(I-\widehat{\Pi}')\widehat{\mu}_i||_2^2 \le 2||\widehat{Z}^T\widehat{\mu}_i||_2^2 \le \frac{20\sqrt{\epsilon}}{\varphi}||\widehat{\mu}_i||_2^2 \tag{3.191}$$

Hence,

$$||\widehat{\Pi}\widehat{\mu}_i||_2^2 \ge \left(1-20\frac{\sqrt{\epsilon}}{\varphi}||\widehat{\mu}_i||_2^2\right).$$

Proof of item (2)**:** Note that

$$\langle\widehat{\mu}_i,\widehat{\mu}_j\rangle = \langle(I-\widehat{\Pi})\widehat{\mu}_i+\widehat{\Pi}\widehat{\mu}_i,(I-\widehat{\Pi})\widehat{\mu}_j+\widehat{\Pi}\widehat{\mu}_j\rangle = \langle(I-\widehat{\Pi})\widehat{\mu}_i,(I-\widehat{\Pi})\widehat{\mu}_j\rangle + \langle\widehat{\Pi}\widehat{\mu}_i,\widehat{\Pi}\widehat{\mu}_j\rangle$$

Thus by triangle inequality we have

$$|\langle\widehat{\Pi}\widehat{\mu}_i,\widehat{\Pi}\widehat{\mu}_j\rangle| \le |\langle\widehat{\mu}_i,\widehat{\mu}_j\rangle| + |\langle(I-\widehat{\Pi})\widehat{\mu}_i,(I-\widehat{\Pi})\widehat{\mu}_j\rangle|$$

By Cauchy-Schwarz we have

$$\begin{aligned}
|\langle(I-\widehat{\Pi})\widehat{\mu}_i,(I-\widehat{\Pi})\widehat{\mu}_j\rangle| &\le ||(I-\widehat{\Pi})\widehat{\mu}_i||_2||(I-\widehat{\Pi})\widehat{\mu}_i||_2\\
&\le \frac{20\sqrt{\epsilon}}{\varphi}||\widehat{\mu}_i||_2||\widehat{\mu}_j||_2 && \text{By (3.191)}\\
&\le \frac{40\sqrt{\epsilon}}{\varphi}\cdot\frac{1}{\sqrt{|C_i||C_j|}} && \text{By Lemma 3.7 and } ||\widehat{\mu_i}-\mu_i||_2 \le \zeta||\mu_i||_2
\end{aligned}$$

Also for any $i, j \in [k]$ we have

$$\begin{aligned}
&\left|\left\langle \widehat{\mu}_i, \widehat{\mu}_j \right\rangle - \left\langle \mu_i, \mu_j \right\rangle\right| && \\
&= \left|\left\langle \mu_i + (\widehat{\mu}_i - \mu_i), \mu_j + (\widehat{\mu}_j - \mu_j) \right\rangle - \left\langle \mu_i, \mu_j \right\rangle\right| && \\
&\le |\left\langle \widehat{\mu}_i - \mu_i, \widehat{\mu}_j - \mu_j \right\rangle| + |\left\langle \widehat{\mu}_i - \mu_i, \mu_j \right\rangle| + |\left\langle \widehat{\mu}_j - \mu_j, \mu_i \right\rangle| && \text{By triangle inequality} \\
&\le \ ||\widehat{\mu}_i - \mu_i||_2 ||\widehat{\mu}_j - \mu_j||_2 + ||\widehat{\mu}_i - \mu_i||_2 ||\mu_j||_2 + ||\widehat{\mu}_j - \mu_j||_2 ||\mu_i||_2 && \text{By Cauchy-Schwarz} \\
&\le (\zeta^2 + 2\zeta) \left(||\mu_i||_2 ||\mu_j||_2\right) && \text{Since } ||\widehat{\mu_i} - \mu_i||_2 \le \zeta ||\mu_i||_2 \text{ for all } i \\
&\le 6 \cdot \zeta \cdot \frac{1}{\sqrt{|C_i||C_j|}} && \text{By Lemma 3.7 } ||\mu_i||_2^2 \le \frac{2}{|C_i|} \text{ for all } i
\end{aligned} \tag{3.192}$$

Note that

$$\begin{aligned}
|\left\langle \widehat{\mu}_i, \widehat{\mu}_j \right\rangle| &\le |\left\langle \mu_i, \mu_j \right\rangle| + |\left\langle \mu_i, \mu_j \right\rangle - \left\langle \widehat{\mu}_i, \widehat{\mu}_j \right\rangle| && \text{By triangle inequality} \\
&\le \frac{8\sqrt{\epsilon}}{\varphi} \cdot \frac{1}{\sqrt{|C_i||C_j|}} + 6\zeta \cdot \frac{1}{\sqrt{|C_i||C_j|}} && \text{By Lemma 3.7 and (3.192)} \\
&\le \frac{10\sqrt{\epsilon}}{\varphi} \frac{1}{\sqrt{|C_i||C_j|}} && \text{Since } \zeta \le \frac{\sqrt{\epsilon}}{20 \cdot k \cdot \varphi}
\end{aligned}$$

Therefore we get

$$|\langle \widehat{\Pi} \widehat{\mu}_i, \widehat{\Pi} \widehat{\mu}_j \rangle| \le |\langle \widehat{\mu}_i, \widehat{\mu}_j \rangle| + |\langle (I - \widehat{\Pi}) \widehat{\mu}_i, (I - \widehat{\Pi}) \widehat{\mu}_j \rangle| \le \frac{50\sqrt{\epsilon}}{\varphi} \cdot \frac{1}{\sqrt{|C_i||C_j|}}.$$

□

3.6.5 Partitioning scheme works with *approximate* cluster means & dot products

In Section 3.6.3 we showed that the partitioning scheme works if we have access to real centers (i.e. $\mu_1,\dots,\mu_k$), to exact dot product evaluations (i.e $\langle\cdot,\cdot\rangle$) and OUTERCONDUCTANCE is precise.

In this section we show that approximations to all above is enough for the partitioning scheme to work. More precisely we show that if we have access only to $\langle\cdot,\cdot\rangle_{apx} \approx \langle\cdot,\cdot\rangle$, the search procedure finds $\hat{\mu}_i$'s that are only approximately equal to μ_i's and OUTERCONDUCTANCE is only approximately correct then FINDCENTERS still succeeds with high probability.

In order to prove such a statement we first show a technical Lemma (*Lemma 3.43*), that relates the approximate dot product with approximate centers to the dot product with the actual cluster centers.

Note that the following Lemma 3.43 works for any $S \subset \{\mu_1,\dots,\mu_k\}$ and the corresponding $\widehat{S}$. This is useful for application in Lemma 3.45 because it allows to reason about candidate sets $\widehat{C}^{(}T_1,\dots,T_b)_{\widehat{\mu}}$, after we associate $\bigcup_{i\in[b]} T_i$ with $\widehat{S}$.

Lemma 3.43. *Let $k \geq 2$, $\varphi \in (0,1)$, $\frac{\epsilon}{\varphi^2}$ be smaller than a sufficiently small constant. Let $G=(V,E)$ be a d-regular graph that admits a (k,φ,ϵ)-clustering $C_1,\dots,C_k$. Then conditioned on the success of the spectral dot product oracle the following conditions hold.*

Let $\widehat{\mu}_1,\widehat{\mu}_2,\dots,\widehat{\mu}_k$ be such that for all $i\in[k]$ $\|\widehat{\mu}_i-\mu_i\|^2 \leq 10^{-12}\cdot\frac{\epsilon}{\varphi^2\cdot k^2}\|\mu_i\|^2$. Let $i\in[k]$ and $S\subseteq\{\mu_1,\dots,\mu_k\}\setminus\{\mu_i\}$ and $\widehat{S}\subseteq\{\widehat{\mu}_1,\dots,\widehat{\mu}_k\}\setminus\{\widehat{\mu}_i\}$ be the corresponding subset to S. Let Π be the orthogonal projection onto $span(S)^\perp$ and $\widehat{\Pi}$ be the orthogonal projection onto $span(\widehat{S})^\perp$. Let also $\pi_i:\mathbb{R}^k\to\mathbb{R}^k$ be the projection onto the subspace spanned by $\Pi\mu_i$ and $\widehat{\Pi}\widehat{\mu}_i$. Then if $\|\Pi_i f_x\|^2 \leq \frac{10^4}{\min_{p\in[k]}|C_p|}$ then:

$$\left|\frac{\langle f_x,\Pi\mu_i\rangle}{\|\Pi\mu_i\|^2}-\frac{\left\langle f_x,\widehat{\Pi}\widehat{\mu}_i\right\rangle_{apx}}{\left\|\widehat{\Pi}\widehat{\mu}_i\right\|^2_{apx}}\right| \leq 0.02$$

Furthermore if $\widehat{\mu}_i$'s are averages of s points, then $\frac{\langle f_x,\widehat{\Pi}\widehat{\mu}_i\rangle_{apx}}{\|\widehat{\Pi}\widehat{\mu}_i\|^2_{apx}}$ can be computed in $\widetilde{O}_\varphi\left(s^4\cdot\left(\frac{k}{\epsilon}\right)^{O(1)}\cdot n^{1/2+O(\epsilon/\varphi^2)}\right)$ time with preprocessing time of $\widetilde{O}_\varphi\left(\left(\frac{k}{\epsilon}\right)^{O(1)}\cdot n^{1/2+O(\epsilon/\varphi^2)}\right)$ and space $\widetilde{O}_\varphi\left(\left(\frac{k}{\epsilon}\right)^{O(1)}\cdot n^{1/2+O(\epsilon/\varphi^2)}\right)$

Proof. First we prove the runtime guarantee and then we show correctness.

Runtime. We first bound the running time. If we set the precision parameter of Algorithm 6 to $\xi = 10^{-6}\cdot\frac{\sqrt{\epsilon}}{\varphi}$ then by Theorem 3.2 the preprocessing time takes $\widetilde{O}_\varphi\left(\left(\frac{k}{\epsilon}\right)^{O(1)}\cdot n^{1/2+O(\epsilon/\varphi^2)}\right)$ time, $\widetilde{O}_\varphi\left(\left(\frac{k}{\epsilon}\right)^{O(1)}\cdot n^{1/2+O(\epsilon/\varphi^2)}\right)$ space, and by Corollary 3.1 computing $\frac{\langle f_x,\widehat{\Pi}\widehat{\mu}_i\rangle_{apx}}{\|\widehat{\Pi}\widehat{\mu}_i\|^2_{apx}}$ takes $\widetilde{O}_\varphi\left(s^4\cdot\left(\frac{k}{\epsilon}\right)^{O(1)}\cdot n^{1/2+O(\epsilon/\varphi^2)}\right)$ time.

Correctness. Now we show that we also obtain a good approximation. We will show it in two steps:

1. $\left|\frac{\langle f_x, \Pi\mu_i\rangle}{\|\Pi\mu_i\|^2} - \frac{\langle f_x, \widehat{\Pi}\widehat{\mu}_i\rangle}{||\widehat{\Pi}\widehat{\mu}_i||^2}\right| \leq 0.01$

2. $\left|\frac{\langle f_x, \widehat{\Pi}\widehat{\mu}_i\rangle}{\|\widehat{\Pi}\widehat{\mu}_i\|^2} - \frac{\langle f_x, \widehat{\Pi}\widehat{\mu}_i\rangle_{apx}}{\|\widehat{\Pi}\widehat{\mu}_i\|^2_{apx}}\right| \leq 0.01$

If we are able to prove 1 and 2 then the claim of the Lemma follows from triangle inequality.

Before we present the two proofs we show a useful fact:

$$\begin{aligned}
||\widehat{\Pi}\widehat{\mu}_i - \Pi\mu_i|| &\leq ||\widehat{\Pi}\widehat{\mu}_i - \hat{\mu}_i|| + ||\Pi\mu_i - \mu_i|| + ||\widehat{\mu}_i - \mu_i|| && \text{By triangle inequality} \\
&\leq \frac{20\epsilon^{1/4}}{\sqrt{\varphi}}||\hat{\mu}_i|| + \frac{16\epsilon^{1/4}}{\sqrt{\varphi}}||\mu_i|| + 10^{-6}\cdot\frac{\sqrt{\epsilon}}{\varphi\cdot k}\|\mu_i\| && \text{By Lemma 3.41, 3.12 and the bound on } \|\widehat{\mu}_i - \mu_i\|^2 \\
&\leq \frac{40\epsilon^{1/4}}{\sqrt{\varphi}}||\mu_i|| && \text{As } \|\widehat{\mu}_i - \mu_i\|^2 \leq 10^{-12}\cdot\frac{\epsilon}{\varphi^2\cdot k^2}\|\mu_i\|^2
\end{aligned} \tag{3.193}$$

Proof of 1: Notice that

$$\begin{aligned}
\left|\frac{\langle f_x, \Pi\mu_i\rangle}{||\Pi\mu_i||^2} - \frac{\langle f_x, \widehat{\Pi}\widehat{\mu}_i\rangle}{||\widehat{\Pi}\widehat{\mu}_i||^2}\right| &= \left|\left\langle f_x, \frac{\Pi\mu_i}{||\Pi\mu_i||^2} - \frac{\widehat{\Pi}\widehat{\mu}_i}{||\widehat{\Pi}\widehat{\mu}_i||^2}\right\rangle\right| \\
&= \left|\left\langle \Pi_i f_x, \frac{\Pi\mu_i}{||\Pi\mu_i||^2} - \frac{\widehat{\Pi}\widehat{\mu}_i}{||\widehat{\Pi}\widehat{\mu}_i||^2}\right\rangle\right| && \text{By definition of } \pi_i \\
&\leq ||\Pi_i f_x||\left|\left|\frac{\Pi\mu_i}{||\Pi\mu_i||^2} - \frac{\widehat{\Pi}\widehat{\mu}_i}{||\widehat{\Pi}\widehat{\mu}_i||^2}\right|\right| && \text{By Cauchy-Schwarz}
\end{aligned} \tag{3.194}$$

First we will upper bound $\left|\left|\frac{\Pi\mu_i}{||\Pi\mu_i||^2} - \frac{\widehat{\Pi}\widehat{\mu}_i}{||\widehat{\Pi}\widehat{\mu}_i||^2}\right|\right|$. We split it into two cases:

Case 1. If $\frac{\Pi\mu_i}{||\Pi\mu_i||^2} \geq \frac{\widehat{\Pi}\widehat{\mu}_i}{||\widehat{\Pi}\widehat{\mu}_i||^2}$ then we have:

$$\begin{aligned}
\left|\left|\frac{\Pi\mu_i}{||\Pi\mu_i||^2} - \frac{\widehat{\Pi}\widehat{\mu}_i}{||\widehat{\Pi}\widehat{\mu}_i||^2}\right|\right| &\leq \left|\left|\frac{\Pi\mu_i}{(1-\frac{16\sqrt{\epsilon}}{\varphi})||\mu_i||^2} - \frac{\widehat{\Pi}\widehat{\mu}_i}{||\widehat{\Pi}\widehat{\mu}_i||^2}\right|\right| && \text{By Lemma 3.12} \\
&\leq \left|\left|\frac{\Pi\mu_i}{(1-\frac{16\sqrt{\epsilon}}{\varphi})||\mu_i||^2} - \frac{\widehat{\Pi}\widehat{\mu}_i}{(1+\frac{20\sqrt{\epsilon}}{\varphi})(1+10^{-12}\cdot\frac{\epsilon}{\varphi^2\cdot k^2})||\mu_i||^2}\right|\right| && \text{Lemma 3.41, assumptions} \\
&\leq \frac{2}{||\mu_i||^2}\left|\left|\Pi\mu_i - \left(1-\frac{1600\sqrt{\epsilon}}{\varphi}\right)\widehat{\Pi}\widehat{\mu}_i\right|\right| \\
&\leq \frac{2}{||\mu_i||^2}\left(\left|\left|\frac{1600\sqrt{\epsilon}}{\varphi}\Pi\mu_i\right|\right| + \left(1-\frac{1600\sqrt{\epsilon}}{\varphi}\right)||\widehat{\Pi}\widehat{\mu}_i - \Pi\mu_i||\right) && \text{By triangle inequality} \\
&\leq \frac{12800\sqrt{\epsilon}}{\varphi}\frac{1}{||\mu_i||} && \text{By (3.193) and Lemma 3.12}
\end{aligned}$$

Case 2. If $\frac{\Pi\mu_i}{||\Pi\mu_i||^2} < \frac{\widehat{\Pi}\widehat{\mu}_i}{||\widehat{\Pi}\widehat{\mu}_i||^2}$ then we have:

$$\begin{aligned}
\left|\left|\frac{\Pi\mu_i}{||\Pi\mu_i||^2} - \frac{\widehat{\Pi}\widehat{\mu}_i}{||\widehat{\Pi}\widehat{\mu}_i||^2}\right|\right| &\leq \left|\left|\frac{\Pi\mu_i}{(1+\frac{16\sqrt{\epsilon}}{\varphi})||\mu_i||^2} - \frac{\widehat{\Pi}\widehat{\mu}_i}{||\widehat{\Pi}\widehat{\mu}_i||^2}\right|\right| && \text{By Lemma 3.12}\\
&\leq \left|\left|\frac{\Pi\mu_i}{(1+\frac{16\sqrt{\epsilon}}{\varphi})||\mu_i||^2} - \frac{\widehat{\Pi}\widehat{\mu}_i}{(1-\frac{20\sqrt{\epsilon}}{\varphi})(1-10^{-12}\cdot\frac{\epsilon}{\varphi^2\cdot k^2})||\mu_i||^2}\right|\right| && \text{Lemma 3.41, assumptions}\\
&\leq \frac{2}{||\mu_i||^2}\left|\left|\Pi\mu_i - \left(1+\frac{1600\sqrt{\epsilon}}{\varphi}\right)\widehat{\Pi}\widehat{\mu}_i\right|\right|\\
&\leq \frac{2}{||\mu_i||^2}\left(\left|\left|\frac{1600\sqrt{\epsilon}}{\varphi}\Pi\mu_i\right|\right| + \left(1+\frac{1600\sqrt{\epsilon}}{\varphi}\right)||\widehat{\Pi}\widehat{\mu}_i - \Pi\mu_i||\right) && \text{By triangle inequality}\\
&\leq \frac{12800\sqrt{\epsilon}}{\varphi}\frac{1}{||\mu_i||} && \text{By (3.193) and Lemma 3.12}
\end{aligned}$$

Combining the two cases we get:

$$\left|\left|\frac{\Pi\mu_i}{||\Pi\mu_i||^2} - \frac{\widehat{\Pi}\widehat{\mu}_i}{||\widehat{\Pi}\widehat{\mu}_i||^2}\right|\right| \leq \frac{12800\sqrt{\epsilon}}{\varphi}\frac{1}{||\mu_i||}.$$

Substituting into (3.194) we get:

$$\begin{aligned}
\left|\frac{\langle f_x, \Pi\mu_i\rangle}{||\Pi\mu_i||^2} - \frac{\langle f_x, \widehat{\Pi}\widehat{\mu}_i\rangle}{||\widehat{\Pi}\widehat{\mu}_i||^2}\right| &\leq ||\Pi_i f_x||\cdot\frac{12800\sqrt{\epsilon}}{\varphi}\frac{1}{||\mu_i||}\\
&\leq \frac{100}{\sqrt{\min_{p\in[k]}|C_p|}}\cdot\frac{12800\sqrt{\epsilon}}{\varphi}\frac{1}{||\mu_i||} && \text{By assumption of the Lemma}\\
&\leq 0.005\frac{1}{\sqrt{\max_{p\in[k]}|C_p|}\cdot||\mu_i||} && \text{As } \frac{\epsilon}{\varphi^2} \text{ is sufficiently small and } \frac{\max_{p\in[k]}|C_p|}{\min_{p\in[k]}|C_p|} = O(1\\
&\leq 0.01 && \text{By Lemma 3.7}
\end{aligned}$$

Proof of 2:

$$
\begin{aligned}
\left\|\widehat{\Pi}\widehat{\mu}_i\right\|_{apx}^2 &\geq ||\widehat{\Pi}\widehat{\mu}_i||^2 - 10^{-6}\cdot\frac{\sqrt{\epsilon}}{\varphi}\cdot n^{-1} && \text{By Corollary 3.1, setting of } \xi \text{ and assumptions}\\
&\geq \left(1-\frac{20\sqrt{\epsilon}}{\varphi}\right)\cdot||\widehat{\mu}_i||^2 - 0.01\cdot n^{-1} && \text{By Lemma 3.41 and } \frac{\epsilon}{\varphi^2} \text{ small}\\
&\geq \left(1-10^{-12}\frac{\epsilon}{\varphi^2\cdot k}\right)\cdot 0.99\cdot||\mu_i||^2 - 0.01\cdot n^{-1} && \text{By } ||\widehat{\mu}_i-\mu_i||^2 \leq 10^{-12}\frac{\epsilon}{\varphi^2\cdot k}||\mu_i||^2 \text{ and } \frac{\epsilon}{\varphi^2} \text{ small}\\
&\geq \left(1-\frac{4\sqrt{\epsilon}}{\varphi}\right)\cdot 0.98\cdot n^{-1} - 0.01\cdot n^{-1} && \text{By Lemma 3.7, } |C_i|\leq n,\ \frac{\epsilon}{\varphi^2} \text{ small}\\
&\geq 0.5\cdot n^{-1} && \text{As } \frac{\epsilon}{\varphi^2} \text{ small} \qquad (3.195)
\end{aligned}
$$

Next notice that:

$$
\begin{aligned}
\left|\frac{\langle f_x,\widehat{\Pi}\widehat{\mu}_i\rangle}{\|\widehat{\Pi}\widehat{\mu}_i\|^2} - \frac{\left\langle f_x,\widehat{\Pi}\widehat{\mu}_i\right\rangle_{apx}}{\left\|\widehat{\Pi}\widehat{\mu}_i\right\|_{apx}^2}\right| &\leq \left|\frac{\langle f_x,\widehat{\Pi}\widehat{\mu}_i\rangle}{\|\widehat{\Pi}\widehat{\mu}_i\|^2} - \frac{\left\langle f_x,\widehat{\Pi}\widehat{\mu}_i\right\rangle}{\left\|\widehat{\Pi}\widehat{\mu}_i\right\|_{apx}^2}\right| + \left|\frac{10^{-6}\cdot\frac{\sqrt{\epsilon}}{\varphi}\cdot n^{-1}}{\left\|\widehat{\Pi}\widehat{\mu}_i\right\|_{apx}^2}\right| && \text{By Corollary 3.1}\\
&\leq \left|\left\langle f_x,\widehat{\Pi}\widehat{\mu}_i\right\rangle\left(\frac{1}{\|\widehat{\Pi}\widehat{\mu}_i\|^2}-\frac{1}{\left\|\widehat{\Pi}\widehat{\mu}_i\right\|_{apx}^2}\right)\right| + \left|\frac{10^{-6}\cdot n^{-1}}{0.5\cdot n^{-1}}\right| && \text{By (3.195) and } \frac{\epsilon}{\varphi^2} \text{ small}\\
&\leq \left|\left\langle f_x,\widehat{\Pi}\widehat{\mu}_i\right\rangle\right|\left|\frac{1}{\|\widehat{\Pi}\widehat{\mu}_i\|^2}-\frac{1}{\left\|\widehat{\Pi}\widehat{\mu}_i\right\|_{apx}^2}\right| + 10^{-5} && (3.196)
\end{aligned}
$$

Now we will separately bound $\left|\left\langle f_x,\widehat{\Pi}\widehat{\mu}_i\right\rangle\right|$ and $\left|\frac{1}{\|\widehat{\Pi}\widehat{\mu}_i\|^2}-\frac{1}{\left\|\widehat{\Pi}\widehat{\mu}_i\right\|_{apx}^2}\right|$ from (3.196). As $|\langle a,b\rangle|\leq ||a||\cdot||b||$ we get:

$$
\left|\left\langle f_x,\widehat{\Pi}\widehat{\mu}_i\right\rangle\right| \leq ||\Pi_i f_x||\cdot||\widehat{\Pi}\widehat{\mu}_i|| \qquad (3.197)
$$

Now we bound the second term from (3.196):

$$\begin{aligned}
\left|\frac{1}{\|\widehat{\Pi}\widehat{\mu}_i\|^2} - \frac{1}{\left\|\widehat{\Pi}\widehat{\mu}_i\right\|^2_{apx}}\right| &= \left|\frac{\left\|\widehat{\Pi}\widehat{\mu}_i\right\|^2_{apx} - ||\widehat{\Pi}\widehat{\mu}_i||^2}{||\widehat{\Pi}\widehat{\mu}_i||^2 \left\|\widehat{\Pi}\widehat{\mu}_i\right\|^2_{apx}}\right| \\
&\leq \left|\frac{10^{-6}\cdot\frac{\sqrt{\epsilon}}{\varphi}\cdot n^{-1}}{||\widehat{\Pi}\widehat{\mu}_i||^2 \left\|\widehat{\Pi}\widehat{\mu}_i\right\|^2_{apx}}\right| && \text{Corollary 3.1, setting of } \xi \text{ and assumptions} \\
&\leq 10^{-5}\cdot\frac{\sqrt{\epsilon}}{\varphi}\cdot\left|\frac{0.5\cdot n^{-1}}{||\widehat{\Pi}\widehat{\mu}_i||^2\cdot 0.5\cdot n^{-1}}\right| && \text{By (3.195)} \\
&\leq 10^{-5}\cdot\frac{\sqrt{\epsilon}}{\varphi}\cdot\left|\frac{1}{||\widehat{\Pi}\widehat{\mu}_i||\cdot(||\Pi\mu_i|| - \frac{40\epsilon^{1/4}}{\sqrt{\varphi}}||\mu_i||)}\right| && \text{By (3.193)} \\
&\leq 10^{-4}\cdot\frac{\sqrt{\epsilon}}{\varphi}\cdot\frac{1}{||\widehat{\Pi}\widehat{\mu}_i||\cdot||\mu_i||} && \text{By Lemma 3.12 and } \frac{\epsilon}{\varphi^2} \text{ small}
\end{aligned} \tag{3.198}$$

Substituting (3.197) and (3.198) in (3.196) we get:

$$\begin{aligned}
\left|\frac{\langle f_x,\widehat{\Pi}\widehat{\mu}_i\rangle}{\|\widehat{\Pi}\widehat{\mu}_i\|^2} - \frac{\langle f_x,\widehat{\Pi}\widehat{\mu}_i\rangle_{apx}}{\left\|\widehat{\Pi}\widehat{\mu}_i\right\|^2_{apx}}\right| &\leq 10^{-5} + 10^{-4}\cdot\frac{\sqrt{\epsilon}}{\varphi}\cdot\frac{||\Pi_i f_x||}{||\mu_i||} \\
&\leq 10^{-5} + 10^{-4}\cdot\frac{\sqrt{\epsilon}}{\varphi}\cdot\frac{100}{\sqrt{\min_{p\in[k]}|C_p|}}\cdot\frac{1}{||\mu_i||} && \text{By assumption} \\
&\leq 10^{-5} + 10^{-3}\frac{1}{\sqrt{\max_{p\in[k]}|C_p|}\cdot||\mu_i||} && \text{As } \frac{\epsilon}{\varphi^2} \text{ small, } \frac{\max_{p\in[k]}|C_p|}{\min_{p\in[k]}|C_p|} = O(1) \\
&\leq 0.01 && \text{By Lemma 3.7}
\end{aligned}$$

□

Now we are ready to show that there exist an algorithm (Algorithm 11) that can estimate accurately the size of candidate clusters of the form $\widehat{C}_{\widehat{\mu}}^{(T_1,\ldots,T_b)}$ and then, if the size is not too small, estimate outer-conductance of all candidate clusters. The proof of correctness of the algorithm is based on applications of standard concentration bounds.

Lemma 3.44. *Let $k \geq 2$, $\varphi,\epsilon,\gamma \in (0,1)$. Let $G = (V,E)$ be a d-regular graph that admits a (k,φ,ϵ)-clustering $C_1,\ldots,C_k$.*

For a set of approximate centers $\{\widehat{\mu}_1,\ldots,\widehat{\mu}_k\}$, where each $\widehat{\mu}_i$ is represented as an average of at most s embedded vertices (i.e f_x's), an ordered partial partition $(T_1,\ldots,T_b)$ of $\{\widehat{\mu}_1,\ldots,\widehat{\mu}_k\}$ and $\widehat{\mu} \in \{\widehat{\mu}_1,\ldots,\widehat{\mu}_k\} \setminus \bigcup_{j\in[b]} T_i$ the following conditions hold.

Algorithm 11 OuterConductance$(G, \widehat{\mu}, (T_1, T_2, \dots, T_b), S, s_1, s_2)$

```
                                      ▷ $T_i$'s are sets of $\widehat{\mu}_j$ where $\widehat{\mu}_j$'s are given as sets of points
                                      ▷ see Section 3.5.6 for the reason of such representation
                                      ▷ $s_1$ is # sampled points for size estimation
                                      ▷ $s_2$ is # sampled points for outer-conductance estimation
 1: cnt := 0
 2: for $t = 1$ to $s_1$ do
 3:     $x \sim$ Uniform$\{1..n\}$      ▷ Sample a random vertex and test if it belongs to the cluster
 4:     if IsInside$(x, \widehat{\mu}, (T_1, T_2, \dots, T_b), S)$ then
 5:         cnt := cnt + 1
 6: if $\frac{n}{s_1} \cdot$ cnt $< \min_{p \in [k]} |C_p|/2$ then
 7:     return $\infty$                    ▷ If the estimated size is too small return $\infty$
 8: $e := 0, a := 0$
 9: for $t = 1$ to $s_2$ do
10:     $x \sim$ Uniform$\{1..n\}$
11:     $y \sim$ Uniform$\{w \in \mathcal{N}(u)\}$          ▷ $\mathcal{N}(u)$ = neighbors of $u$ in $G$
12:     if IsInside$(x, \widehat{\mu}, (T_1, T_2, \dots, T_b), S)$ then
13:         $a := a + 1$
14:         if $\neg$IsInside$(y, \widehat{\mu}, (T_1, T_2, \dots, T_b), S)$ then
15:             $e = e + 1$
16: return $\frac{e}{a}$
```

If Algorithm 11 is invoked with $(G, \widehat{\mu}, (T_1, \dots, T_b), \{\widehat{\mu}_1, \dots, \widehat{\mu}_k\} \setminus \bigcup_{j \in [b]} T_i, s_1, s_2)$ *then it runs in* $\widetilde{O}_\varphi\left((s_1 + s_2) \cdot s^4 \cdot \left(\frac{k}{\epsilon}\right)^{O(1)} \cdot n^{1/2 + O(\epsilon/\varphi^2)}\right)$ *time and if* $s_1 = \Theta(k \log(\frac{1}{\eta}))$ *and* $s_2 = \Theta(\frac{\varphi^2 \cdot k}{\epsilon} \log(\frac{1}{\eta}))$ *then with probability* $1 - \eta$ *it returns a value* q *with the following properties.*

- *If* $|\widehat{C}_{\widehat{\mu}}^{(T_1, \dots, T_b)}| \geq \frac{3}{4} \min_{p \in [k]} |C_p|$ *then* $q \in \left[\frac{1}{2}\phi\left(\widehat{C}_{\widehat{\mu}}^{(T_1, \dots, T_b)}\right) - \epsilon/\varphi^2, \frac{3}{2}\phi\left(\widehat{C}_{\widehat{\mu}}^{(T_1, \dots, T_b)}\right) + \epsilon/\varphi^2\right]$,
- *If* $|\widehat{C}_{\widehat{\mu}}^{(T_1, \dots, T_b)}| < \frac{3}{4} \min_{p \in [k]} |C_p|$ *then* $q \geq \frac{1}{2}\phi\left(\widehat{C}_{\widehat{\mu}}^{(T_1, \dots, T_b)}\right) - \epsilon/\varphi^2$.

Proof. We start with the runtime analysis then follows the correctness analysis.

Runtime. Algorithm 11 has two phases: one from line 1 to line 7 and second from line 8 to line 16.

During the first phase Algorithm 11 calls Algorithm 9 s_1 times and Algorithm 9 runs in $\widetilde{O}_\varphi(s^4 \cdot \left(\frac{k}{\epsilon}\right)^{O(1)} \cdot n^{1/2 + O(\epsilon/\varphi^2)})$ time as it computes $k^{O(1)}$ values of the form $\frac{\langle f_x, \hat{\mu}_i \rangle_{apx}}{||\hat{\mu}_i||^2_{apx}}$ which are computed in time $\widetilde{O}_\varphi(s^4 \cdot \left(\frac{k}{\epsilon}\right)^{O(1)} \cdot n^{1/2 + O(\epsilon/\varphi^2)})$ by Lemma 3.43, so in total the runtime of this phase is $\widetilde{O}_\varphi(s_1 \cdot s^4 \cdot \left(\frac{k}{\epsilon}\right)^{O(1)} \cdot n^{1/2 + O(\epsilon/\varphi^2)})$.

During the second phase Algorithm 11 calls Algorithm 9 $2s_2$ times so the runtime of this phase is $\widetilde{O}_\varphi(s_2 \cdot s^4 \cdot \left(\frac{k}{\epsilon}\right)^{O(1)} \cdot n^{1/2+O(\epsilon/\varphi^2)})$ in total.

So in total the runtime is $\widetilde{O}_\varphi((s_1+s_2) \cdot s^4 \cdot \left(\frac{k}{\epsilon}\right)^{O(1)} \cdot n^{1/2+O(\epsilon/\varphi^2)})$.

Correctness. For simplicity we denote $\widehat{C}_{\widehat{\mu}}^{(T_1,\dots,T_b)}$ by $\widehat{C}$ and $\min_{p\in[k]}|C_p|$ by $r_{\min}$ in this proof. Notice that the Algorithm 11 in the first phase computes $\text{cnt} = \sum_{i=1}^{s} X_i$, where X_i's are independent Bernoulli trials with success probability $p = \frac{|\widehat{C}|}{n}$. Let $z := \frac{n}{s_1}\sum_{i=1}^{s_1} X_i$. We introduce notation $x \approx_{\delta,\alpha} y$ to denote $x \in [(1-\delta)y-\alpha,(1+\delta)y+\alpha]$. By Chernoff-Hoeffding bounds we get that there exists a universal constant Γ such that for all $0<\delta\le 1/2, \alpha>0$

$$z \approx_{\delta,\alpha\cdot n} |\widehat{C}| \text{ with probability } 1-2^{-\Gamma s_1 \alpha\delta}.$$

Setting $\delta = 1/2, \alpha = \frac{r_{\min}}{8n}$ we get that $z \approx_{1/2, r_{\min}/8} |\widehat{C}|$ with probability

$$1-2^{-\Gamma s_1 \frac{r_{\min}}{32n}} \ge 1-2^{-\Omega(s_1/k)},$$

as $\frac{\max_{p\in[k]}|C_p|}{\min_{p\in[k]}|C_p|} = O(1)$. So if $s_1 = \Theta(k\log(1/\eta))$ then with probability $1-\eta/2$ we have

$$z \approx_{1/2, r_{\min}/8} |\widehat{C}|. \tag{3.199}$$

Observe that if $\widehat{C} < r_{\min}/4$ then by (3.199) we have that $z \le (1+1/2)|\widehat{C}| + r_{\min}/8 < r_{\min}/2$, which means that Algorithm 11 returns ∞. Note that it is consistent with the conclusion of the Lemma.

For the analysis of the second stage we assume that $|\widehat{C}| \ge r_{\min}/4$. We will analyze what value is returned in the second stage. First we will bound the probability that $a \le \frac{s_2\cdot r_{\min}}{8\cdot n}$. For $i \in [1\dots s_2]$ let X_i be a binary random variable which is equal 1 iff in $i-th$ iteration of the for loop we increase the a counter. We have that, for every i, $P[X_i = 1] = |\widehat{C}|/n$ and the X_i's are independent. Notice that $a = \sum_{i=1}^{s_2} X_i$. From Chernoff bound we have that for $\delta < 1$:

$$P\left[\left|\sum_{i=1}^{s_2} X_i - \mathbb{E}\left[\sum_{i=1}^{s_2} X_i\right]\right| > \delta\cdot\mathbb{E}\left[\sum_{i=1}^{s_2} X_i\right]\right] \le 2e^{-\frac{\delta^2}{3}\mathbb{E}[\sum_{i=1}^{s_2} X_i]}, \tag{3.200}$$

Noticing that $\mathbb{E}\left[\sum_{i=1}^{s_2} X_i\right] = s_2\frac{|\widehat{C}|}{n}$ if we set $\delta = 1/2$ we get that

$$P\left[\left|\sum_{i=1}^{s_2} X_i - s_2\frac{|\widehat{C}|}{n}\right| > s_2\frac{|\widehat{C}|}{2n}\right] \le 2e^{-s_2\frac{|\widehat{C}|}{12n}} \le 2e^{-\frac{s_2\cdot r_{\min}}{48\cdot n}}, \tag{3.201}$$

So with probability at least $1-2e^{-\frac{s_2\cdot r_{\min}}{48\cdot n}} \ge 1-2e^{-\Omega(s_2/k)}$ (as $\frac{\max_{p\in[k]}|C_p|}{\min_{p\in[k]}|C_p|} = O(1)$) we have that

$$a = \sum_{i=1}^{s} X_i \ge \frac{1}{2}\cdot s_2 \cdot \frac{|\widehat{C}|}{n} \ge \frac{s_2\cdot r_{\min}}{8\cdot n} \ge \Omega(s_2/k). \tag{3.202}$$

Now observe that line 14 of OUTERCONDUCTANCE is invoked exactly a times. Let Y_j be the indicator random variable that is 1 iff e is increased in the j-th call of line 14. Notice that

$$P[Y_i = 1] = \phi(\widehat{C}) \tag{3.203}$$

That is because if U_i is a random variable denoting a vertex u sampled in i-th step then U_i is uniform on set $\widehat{C}$ conditioned on $X_i = 1$ and the graph is regular. Now by the Chernoff-Hoeffging bounds we get that for all $0 < \delta \leq 1/2, \alpha > 0$ we have:

$$\frac{1}{a}\sum_{i=1}^{a} Y_i \approx_{\delta,\alpha} \phi(\widehat{C}) \text{ with probability } 1 - 2e^{-\Gamma a\alpha\delta}.$$

Setting $\delta = 1/2, \alpha = \frac{\epsilon}{\varphi^2}$ we get that $\frac{1}{a}\sum_{i=1}^{a} Y_i \approx_{1/2,\epsilon/\varphi^2} \phi(\widehat{C})$ with probability:

$$1 - 2e^{-\Gamma a\epsilon/(4\varphi^2)} \geq 1 - 2e^{-\Omega(a\epsilon/\varphi^2)} \tag{3.204}$$

Now taking the union bound over (3.202) and (3.204) we get that if we set $s_2 = \Theta(\frac{\varphi^2 \cdot k}{\epsilon} \log(1/\eta))$ then $\frac{1}{a}\sum_{i=1}^{a} Y_i \approx_{1/2,\epsilon/\varphi^2} \phi(\widehat{C})$ with probability:

$$\begin{aligned} 1 - 2e^{-\Omega(s_2/k)} - 2e^{-\Omega(a\epsilon/\varphi^2)} &\geq 1 - 2e^{-\Omega(s_2/k)} - 2e^{-\Omega(\frac{\epsilon \cdot s_2}{\varphi^2 \cdot k})} && \text{By (3.202)} \\ &\geq 1 - \eta/2 \end{aligned}$$

To conclude the proof we observe the following.

- If $|\widehat{C}| < \frac{r_{\min}}{4}$ then with probability $1 - \eta/2$ the Algorithm returns ∞,
- If $|\widehat{C}| \in [\frac{r_{\min}}{4}, \frac{3 \cdot r_{\min}}{4})$ then either the Algorithm returns ∞ in the first stage or it reaches the second stage and with probability $1 - \eta$ it returns a value ψ such that $\psi \approx_{1/2,\epsilon/\phi^2} \varphi(\widehat{C})$,
- If $|\widehat{C}| \geq \frac{r_{\min}}{4}$ then by the union bound over the two stages with probability $1 - \eta$ it reaches the second stage and returns a value ψ such that $\psi \approx_{1/2,\epsilon/\varphi^2} \phi(\widehat{C})$.

The above covers all the cases and is consistent with the conclusions of the Lemma.

□

Before we give the statement of the next Lemma we introduce some definitions. In Lemma 3.44 we proved that for every call to OUTERCONDUCTANCE the value returned by the Algorithm 11 is, in a sense given by the conclusions of Lemma 3.44, a good approximation to outer-conductance of $\widehat{C}_{\widehat{\mu}}^{(T_1,\ldots,T_b)}$ (where $\widehat{\mu}, (T_1, \ldots, T_b)$ are the parameters of the call) with high probability. What follows is a definition of an event that the values returned by OUTERCONDUCTANCE throughout the run of the final algorithm always satisfy one conclusion of Lemma 3.44. Later we use Definition 3.13 in Lemma 3.45 and then in the proof of Theorem 3.8 we will lower bound the probability of $\mathscr{E}_{\text{conductance}}$.

Definition 3.13 (**Event $\mathscr{E}_{\textbf{conductance}}$**)**.** Let $k \geq 2$, $\varphi, \epsilon, \gamma \in (0,1)$. Let $G = (V, E)$ be a d-regular graph that admits a (k, φ, ϵ)-clustering $C_1, \ldots, C_k$.

We define $\mathscr{E}_{\text{conductance}}$ as an event:

For every call to Algorithm 11 (i.e. OUTERCONDUCTANCE) that is made throughout the run of FINDCENTERS the following is true. If Algorithm 11 is invoked with $(G, \widehat{\mu}, (T_1, \ldots, T_b), \{\widehat{\mu}_1, \ldots, \widehat{\mu}_k\} \setminus \bigcup_{j \in [b]} T_i, s_1, s_2)$ then it returns a value q with the following property.

- If $|\widehat{C}_{\widehat{\mu}}^{(T_1,\ldots,T_b)}| \geq \frac{3}{4} \min_{p \in [k]} |C_p|$ then $q \in \left[\frac{1}{2}\phi\left(\widehat{C}_{\widehat{\mu}}^{(T_1,\ldots,T_b)}\right) - \epsilon/\varphi^2, \frac{3}{2}\phi\left(\widehat{C}_{\widehat{\mu}}^{(T_1,\ldots,T_b)}\right) + \epsilon/\varphi^2\right]$.

The following Lemma is the key part of the corresponding proof of correctness of Algorithm 8 (see Theorem 3.8 below). It is a generalization of Lemma 3.37. We show that if $\widehat{\mu}$'s are close to real centers and $\mathscr{E}$ and $\mathscr{E}_{\text{conductance}}$ hold then at every stage of the for loop from line 4 of Algorithm 8 at least half of the candidate clusters:

$$\mathscr{C}_i := \bigcup_{\widehat{\mu} \in S} \{\widehat{C}_{\widehat{\mu}}^{(T_1,\ldots,T_{i-1})}\},$$

pass the test from line 6 of Algorithm 8, which means that they have small outer-conductance and satisfy condition (3.143).

Lemma 3.45. *Let $k \geq 2$, $\varphi \in (0,1)$, $\frac{\epsilon}{\varphi^2}$ be smaller than a sufficiently small constant. Let $G = (V, E)$ be a d-regular graph that admits a (k, φ, ϵ)-clustering $C_1, \ldots, C_k$. Then conditioned on the success of the spectral dot product oracle there exists an absolute constant Υ such that the following conditions hold.*

If COMPUTEORDEREDPARTITION*$(G, \widehat{\mu}_1, \widehat{\mu}_2, \ldots, \widehat{\mu}_k, s_1, s_2)$ is called with $(\widehat{\mu}_1, \ldots, \widehat{\mu}_k)$ such that for every $i \in [k]$ we have $\|\widehat{\mu}_i - \mu_i\|^2 \leq 10^{-12} \cdot \frac{\epsilon}{\varphi^2 \cdot k^2} \|\mu_i\|^2$ then the following holds. Assume that at the beginning of the i-th iteration of the for loop from line 4 of Algorithm 8 $|S| = b$ and, up to renaming of $\widehat{\mu}$'s, $S = \{\widehat{\mu}_1, \ldots, \widehat{\mu}_b\}$, the corresponding clusters are $\mathscr{C} = \{C_1, \ldots, C_b\}$ respectively and the ordered partial partition of μ's is equal to $(T_1, \ldots, T_{i-1})$. Then if for every $C \in \mathscr{C}$ we have that $|V^{(T_1,\ldots,T_{i-1})} \cap C| \geq \left(1 - \Upsilon \cdot i \cdot \frac{\epsilon}{\varphi^2}\right)|C|$ then at the beginning of $(i+1)$-th iteration:*

1. *$|S| \leq b/2$ (that is at least half of the remaining cluster means were removed in i-th iteration),*
2. *for every $\mu \in S$ the corresponding cluster C satisfies $|V^{(T_1,\ldots,T_i)} \cap C| \geq \left(1 - \Upsilon \cdot (i+1) \cdot \frac{\epsilon}{\varphi^2}\right)|C|$, where $(T_1, \ldots, T_i)$ is the ordered partial partition of μ's created in the first i iterations.*

Proof. **Outline of the proof.** We start but defining a subset of vertices called outliers and then we show that the number of them is small. Next we prove that for vertices that are not outliers the evaluations of $\frac{\langle f_x, \widehat{\Pi}\widehat{\mu}_i \rangle_{apx}}{\|\widehat{\Pi}\widehat{\mu}_i\|_{apx}^2}$ are approximately correct (as in Lemma 3.43). Next we mimic the structure, and on the high level the logic, of the proof of Lemma 3.37: we first show the first conclusion of the Lemma and then the second one.

For simplicity we will denote $\min_{p\in[k]}|C_p|$ by $r_{\min}$ in this proof. Without loss of generality we can assume $S = \{\hat{\mu}_1,\ldots,\hat{\mu}_b\}$ at the beginning of the i-th iteration of the for loop from line 4 of Algorithm 8 and the corresponding clusters be $C_1,\ldots,C_b$ respectively. Assume that for every $C \in \mathscr{C}$ we have that $|V^{(T_1,\ldots,T_{i-1})} \cap C| \geq \left(1 - \Upsilon \cdot i \cdot \frac{\epsilon}{\varphi^2}\right)|C|$.

Let $\widehat{\Pi}$ be the projection onto the $\text{span}(\bigcup_{j<i} T_j)^{\perp}$. Recall that each T_j is a subset of $\{\widehat{\mu}_1,\ldots,\widehat{\mu}_k\}$. For every $j < i$ let

$$T'_j := \bigcup_{\widehat{\mu}\in T_j} \{\mu\}.$$

That is T'_j's are T_j's with $\widehat{\mu}$'s replaced by the corresponding μ's. Now let Π be the projection onto the $\text{span}(\bigcup_{j<i} T'_j)^{\perp}$.

Outliers. First we define a set of outliers, i.e. X, as the set of points with abnormally long projection onto the subspace spanned by $\{\Pi\mu_1,\ldots,\Pi\mu_b,\widehat{\Pi}\widehat{\mu}_1,\ldots,\widehat{\Pi}\widehat{\mu}_b\}$. Then we show that the number of outliers is small.

Let Q be the orthogonal projection onto the $\text{span}(\{\Pi\mu_1,\ldots,\Pi\mu_b,\widehat{\Pi}\widehat{\mu}_1,\ldots,\widehat{\Pi}\widehat{\mu}_b\})$ and let:

$$X := \left\{x \in V : ||Qf_x||^2 > \frac{10^4}{r_{\min}}\right\}$$

By Lemma 3.33 we get that

$$\sum_{x\in V} ||Qf_x - Q\mu_x||^2 \leq O\left(b\cdot\frac{\epsilon}{\varphi^2}\right). \tag{3.205}$$

Moreover for every $x \in X$:

$$\begin{aligned} ||Qf_x - Q\mu_x|| &\geq ||Qf_x|| - ||Q\mu_x|| && \text{By triangle inequality} \\ &\geq ||Qf_x|| - ||\mu_x|| && \text{As projection can only decrease the norm} \\ &> \frac{10^2}{\sqrt{r_{\min}}} - \left(1 + O\left(\frac{\sqrt{\epsilon}}{\varphi}\right)\right)\frac{1}{\sqrt{r_{\min}}} && \text{By Lemma 3.7 and Definition of } X \\ &\geq \frac{90}{\sqrt{r_{\min}}} && \text{For } \frac{\epsilon}{\varphi^2} \text{ small enough} \end{aligned} \tag{3.206}$$

Combining (3.205) and (3.206) we get:

$$|X| \leq O\left(b\cdot\frac{\epsilon}{\varphi^2}\right)\cdot r_{\min} \leq O\left(b\cdot\frac{\epsilon}{\varphi^2}\right)\cdot\frac{n}{k} \tag{3.207}$$

Tests performed for non-outliers are approximately correct. Observe that by the fact that spectral dot product succeeds we have by Lemma 3.43 that for all $x \in V \setminus X$ and for all $i \in [k]$:

$$\left| \frac{\langle f_x, \Pi\mu_i \rangle}{\|\Pi\mu_i\|^2} - \frac{\langle f_x, \widehat{\Pi}\widehat{\mu}_i \rangle_{apx}}{\|\widehat{\Pi}\widehat{\mu}_i\|^2_{apx}} \right| \leq 0.02, \tag{3.208}$$

as $||Qf_x||^2 \leq \frac{10^4}{r_{\min}}$ and the norm in any subspace can only be smaller and thus the assumption of Lemma 3.43 is satisfied.

1. At least half of the cluster means is removed from S. Now we proceed with proving that most of the candidate clusters $\widehat{C}^{(T_1,\ldots,T_{i-1})}_{\widehat{\mu}}$ have small outer-conductance and thus the corresponding $\widehat{\mu}$'s are removed from set S (see line 6 of ComputeOrderedPartition). For brevity we will refer to $(T_1, \ldots, T_{i-1})$ as P in this proof.

Let $\mu \in S$. Let

$$I := \bigcup_{\mu', \mu'' \in \{\mu_1, \ldots, \mu_d\}} C_{\Pi\mu', 0.9} \cap C_{\Pi\mu'', 0.9}.$$

By Lemma 3.36 we have that

$$|I| \leq O\left(b \cdot \frac{\epsilon}{\varphi^2}\right) \cdot \frac{n}{k}. \tag{3.209}$$

So by (3.207) and (3.209) and Markov inequality we get that there exists a subset of clusters $\mathscr{R} \subseteq \mathscr{C}$ such that $|\mathscr{R}| \geq b/2$ and for every $C \in \mathscr{R}$ we have that:

$$|C \cap (I \cup X)| \leq O\left(\frac{\epsilon}{\varphi^2}\right) \cdot \frac{n}{k} \tag{3.210}$$

We will argue that for any order of the for loop from line 4 of Algorithm 8 it is true that for every $C \in \mathscr{R}$ with corresponding means $\mu, \widehat{\mu}$ the candidate cluster $\widehat{C}^P_{\widehat{\mu}}$ satisfies the if statement from line 6 of Algorithm 8. Recall that as per Definition 3.12:

$$\widehat{C}^P_{\widehat{\mu}} = \left\{ x \in V : \textsc{IsInside}\left(x, \widehat{\mu}, P, \{\widehat{\mu}_1, \ldots, \widehat{\mu}_k\} \setminus \bigcup_{j \in [i-1]} T_j \right) = \textsc{True} \right\}.$$

First note that behavior of the algorithm is independent of the order of the for loop from line 4 of Algorithm 8 as by definition $\widehat{C}^P_{\widehat{\mu}}$'s for $\widehat{\mu} \in S$ are pairwise disjoint. Now let $C \in \mathscr{R}$, $\mu, \widehat{\mu}$ be the means corresponding to C and $\widehat{C}^P_{\widehat{\mu}}$ be the candidate cluster corresponding to $\widehat{\mu}$ with respect to $P = (T_1, \ldots, T_{i-1})$.

Now the goal is to show:

$$|\widehat{C}^P_{\widehat{\mu}} \triangle C| \leq O\left(\frac{\epsilon}{\varphi^2} \cdot \log(k)\right) \cdot |C|,$$

from which we will later conclude that the outer-conductance of the candidate set $\widehat{C}^P_{\widehat{\mu}}$ is small.

Intuitively we would like to argue that

$$C_{\Pi\mu,0.96} \,\widetilde{\subseteq}\, C^{apx}_{\widehat{\Pi}\widehat{\mu},0.93} \,\widetilde{\subseteq}\, C_{\Pi\mu,0.9}, \tag{3.211}$$

and then use Lemmas from Section 3.6.2. The equation (3.211) *is true up to the outliers as Lemma 3.43 guarantees a bound of* 0.02 *for the test computations for vertices of small norm.*

Now we give a formal proof, which is split into 2 parts:

Showing $|\widehat{C}^P_{\widehat{\mu}} \cap C| \geq \left(1 - O\left(\frac{\epsilon}{\varphi^2} \cdot \log(k)\right)\right)|C|$**.**First we note that by (3.208) $C_{\Pi\mu,0.96}$ is mostly contained in $C^{apx}_{\widehat{\Pi}\widehat{\mu},0.93}$. Recall that (see Definition 3.9 and Definition 3.8) we have:

$$C^{apx}_{\widehat{\Pi}\widehat{\mu},0.93} = \left\{x \in V : \left\langle f_x, \widehat{\Pi}\widehat{\mu}\right\rangle_{apx} \geq 0.93 \left\|\widehat{\Pi}\widehat{\mu}\right\|^2_{apx}\right\},$$

$$C_{\Pi\mu,0.96} = \left\{x \in V : \left\langle f_x, \Pi\mu\right\rangle \geq 0.96||\Pi\mu||^2\right\}.$$

And (3.208) gives us that the errors for non-outliers are bounded by 0.02, so formally we get:

$$C_{\Pi\mu,0.96} \setminus C^{apx}_{\widehat{\Pi}\widehat{\mu},0.93} \subseteq X \tag{3.212}$$

Similarly, also by (3.208) we get that the intersections of candidate clusters $C^{apx}_{\widehat{\Pi}\widehat{\mu},0.93}$ lie mostly in I. Formally:

$$C^{apx}_{\widehat{\Pi}\widehat{\mu},0.93} \cap \bigcup_{\widehat{\mu}' \neq \widehat{\mu}} C^{apx}_{\widehat{\Pi}\widehat{\mu}',0.93} \subseteq I \cup X \tag{3.213}$$

By Lemma 3.31 we get that

$$|C \cap C_{\Pi\mu,0.96}| \geq \left(1 - O\left(\frac{\epsilon}{\varphi^2}\right)\right)|C| \tag{3.214}$$

*Note that having two thresholds (*0.9 *and* 0.96*) is very important here (see Remark 3.7). Intuitively we need some slack to show* $C_{\Pi\mu,0.96} \,\widetilde{\subseteq}\, C^{apx}_{\widehat{\Pi}\widehat{\mu},0.93} \,\widetilde{\subseteq}\, C_{\Pi\mu,0.9}$ *as there is always some error in computation of* $\frac{\langle f_x, \widehat{\Pi}\widehat{\mu}_i\rangle_{apx}}{\|\widehat{\Pi}\widehat{\mu}_i\|^2_{apx}}$.

Now combining inductive assumption $|V^P \cap C| \geq \left(1 - \Upsilon \cdot i \cdot \frac{\epsilon}{\varphi^2}\right)|C|$, (3.210), (3.212), (3.213) and (3.214) we get that:

$$\begin{aligned}|\widehat{C}^P_{\widehat{\mu}} \cap C| &\geq \left(1 - \Upsilon \cdot i \cdot \frac{\epsilon}{\varphi^2}\right)|C| - O\left(\frac{\epsilon}{\varphi^2}\right) \cdot \frac{n}{k} - O\left(\frac{\epsilon}{\varphi^2}\right) \cdot |C| \\ &\geq \left(1 - O\left(\frac{\epsilon}{\varphi^2} \cdot \log(k)\right)\right)|C| \end{aligned} \tag{3.215}$$

Showing $|\widehat{C}^P_{\widehat{\mu}} \cap (V^P \setminus C)| \leq O\left(\frac{\epsilon}{\varphi^2}\right)|C|$**.**Recall that as per Definition 3.12 we have:

$$V^P = V \setminus \bigcup_{j<i} \bigcup_{\widehat{\mu} \in T_j} \widehat{C}^{(T_1,\ldots,T_{j-1})}_{\widehat{\mu}}$$

By Lemma 3.32 we get that:

$$|C_{\Pi\mu,0.9} \cap (V^P \setminus C)| \leq |C_{\Pi\mu,0.9} \cap (V \setminus C)| \leq O\left(\frac{\epsilon}{\varphi^2}\right)|C| \tag{3.216}$$

By (3.208) we get:

$$C^{apx}_{\widehat{\Pi}\widehat{\mu},0.93} \setminus C_{\Pi\mu,0.9} \subseteq X \tag{3.217}$$

Let π' be the projection onto the span of $\{\Pi\mu, \widehat{\Pi}\widehat{\mu}\}$. Moreover let:

$$X' := \left\{x \in V : \|\pi' f_x\|^2 > \frac{10^4}{r_{\min}}\right\}.$$

Note that by Lemma 3.33 we have:

$$\sum_{x\in V} ||\pi' f_x - \pi' \mu_x||^2 \leq O\left(\frac{\epsilon}{\varphi^2}\right) \tag{3.218}$$

Moreover for every $x \in X'$ we have:

$$\begin{aligned} ||\pi' f_x - \pi' \mu_x|| &\geq ||\pi' f_x|| - ||\pi' \mu_x|| && \text{By } \triangle \text{ inequality} \\ &\geq \frac{10^2}{\sqrt{r_{\min}}} - \frac{2}{\sqrt{r_{\min}}} && \text{By Lemma 3.7} \\ &\geq \frac{90}{\sqrt{r_{\min}}} \end{aligned} \tag{3.219}$$

Combining (3.218) and (3.219) we get that:

$$|X'| \leq O\left(\frac{\epsilon}{\varphi^2}\right) \cdot r_{\min} \leq O\left(\frac{\epsilon}{\varphi^2}\right) \cdot \frac{n}{k} \tag{3.220}$$

Then similarly to the analysis of (3.208) by Lemma 3.43 and the fact that spectral dot product succeeds we have that for every $x \in V \setminus X'$:

$$\left| \frac{\langle f_x, \Pi\mu\rangle}{\|\Pi\mu\|^2} - \frac{\left\langle f_x, \widehat{\Pi}\widehat{\mu}\right\rangle_{apx}}{\left\|\widehat{\Pi}\widehat{\mu}\right\|^2_{apx}} \right| \leq 0.02$$

Thus we get:

$$C^{apx}_{\widehat{\Pi}\widehat{\mu},0.93} \setminus C_{\Pi\mu,0.9} \subseteq X', \tag{3.221}$$

as for points not belonging to X' the error in the tests performed by the Algorithm is upper bounded by 0.02. Combining (3.216) and (3.221) we have:

$$|\widehat{C}^P_{\widehat{\mu}} \cap (V^P \setminus C)| \le O\left(\frac{\epsilon}{\varphi^2}\right)|C| \tag{3.222}$$

And finally putting (3.215) and (3.222) together we have:

$$|\widehat{C}^P_{\widehat{\mu}} \triangle C| \le O\left(\frac{\epsilon}{\varphi^2} \cdot \log(k)\right) \cdot |C| \tag{3.223}$$

Outer-conductance of $\widehat{C}^P_{\widehat{\mu}}$ is small. Now we want to argue that $\widehat{C}^P_{\widehat{\mu}}$ passes the outer-conductance test from line 6 in Algorithm 8. From the definition of outer-conductance:

$$\begin{aligned}
\phi(\widehat{C}^P_{\widehat{\mu}}) &\le \frac{E(C, V \setminus C) + d|\widehat{C}^P_{\widehat{\mu}} \triangle C|}{d(|C| - |\widehat{C}^P_{\widehat{\mu}} \triangle C|)} \\
&\le \frac{E(C, V \setminus C) + d \cdot O\left(\frac{\epsilon}{\varphi^2} \cdot \log(k)\right)|C|}{d(|C| - O\left(\frac{\epsilon}{\varphi^2} \cdot \log(k)\right)|C|)} && \text{from (3.223)} \\
&\le \frac{O\left(\frac{\epsilon}{\varphi^2}\right) + O\left(\frac{\epsilon}{\varphi^2} \cdot \log(k)\right)}{1 - O\left(\frac{\epsilon}{\varphi^2} \cdot \log(k)\right)} && \text{because } \frac{E(C, V \setminus C)}{d|C|} \le O\left(\frac{\epsilon}{\varphi^2}\right) \\
&\le O\left(\frac{\epsilon}{\varphi^2} \cdot \log(k)\right) && \text{for sufficiently small } \frac{\epsilon}{\varphi^2} \cdot \log(k)
\end{aligned}$$

and it follows that

$$\phi(\widehat{C}^P_{\widehat{\mu}}) \le O\left(\frac{\epsilon}{\varphi^2} \cdot \log(k)\right),$$

To conclude we notice that by (3.223) we have $|\widehat{C}^P_{\widehat{\mu}}| > \frac{3 \cdot r_{\min}}{4}$, so as $\mathcal{E}_{\text{conductance}}$ is true we get that the candidate cluster $\widehat{C}^P_{\widehat{\mu}}$ passes the test.

2. Clusters corresponding to unremoved $\widehat{\mu}$'s satisfy condition 2. Now we prove that for every $\widehat{\mu}$ that was not removed from set S only small fraction of its corresponding cluster is removed.

Let $\widehat{\mu} \in S$ be such that it is not removed in the i-th step and let μ be the corresponding real center. Let $C \in \mathcal{C}$ be the cluster corresponding to μ. By assumption $|V^P \cap C| \ge \left(1 - \Upsilon \cdot i \cdot \frac{\epsilon}{\varphi^2}\right)|C|$, where recall that $P = (T_1, \ldots, T_{i-1})$.

Now the goal is to show:

$$|C \cap (V^{(T_1,\ldots,T_{i-1})} \setminus V^{(T_1,\ldots,T_i)})| \le O\left(\frac{\epsilon}{\varphi^2}\right)|C| + O\left(\frac{\epsilon}{\varphi^2}\right) \cdot \frac{n}{k} \le O\left(\frac{\epsilon}{\varphi^2}\right)|C|,$$

that is, that there is only a small number of vertices that were removed in the i-th stage and *belong to C at the same time. Intuitively we want to show that:*

$$(V^{(T_1,\ldots,T_{i-1})} \setminus V^{(T_1,\ldots,T_i)}) \cap C_{\Pi\mu,0.96} \approx \emptyset,$$

and then use Lemmas from Section 3.6.2. The equation above is true up to the outliers as Lemma 3.43 guarantees a bound of 0.02 *for the test computations for vertices of small norm.*

Now we give a formal proof. Let $x \in V^{(T_1,\ldots,T_{i-1})} \setminus V^{(T_1,\ldots,T_i)} = V^P \setminus V^{(T_1,\ldots,T_i)}$, where $(T_1,\ldots,T_i)$ is the partial partition of $\widehat{\mu}$'s created in the first i steps of the for loop of COMPUTEORDEREDPARTITION. Then there exists $\widehat{\mu}' \in \{\widehat{\mu}_1,\ldots,\widehat{\mu}_b\}$ such that $x \in \widehat{C}^P_{\widehat{\mu}'}$ (recall that $\widehat{C}^P_{\widehat{\mu}'}$ is the candidate cluster corresponding to $\widehat{\mu}'$ with respect to $P = (T_1,\ldots,T_{i-1})$). Recall (Definition 3.12) that $\widehat{C}^P_{\widehat{\mu}'}$ is defined as:

$$\widehat{C}^P_{\widehat{\mu}'} = \left\{ x \in V : \textsc{IsInside}\left(x, \widehat{\mu}', P, \{\widehat{\mu}_1,\ldots,\widehat{\mu}_k\} \setminus \bigcup_{j\in[i-1]} T_j \right) = \textsc{True} \right\}.$$

This in particular means (see line 8 of Algorithm ISINSIDE) that:

$$\widehat{C}^P_{\widehat{\mu}'} \subseteq C^{apx}_{\widehat{\Pi}\widehat{\mu}',0.93} \setminus \bigcup_{\widehat{\mu}''\in S\setminus\{\widehat{\mu}'\}} C^{apx}_{\widehat{\Pi}\widehat{\mu}'',0.93},$$

which, as $\widehat{\mu} \in S \setminus \{\widehat{\mu}'\}$, gives us that:

$$\widehat{C}^P_{\widehat{\mu}'} \cap C^{apx}_{\widehat{\Pi}\widehat{\mu},0.93} = \emptyset,$$

which using Definition 3.8 gives that:

$$\left\langle f_x, \widehat{\Pi}\widehat{\mu} \right\rangle_{apx} < 0.93 \left\| \widehat{\Pi}\widehat{\mu} \right\|^2_{apx}. \tag{3.224}$$

We define X' similarly as in point 1. Let π' be the projection onto the span of $\{\Pi\mu, \widehat{\Pi}\widehat{\mu}\}$. Moreover let:

$$X' := \left\{ x \in V : \|\pi' f_x\|^2 > \frac{10^4}{r_{\min}} \right\}.$$

Similarly to the proof of (3.220) we get

$$|X'| \le O\left(\frac{\epsilon}{\varphi^2}\right) \cdot r_{\min} \le O\left(\frac{\epsilon}{\varphi^2}\right) \cdot \frac{n}{k} \tag{3.225}$$

Again similarly to the analysis of (3.208) we note that by Lemma 3.43 and the fact that spectral dot product succeeds:

$$\text{for every } y \in V \setminus X' \text{ we have } \left| \frac{\langle f_y, \Pi\mu\rangle}{\|\Pi\mu\|^2} - \frac{\left\langle f_y, \widehat{\Pi}\widehat{\mu}\right\rangle_{apx}}{\left\|\widehat{\Pi}\widehat{\mu}\right\|^2_{apx}} \right| \le 0.02 \tag{3.226}$$

Combining (3.226) and (3.224) we get that if $x \in V \setminus X'$ then

$$\begin{aligned}\frac{\langle f_x, \Pi\mu\rangle}{\|\Pi\mu\|^2} &\leq \frac{\left\langle f_y, \widehat{\Pi}\widehat{\mu}\right\rangle_{apx}}{\left\|\widehat{\Pi}\widehat{\mu}\right\|^2_{apx}} + 0.02 \\ &< 0.93 + 0.02 \\ &< 0.96\end{aligned}$$

which also means that $x \notin C_{\Pi\mu,0.96}$. This means that:

$$(V^{(T_1,\ldots,T_{i-1})} \setminus V^{(T_1,\ldots,T_i)}) \cap C_{\Pi\mu,0.96} \subseteq X' \tag{3.227}$$

But by Lemma 3.31:

$$|\{x \in C : \langle \Pi f_x, \Pi\mu\rangle < 0.96\|\Pi\mu\|_2^2\}| \leq O\left(\frac{\epsilon}{\varphi^2}\right) \cdot |C| \tag{3.228}$$

Combining (3.227), (3.225) and (3.228) we get that:

$$|C \cap (V^{(T_1,\ldots,T_{i-1})} \setminus V^{(T_1,\ldots,T_i)})| \leq O\left(\frac{\epsilon}{\varphi^2}\right)|C| + O\left(\frac{\epsilon}{\varphi^2}\right) \cdot \frac{n}{k} \leq O\left(\frac{\epsilon}{\varphi^2}\right)|C|. \tag{3.229}$$

By assumption that $|V^{(T_1,\ldots,T_{i-1})} \cap C| \geq \left(1 - \Upsilon \cdot i \cdot \frac{\epsilon}{\varphi^2}\right)|C|$ and (3.229) we get that:

$$|V^{(T_1,\ldots,T_i)} \cap C| \geq \left(1 - \Upsilon \cdot (i+1) \cdot \frac{\epsilon}{\varphi^2}\right)|C|,$$

provided that Υ is bigger than the constant hidden under O notation in (3.229).

□

The following Lemma is a generalization of Theorem 3.7 that uses Lemma 3.45 as an inductive step to show that if ComputeOrderedPartition is called with $\widehat{\mu}$'s that are good approximations to μ's then it returns an ordered partition that induces a good collection of clusters.

Lemma 3.46. *Let $k \geq 2$, $\varphi \in (0,1)$ and $\frac{\epsilon}{\varphi^2} \cdot \log(k)$ be smaller than a sufficiently small constant. Let $G = (V,E)$ be a d-regular graph that admits a (k,φ,ϵ)-clustering $C_1,\ldots,C_k$. Then conditioned on the success of the spectral dot product oracle the following conditions hold.*

If ComputeOrderedPartition*$(G, \widehat{\mu}_1, \widehat{\mu}_2, \ldots, \widehat{\mu}_k, s_1, s_2)$ is called with $(\hat{\mu}_1,\ldots,\hat{\mu}_k)$ such that for every $i \in [k]$ we have $\|\widehat{\mu}_i - \mu_i\|^2 \leq 10^{-12} \cdot \frac{\epsilon}{\varphi^2 \cdot k^2}\|\mu_i\|^2$ then* ComputeOrderedPartition *returns* (True, $(T_1,\ldots,T_b)$) *such that $(T_1,\ldots,T_b)$ induces a collection of clusters $\{\widehat{C}_{\widehat{\mu}_1},\ldots,\widehat{C}_{\widehat{\mu}_k}\}$ such that* *there exists a permutation π on k elements such that for all $i \in [k]$:*

$$\left|\widehat{C}_{\widehat{\mu}_i} \triangle C_{\pi(i)}\right| \le O\left(\frac{\epsilon}{\varphi^3} \cdot \log(k)\right) |C_{\pi(i)}|$$

and

$$\phi(\widehat{C}_{\widehat{\mu}_i}) \le O\left(\frac{\epsilon}{\varphi^2} \cdot \log(k)\right).$$

Proof. Note that for $i = 0$ in the for loop in line 2 of COMPUTEORDEREDPARTITION S and clusters $\{C_1, \ldots, C_k\}$ trivially satisfy assumptions of Lemma 3.45. So using Lemma 3.45 and induction we get that for every $i \in [0..\lceil\log(k)\rceil]$ at the beginning of the i-th iteration:

- $|S| \le k/2^i$,
- for every $\widehat{\mu} \in S$ with corresponding μ and corresponding cluster C we have $|V^{(T_1,\ldots,T_{i-1})} \cap C| \ge \left(1 - \Upsilon \cdot i \cdot \frac{\epsilon}{\varphi^2}\right)|C|$ (where Υ is the constant from the statement of Lemma 3.45).

In particular this means that after $O(\log(k))$ iterations set S becomes empty. This also means that COMPUTEORDEREDPARTITION returns in line 10, so it returns TRUE and the ordered partial partition $(T_1, \ldots, T_b)$ is in fact an ordered partition of $\{\widehat{\mu}_1, \ldots, \widehat{\mu}_k\}$.

Note that by definition (see Definition 3.10) all the approximate clusters $\{\widehat{C}_{\widehat{\mu}_1}, \ldots, \widehat{C}_{\widehat{\mu}_k}\}$ are pairwise disjoint and moreover for every constructed cluster $\widehat{C} \in \{\widehat{C}_{\widehat{\mu}_1}, \ldots, \widehat{C}_{\widehat{\mu}_k}\}$ we have:

$$\phi(\widehat{C}) \le O\left(\frac{\epsilon}{\varphi^2} \cdot \log(k)\right),$$

as it passed the test in line 6 of COMPUTEORDEREDPARTITION. So by Lemma 3.16 it means that there exists a permutation π on k elements such that for all $i \in [k]$:

$$\left|\widehat{C}_{\widehat{\mu}_i} \triangle C_{\pi(i)}\right| \le O\left(\frac{\epsilon}{\varphi^3} \cdot \log(k)\right) |C_{\pi(i)}|.$$

Recall Remark 3.9 for why the proof follows this framework of first arguing about outer-conductance and only after that, using Lemma 3.16, reasoning about symmetric difference. □

Now we present the final Theorem of this section which shows that FINDCENTERS with high probability returns an ordered partition that induces a good collection of clusters. The proof is a careful union bound of error probabilities.

Theorem 3.8. *Let $k \ge 2$, $\varphi \in (0,1)$, $\frac{\epsilon \log(k)}{\varphi^3}$ be smaller than a sufficiently small constant. Let $G = (V, E)$ be a d-regular graph that admits a (k, φ, ϵ)-clustering $C_1, \ldots, C_k$. Then Algorithm 10 with probability $1 - \eta$ returns an ordered partition $(T_1, \ldots, T_b)$ such that $(T_1, \ldots, T_b)$ induces a* *collection of clusters* $\{\widehat{C}_{\widehat{\mu}_1},\dots,\widehat{C}_{\widehat{\mu}_k}\}$ *such that there exists a permutation* π *on* k *elements such that for all* $i \in [k]$*:*

$$\left|\widehat{C}_{\widehat{\mu}_i} \triangle C_{\pi(i)}\right| \le O\left(\frac{\epsilon}{\varphi^3}\cdot\log(k)\right)|C_{\pi(i)}|$$

and

$$\phi(\widehat{C}_{\widehat{\mu}_i}) \le O\left(\frac{\epsilon}{\varphi^2}\cdot\log(k)\right).$$

Moreover

- *Algorithm 10 (*FINDCENTERS*) runs in time*

  $$\widetilde{O}_\varphi\left(\log^2(1/\eta)\cdot 2^{O(\frac{\varphi^2}{\epsilon}\cdot k^4\log^2(k))}\cdot n^{1/2+O(\epsilon/\varphi^2)}\right),$$

  and uses $\widetilde{O}_\varphi\left(\left(\frac{k}{\epsilon}\right)^{O(1)}\cdot n^{1/2+O(\epsilon/\varphi^2)}\right)$ *space,*

- *Algorithm 7 (*HYPERPLANEPARTITIONING*) called with* $(T_1,\dots,T_b)$ *as a parameter runs in time* $\widetilde{O}_\varphi\left(\left(\frac{k}{\epsilon}\right)^{O(1)}\cdot n^{1/2+O(\epsilon/\varphi^2)}\right)$ *per one evaluation.*

Proof. We first prove the runtime guarantee and then we show correctness.

Runtime. The first step of FINDCENTERS (Algorithm 10) is to call INITIALIZEORACLE$(G,1/2)$ (Algorithm 4) which by Lemma 3.43 runs in time $\widetilde{O}_\varphi\left(\left(\frac{k}{\epsilon}\right)^{O(1)}\cdot n^{1/2+O(\epsilon/\varphi^2)}\right)$ and uses $\widetilde{O}_\varphi\left(\left(\frac{k}{\epsilon}\right)^{O(1)}\cdot n^{1/2+O(\epsilon/\varphi^2)}\right)$ space (It's the preprocessing time in the statement of Lemma 3.43). Then Algorithm 10 repeats the following procedure $O(\log(1/\eta))$ times.

It tests all partitions of a set of sampled vertices of size $s = O(\frac{\varphi^2}{\epsilon}\cdot k^4\log(k))$ into k sets. There is at most $k^s = 2^{O(\frac{\varphi^2}{\epsilon}\cdot k^4\log^2(k))}$ of them. Notice that for each partition each $\widehat{\mu}_i$ is defined as

$$\widehat{\mu}_i := \frac{1}{|P_i|}\sum_{x\in P_i} f_x,$$

so as the number of sampled points is $O(\frac{\varphi^2}{\epsilon}\cdot k^4\log(k))$ then each $\widehat{\mu}_i$ is an average of at most $O(\frac{\varphi^2}{\epsilon}\cdot k^4\log(k))$ points. To analyze the runtime notice that:

- For each partition Algorithm 10 runs Algorithm 8,
- Algorithm 8 invokes Algorithm 11 (OUTERCONDUCTANCE) $k^{O(1)}$ times,
- OUTERCONDUCTANCE takes, by Lemma 3.44, $(s_1+s_2)\cdot\frac{1}{\varphi^2}\cdot s^4\cdot\left(\frac{\varphi^2}{\epsilon}k\right)^{O(1)}\cdot n^{1/2+O(\epsilon/\varphi^2)}\log^2(n)$ time,
- $s_1 = \Theta(\frac{\varphi^2}{\epsilon}k^5\log^2(k)\log(1/\eta))$ and $s_2 = \Theta(\frac{\varphi^4}{\epsilon^2}k^5\log^2(k)\log(1/\eta))$.

So in total the runtime of FINDCENTERS is

$$\frac{1}{\varphi^2}\left(\frac{\varphi^2}{\epsilon}k\right)^{O(1)} n^{1/2+O(\epsilon/\varphi^2)}\log^3(n)+\log(1/\eta)2^{O(\frac{\varphi^2}{\epsilon}\cdot k^4\log^2(k))}k^{O(1)}(s_1+s_2)\frac{s^4}{\varphi^2}\left(\frac{\varphi^2}{\epsilon}k\right)^{O(1)} n^{1/2+O(\epsilon/\varphi^2)}\log^2(n)$$

Substituting for s, s_1, s_2 it simplifies to:

$$\frac{1}{\varphi^2}\log^2(1/\eta)\cdot 2^{O(\frac{\varphi^2}{\epsilon}\cdot k^4\log^2(k))}\cdot n^{1/2+O(\epsilon/\varphi^2)}\log^3(n)$$

Runtime of Algorithm 7: Each $\widehat{\mu}_i$ is an average of at most s points, where $s \le O(\frac{\varphi^2}{\epsilon}\cdot k^4\log(k))$, Algorithm 7 performs $k^{O(1)}$ tests $\left\langle f_x, \widehat{\Pi}(\widehat{\mu})\right\rangle_{apx} \ge 0.93||\widehat{\Pi}(\widehat{\mu})||^2$ and by Lemma 3.43 each test takes $\widetilde{O}_\varphi\left(s^4\cdot\left(\frac{k}{\epsilon}\right)^{O(1)}\cdot n^{1/2+O(\epsilon/\varphi^2)}\right)$ time. So in total the runtime of one invokation of CLASSIFYBYHYPERPLANEPARTIONING$(\cdot,(T_1,\dots,T_b))$ is in:

$$\widetilde{O}_\varphi\left(\left(\frac{k}{\epsilon}\right)^{O(1)}\cdot n^{1/2+O(\epsilon/\varphi^2)}\right)$$

Error of OUTERCONDUCTANCE algorithm. Now we analyze the error probabilities of OUTERCONDUCTANCE across all the iterations of our algorithm. Note that we run the test for each cluster for each partition and for each of the $\log(2/\eta)$ iterations of the algorithm. So in total we run OUTERCONDUCTANCE test $2^{O(\frac{\varphi^2}{\epsilon}\cdot k^4\log(k)^2)}k\log\left(\frac{2}{\eta}\right)$ times. By setting s_1 in

$$O\left(k\left(\log(4/\eta)+\log(k\log(1/\eta))+\frac{\varphi^2}{\epsilon}\cdot k^4\log^2(k)\right)\right)\le O\left(\frac{\varphi^2}{\epsilon}\cdot k^5\cdot\log^2(k)\cdot\log(1/\eta)\right),$$

and s_2 in:

$$O\left(\frac{\varphi^2\cdot k}{\epsilon}\left(\log(4/\eta)+\log(k\log(1/\eta))+\frac{\varphi^2}{\epsilon}\cdot k^4\log^2(k)\right)\right)\le O\left(\frac{\varphi^4}{\epsilon^2}\cdot k^5\cdot\log^2(k)\cdot\log(1/\eta)\right),$$

we get by Lemma 3.44 that the probability that the conclusion of Lemma 3.44 is not satisfied in a single run is bounded by

$$\frac{\eta}{100\cdot 2^{\Omega\left(\frac{\varphi^2}{\epsilon}\cdot k^4\log^2(k)\right)}k\log\left(\frac{1}{\eta}\right)}$$

So by union bound over the clusters, the partitions and the iterations we conclude that with probability $1-\frac{\eta}{50}$ the algorithm for every invokation returns a value satisfying the statement of Lemma 3.44. Moreover observe that this also means that $\mathcal{E}_{\text{conductance}}$ is true as conclusions of Lemma 3.44 are stronger than the property required for event $\mathcal{E}_{\text{conductance}}$ to be true.

W.h.p. every returned ordered partition defines a good clustering. By the lower bound on the error probability of OUTERCONDUCTANCE algorithm above we get that with probability $1-\frac{\eta}{50}$ every cluster $\widehat{C}$ that passes the test from line 6 of Algorithm 8 has to satisfy:

$$\phi(\widehat{C}) \leq O\left(\frac{\epsilon}{\varphi^2}\cdot\log(k)\right),$$

as for $\widehat{C}$ to pass the test the value q returned by OUTERCONDUCTANCE has to satisfy $q \leq O\left(\frac{\epsilon}{\varphi^2}\cdot\log(k)\right)$ but by Lemma 3.44 we have $q \geq \frac{1}{2}\phi\left(\widehat{C}_{\widehat{\mu}}^{(T_1,\ldots,T_b)}\right)-\epsilon/\varphi^2$. Now by Lemma 3.16 this implies that if Algorithm 10 returns an ordered partition, then with probability $1-\frac{\eta}{50}$ the collection of clusters it defines satisfies the statement of the Theorem.

Each iteration succeeds with constant probability. In the remaining part of the proof we will show that a clustering is accepted with probability $1-\frac{\eta}{2}$. First note that from the paragraph **Error of OUTERCONDUCTANCE algorithm** we know that $\mathscr{E}_{\text{conductance}}$ holds with probability $1-\frac{\eta}{50}$. Next we show that in each iteration of the outermost for loop of Algorithm 10 it succeeds with probability $1/2$ (conditioned on $\mathscr{E}_{\text{conductance}}$). By amplification this will imply our result.

Now consider one iteration. Let S be the set of sampled vertices. Observe that there exists a partition of $S = P_1 \cup P_2 \cup \cdots \cup P_k$ such that for all $i \in [k]$, $P_i = S \cap C_i$. We set $s = 10^{15}\cdot\frac{\varphi^2}{\epsilon}\cdot k^4\log(k)$. Therefore by Lemma 3.40 with probability at least $\frac{9}{10}$ we have for all $i \in [k]$

$$|S\cap C_i| \geq \frac{0.9\cdot s}{k}\cdot\min_{p,q\in[k]}\frac{|C_p|}{|C_q|} \geq 9\cdot 10^{14}\cdot\frac{\varphi^2}{\epsilon}\cdot k^3\log(k).$$

Let $\delta = k^{-50}$ and $\zeta = \frac{10^{-6}\sqrt{\epsilon}}{\varphi\cdot k}$. Therefore, we have

$$|S\cap C_i| \geq 9\cdot 10^{14}\cdot\frac{\varphi^2}{\epsilon}\cdot k^3\log(k) \geq c\cdot\left(k\cdot\log\left(\frac{k}{\delta}\right)\cdot\left(\frac{1}{\delta}\right)^{(80\cdot\epsilon/\varphi^2)}\cdot\left(\frac{1}{\zeta}\right)^2\right)^{1/(1-(80\cdot\epsilon/\varphi^2))}$$

where c is the constant from Lemma 3.39. The last inequality holds since $\frac{\epsilon}{\varphi^2}\log(k)$ is smaller than a sufficiently small constant, hence, $\left(\frac{\varphi^2}{\epsilon}\right)^{(\epsilon/\varphi^2)} \in O(1)$, and $k^{(\epsilon/\varphi^2)} \in O(1)$. Therefore by Lemma 3.39 for all $i \in [k]$ with probability at least $1-k^{-50}$ we have:

$$\|\widehat{\mu}_i-\mu_i\|_2 \leq \zeta\cdot\|\mu_i\|_2 = \frac{10^{-6}\sqrt{\epsilon}}{\varphi\cdot k}\|\mu_i\|_2.$$

Hence, by union bound over all sets P_i, with probability at least $\frac{9}{10}-k\cdot k^{-50} \geq \frac{7}{8}$ we get $\|\widehat{\mu}_i-\mu_i\|_2 \leq \frac{10^{-6}\sqrt{\epsilon}}{\varphi\cdot k}\|\mu_i\|_2$ for all $i \in [k]$ simultaneously.

Now by Theorem 3.2 and the union bound we get that spectral dot product oracle succeeds with probability $1-n^{-48}$. So by Lemma 3.46 and the union bound FINDCENTERS with probability $\frac{7}{8}-n^{-48} \geq \frac{1}{2}$ returns an ordered partition $(T_1,\ldots,T_b)$ which induces a collection of clusters

$\{\widehat{C}_{\widehat{\mu}_1}, \dots, \widehat{C}_{\widehat{\mu}_k}\}$ such that there exists a permutation π on k elements such that for all $i \in [k]$:

$$\left|\widehat{C}_{\widehat{\mu}_i} \triangle C_{\pi(i)}\right| \leq O\left(\frac{\epsilon}{\varphi^3} \cdot \log(k)\right) |C_{\pi(i)}|$$

and

$$\phi(\widehat{C}_{\widehat{\mu}_i}) \leq O\left(\frac{\epsilon}{\varphi^2} \cdot \log(k)\right).$$

□

3.6.6 LCA

Now we prove the main result of the paper. Recall that a clustering oracle (Definition 3.4) is a randomized algorithm that when given query access to a d-regular graph $G = (V, E)$ that admits (k, φ, ϵ)-clustering $C_1, \dots, C_k$ it provides consistent access to a **partition** $\widehat{C}_1, \dots, \widehat{C}_k$ such that there exists a permutation π on k elements such that for all $i \in [k]$:

$$\left|\widehat{C}_{\widehat{\mu}_i} \triangle C_{\pi(i)}\right| \leq O\left(\frac{\epsilon}{\varphi^3} \cdot \log(k)\right) |C_{\pi(i)}|. \tag{3.230}$$

Consistency means that a vertex $x \in V$ is classified in the same way every time it is queried.

First we will show a Proposition (Proposition 3.3) that shows that it is enough to design an algorithm that returns a **collection of disjoint clusters** (not necessarily a partition) that satisfies (3.230) to get a clustering oracle. Using this Proposition as a reduction we then show Theorem 3.3, which is the main Theorem of the paper.

Proposition 3.3. *If there exists a randomized algorithm $\mathcal{O}$ that when given query access to a d-regular graph $G = (V, E)$ that admits a (k, φ, ϵ)-clustering $C_1, \dots, C_k$, the algorithm $\mathcal{O}$ provides consistent query access to a* ***collection of disjoint clusters*** *$\mathscr{C} = (\widehat{C}_1, \dots, \widehat{C}_k)$ of V. The collection $\mathscr{C}$ is determined solely by G and the algorithm's random seed. Moreover, with probability at least $9/10$ over the random bits of $\mathcal{O}$ the collection $\mathscr{C}$ has the following property: for some permutation π on k elements one has for every $i \in [k]$:*

$$|C_i \triangle \widehat{C}_{\pi(i)}| \leq O\left(\frac{\epsilon}{\varphi^3}\right) |C_i|.$$

Then if clusters have equal sizes and $\frac{\epsilon \cdot n}{\varphi^3 \cdot k \log(k)}$ is bigger than a constant then there exists an algorithm $\mathcal{O}'$ that is a (k, φ, ϵ)-clustering oracle with the same running time and space up to constant factors.

Proof. The idea is to assign the points outside $\bigcup_{i \in [k]} \widehat{C}_i$ randomly. That is to assign vertex $x \in V$, $\mathcal{O}'$ works exactly the same like $\mathcal{O}$ but if $\mathcal{O}$ left x unassigned then $\mathcal{O}'$ assigns x to a value chosen from $[k]$ uniformly at random.

Let $R = V \setminus \bigcup_{i \in [k]} \widehat{C}_i$ and for every $i \in [k]$ let $S_i \subseteq R$ be the set of vertices that were randomly assigned to $\widehat{C}_i$. By the fact that for every $i \in [k]$ $|C_i \triangle \widehat{C}_{\pi(i)}| \leq O\left(\frac{\epsilon}{\varphi^3}\right)|C_i|$ we get that there exists a constant C such that:

$$|R| \leq C \cdot \frac{\epsilon}{\varphi^3} \cdot n. \tag{3.231}$$

Now let $i \in [k]$. By the Chernoff bound we have that for every $\delta \geq 1$:

$$P\left[\left||S_i| - \frac{|R|}{k}\right| \geq \delta \frac{|R|}{k}\right] \leq e^{-\delta \frac{|R|}{3 \cdot k}} \tag{3.232}$$

Setting $\delta = \frac{C \cdot \epsilon \cdot n}{\varphi^3 \cdot |R|}$ we get:

$$P\left[\left||S_i| - \frac{|R|}{k}\right| \geq C \cdot \frac{\epsilon}{\varphi^3} \cdot \frac{n}{k}\right] \leq e^{-\frac{C \cdot \epsilon \cdot n}{3 \cdot \varphi^3 \cdot k}} \tag{3.233}$$

Combining (3.231) and (3.233) and the assumption that $\frac{\epsilon \cdot n}{\varphi^3 \cdot k \log(k)}$ is bigger than a constant we get that

$$P\left[|S_i| \geq 2C \cdot \frac{\epsilon}{\varphi^3} \cdot \frac{n}{k}\right] \leq \frac{1}{100 \cdot k}$$

Using the union bound we get that with probability $9/10 - k \cdot \frac{1}{100 \cdot k} \geq 8/10$ we have that for every $i \in [k]$ $|S_i| \leq 2C \cdot \frac{\epsilon}{\varphi^3} \cdot \frac{n}{k}$. So finally with probability $8/10$ for every $i \in [k]$:

$$\begin{aligned}
|C_i \triangle (\widehat{C}_{\pi(i)} \cup S_{\pi(i)})| &\leq |C_i \triangle \widehat{C}_{\pi(i)}| + |S_{\pi(i)}| \\
&\leq O\left(\frac{\epsilon}{\varphi^3}\right) \cdot |C_i| + O\left(\frac{\epsilon}{\varphi^3}\right) \cdot \frac{n}{k} && \text{By definition of } \mathcal{O} \\
&\leq O\left(\frac{\epsilon}{\varphi^3}\right) \cdot |C_i| && \text{As } \frac{\max_{p \in [k]} |C_p|}{\min_{p \in [k]} |C_p|} = O(1),
\end{aligned}$$

which means that $\mathcal{O}'$ is a (k, φ, ϵ)-clustering oracle. □

Theorem 3.3. *For every integer $k \geq 2$, every $\varphi \in (0,1)$, every $\epsilon \ll \frac{\varphi^3}{\log k}$, every $\delta \in (0, 1/2]$ there exists a (k, φ, ϵ)-clustering oracle that:*

- *has $\widetilde{O}_\varphi\left(2^{O\left(\frac{\varphi^2}{\epsilon} k^4 \log^2(k)\right)} \cdot n^{1-\delta+O(\epsilon/\varphi^2)}\right)$ preprocessing time,*
- *has $\widetilde{O}_\varphi\left(\left(\frac{k}{\epsilon}\right)^{O(1)} \cdot n^{\delta+O(\epsilon/\varphi^2)}\right)$ query time,*
- *uses $\widetilde{O}_\varphi\left(\left(\frac{k}{\epsilon}\right)^{O(1)} \cdot n^{1-\delta+O(\epsilon/\varphi^2)}\right)$ space,*
- *uses $\widetilde{O}_\varphi\left(\left(\frac{k}{\epsilon}\right)^{O(1)} \cdot n^{O(\epsilon/\varphi^2)}\right)$ random bits,*

where O_φ suppresses dependence on φ and $\widetilde{O}$ hides all polylog(n) *factors.*

Proof. By Theorem 3.8 we get that there exists an algorithm that runs in $\widetilde{O}_\varphi\left(2^{O(\frac{\varphi^2}{\epsilon}\cdot k^4\log^2(k))}\cdot n^{1/2+O(\epsilon/\varphi^2)}\right)$ time and that with probability 9/10 returns an ordered partition $(T_1,\ldots,T_b)$ of $\{\widehat{\mu}_1,\ldots,\widehat{\mu}_k\}$ such that the induced collection of clusters $\{\widehat{C}_{\widehat{\mu}_1},\ldots,\widehat{C}_{\widehat{\mu}_k}\}$ satisfies the following. There exists a permutation π on k elements such that for every $i\in[1,\ldots,k]$:

$$|C_{\pi(i)}\triangle\widehat{C}_{\widehat{\mu}_i}|\le O\left(\frac{\epsilon}{\varphi^3}\cdot\log(k)\right)|C_{\pi(i)}|$$

That algorithm is the preprocessing step of oracle $\mathcal{O}$. Then for each query $x_i\in V$ we run Algorithm 7 which outputs $\widehat{\mu}_j$ such that $x_i\in\widehat{C}_{\widehat{\mu}_j}$ (Note that x_i might not belong to any of $\widehat{C}_{\widehat{\mu}_i}$, see Proposition 3.3 for how to deal with that). Algorithm 7 by Theorem 3.8 runs in $\widetilde{O}_\varphi\left(\left(\frac{k}{\epsilon}\right)^{O(1)}\cdot n^{1/2+O(\epsilon/\varphi^2)}\right)$ time.

Runtime tradeoff. Notice however that by Theorem 3.2 we can achieve a tradeoff in the preprocessing/query runtime and achieve $\widetilde{O}_\varphi\left(2^{O(\frac{\varphi^2}{\epsilon}\cdot k^4\log^2(k))}\cdot n^{1-\delta+O(\epsilon/\varphi^2)}\right)$ for preprocessing time and $\widetilde{O}_\varphi(\left(\frac{k}{\epsilon}\right)^{O(1)}\cdot n^{1-\delta+O(\epsilon/\varphi^2)})$ space and $\widetilde{O}_\varphi\left(\left(\frac{k}{\epsilon}\right)^{O(1)}\cdot n^{\delta+O(\epsilon/\varphi^2)}\right)$ for query time.

Random bits. The only thing left to prove is to show that we can implement these two algorithms in LCA model using few random bits. There are couple of places in our Algorithms where we use randomness.

First in InitializeOracle (Algorithm 4) we sample $\widetilde{\Theta}(n^{O(\epsilon/\varphi^2)}\cdot k^{O(1)})$ random points. For that we need $\widetilde{\Theta}(n^{O(\epsilon/\varphi^2)}\cdot k^{O(1)})$ random bits.

For generating random walks in Algorithm 4 and Algorithm 5 we need the following number of random bits. Notice that in all the proofs (see Lemma 3.26) we only need 4-wise independence of random walks. That means that we can implement generating these random walks using a hash function $h(x)$ that for vertex $x\in V$ generates $O(\log(d)\cdot\frac{1}{\varphi^2}\cdot\log(n))$ bit string that can be interpreted as encoding a random walk of length $O(\frac{1}{\varphi^2}\cdot\log(n))$ (remember that graphs we consider are d-regular so $\log(d)$ bits is enough to encode a neighbour). It's enough for the hash function to be 4-wise independent so it can be implemented using $O(\frac{1}{\varphi^2}\cdot\log(d)\cdot\log(n))=\widetilde{O}_\varphi(1)$ random bits.

The partitioning scheme (see Algorithm 7) works in $O(\log(k))$ adaptive stages. The stages are adaptive, that is why we use fresh randomness in every stage. For a single stage we observe that in the proof of Lemma 3.44 we only use Chernoff type bounds. So by [SSS93] we don't need fully independent random variables. In our case it's enough to have $O(\log(n))$-wise independent random variables which can be implemented as hash functions using $O(\log^2(n))$ random bits. This means that in total we need $O(\log(k)\log^2(n))=\widetilde{O}(1)$ random bits for this.

For sampling set S in Algorithm 10 we can use $O(\frac{\varphi^2}{\epsilon}\cdot k^4\log(k)\cdot\log(n))=\widetilde{O}_\varphi(\frac{1}{\epsilon}\cdot k^{O(1)})$ fresh random bits.

So finally the total number of random bits we need is in:

$$\widetilde{O}_{\varphi}\left(n^{O(\epsilon/\varphi^2)} \cdot k^{O(1)} + 1 + 1 + \frac{1}{\epsilon} \cdot k^{O(1)}\right) \leq \widetilde{O}_{\varphi}\left(\frac{1}{\epsilon} \cdot n^{O(\epsilon/\varphi^2)} \cdot k^{O(1)}\right)$$

□

Remark 3.11. *Note that threshold sets $C_{y,\theta}$ (recall Definition 3.8) are well defined in LCA model because for all $x, y \in V$ whenever we compute $\left\langle f_x, f_y \right\rangle_{apx}$ the result is the same as we use consistent randomness (see Definition 3.4).*

4 Conclusion

In this thesis, we have introduced new spectral techniques for understanding the cluster structure of graphs in sublinear time. In Chapter 2, we have developed an optimal sublinear algorithm for testing k-clusterability in the property testing framework. Next, in Chapter 3, we have extended our testing result to an efficient clustering algorithm in the LCA model that misclassifies a small fraction of vertices in every cluster. These results have resolved important graph clustering problems while, at the same time, opening several directions for further progress.

An interesting open problem would be to go beyond *flat-clustering* regime by generalizing the spectral clustering techniques and applying them to the *hierarchical clustering* problems. Hierarchical clustering is the task of partitioning vertices of a graph into nested clusters. Dasgupta introduced an objective function for formulating hierarchical clustering and initiated a line of work developing algorithms that optimize the cost of the hierarchical tree solution [Das16]. This leads to a natural open problem: Given a k-clusterable graph, can we design a local computation algorithm that recovers a hierarchical structure of the graph in sublinear time?

Another research direction is to generalize the spectral clustering results to more robust settings in which the input graph might contain noisy structures such as noisy clusters [Pen20], or the graph might be obtained by *semi-random* models [MMV12]. The existence of noisy structures is closer to real-world applications and modern data analysis require new techniques to quickly detect clusters in the noisy datasets.

Designing robust clustering algorithms, prompt us to address clustering problems in dynamic settings. In most applications, the graph is changing over time and the communities evolve dynamically (for example in social networks, or web graph). We would like to maintain a reasonable clustering solution of the graph at all times. The goal is to design a framework that maintains a *good* clustering of the graph with fast update time and query time. In Chapter 3, we have designed a sublinear algorithm for graph clustering when the external conductance of clusters i.e., ϵ is bounded by $O(1/\log k)$. This result leads to a fully dynamic algorithm for graph clustering with $n^{1/2+O(\epsilon)}$ update and recovery time. Can we decrease the update and recovery time to $n^{o(1)}$ while respecting the external conductance assumption?

Finally, another open problem is to develop *differentially-private* sublinear algorithms for spectral clustering in which the goal is clustering the vertices of a graph while preserving the privacy of individuals in the dataset.

A Supplementary Materials for Chapter 1

A.1 Proof Lemma 2.4

Proof of Lemma 2.4. For the first part, let $V^{\perp}$ matrix whose columns complete the columns of V to an orthonormal basis. Then VV^T projects a vector onto the column space of V, and $(V^{\perp})(V^{\perp})^T$ projects onto the columns space of $V^{\perp}$, which is also the orthogonal complement of the column space of V. Therefore, $VV^T + (V^{\perp})(V^{\perp})^T = I_{m\times m}$. Thus,

$$\mu_h(A^T A) \geq \mu_h(A^T VV^T A) + \mu_{\min}(A^T V^{\perp}(V^{\perp})^T A) \geq \mu_h(A^T VV^T A)$$

where the first inequality follows from Weyl's inequality, and the second one holds because $A^T V^{\perp}(V^{\perp})^T A$ is positive semidefinite.

The second part follows from the first part by observing that $\mu_h(U^T A^T AU) = \mu_h(AUU^T A^T)$ and $\mu_h(A^T A) = \mu_h(AA^T)$. □

A.2 Proof of Lemmas from Section 2.3.2

Lemma 2.19 (restated). Let $G = (V_G, E_G)$ be a graph. Let $0 < \sigma \leq 1$ $t > 0$, $\mu_{\text{err}} > 0$, k be an integer, and let S be a multiset of s vertices, all whose elements are (σ, t)-good. Let

$$R = \max\left(\frac{100 s^2 \sigma^{1/2}}{\mu_{\text{err}}}, \frac{200 s^4 \sigma^{3/2}}{\mu_{\text{err}}^2}\right).$$

For each $a \in S$ and each $b \in V_G$, let $\mathbf{q}_a(b)$ and $\mathbf{q}'_a(b)$ be random variables which denote the fraction out of the R random walks starting from a, which end in b. Let Q and Q' be matrices whose columns are $(D^{-\frac{1}{2}}\mathbf{q}_a)_{a\in S}$ and $(D^{-\frac{1}{2}}\mathbf{q}'_a)_{a\in S}$ respectively. Let $\mathcal{G} = \frac{1}{2}\left(Q^T Q' + Q'^T Q\right)$. Then with probability at least $49/50$, $|\mu_{k+1}(\mathcal{G}) - \mu_{k+1}((D^{-\frac{1}{2}}M^t S)^T (D^{-\frac{1}{2}}M^t S))| \leq \mu_{\text{err}}$.

Proof. Let $X^i_{a,r}$ be a random variable which is $\frac{1}{\sqrt{\deg(i)}}$ if the r^{th} random walk starting from a, ends at vertex i, and 0 otherwise. Thus, $\mathbb{E}[X^i_{a,r}] = \frac{\mathbf{p}^t_a(i)}{\sqrt{\deg(i)}}$. For any two vertices $a, b \in S$, observe that the entry $\mathcal{G}_{a,b}$ is a random variable given by

$$\mathcal{G}_{a,b} = \frac{1}{R^2} \sum_{i \in V_G} (\sum_{r_1=1}^{R} X^i_{a,r_1})(\sum_{r_2=1}^{R} X^i_{b,r_2}).$$

Thus,

$$\begin{aligned}\mathbb{E}[\mathcal{G}_{a,b}] &= \frac{1}{R^2} \sum_{i \in V_G} (\sum_{r_1=1}^{R} \mathbb{E}[X^i_{a,r_1}])(\sum_{r_2=1}^{R} \mathbb{E}[X^i_{b,r_2}]) \\ &= \sum_{i \in V_G} \frac{\mathbf{p}^t_a(i)}{\sqrt{\deg(i)}} \cdot \frac{\mathbf{p}^t_b(i)}{\sqrt{\deg(i)}} = (D^{-\frac{1}{2}} M^t \mathbb{1}_a)^T (D^{-\frac{1}{2}} M^t \mathbb{1}_b). \qquad \text{(A.1)}\end{aligned}$$

We know that $\mathrm{Var}(\mathcal{G}_{a,b}) = \mathbb{E}[\mathcal{G}^2_{a,b}] - \mathbb{E}[\mathcal{G}_{a,b}]^2$. Let us first compute $\mathbb{E}[\mathcal{G}^2_{a,b}]$.

$$\begin{aligned}\mathbb{E}[\mathcal{G}^2_{a,b}] &= \mathbb{E}\left[\frac{1}{R^4} \sum_{i \in V_G} \sum_{j \in V_G} \sum_{r_1=1}^{R} \sum_{r_2=1}^{R} \sum_{r_1'=1}^{R} \sum_{r_2'=1}^{R} X^i_{a,r_1} X^i_{b,r_2} X^j_{a,r_1'} X^j_{b,r_2'}\right] \\ &= \frac{1}{R^4} \sum_{i \in V_G} \sum_{j \in V_G} \sum_{r_1=1}^{R} \sum_{r_2=1}^{R} \sum_{r_1'=1}^{R} \sum_{r_2'=1}^{R} \mathbb{E}[X^i_{a,r_1} X^i_{b,r_2} X^j_{a,r_1'} X^j_{b,r_2'}]\end{aligned}$$

To compute $\mathbb{E}[X^i_{a,r_1} X^i_{b,r_2} X^j_{a,r_1'} X^j_{b,r_2'}]$, we need to consider the following cases.

1. $i \neq j$: $\mathbb{E}[X^i_{a,r_1} X^i_{b,r_2} X^j_{a,r_1'} X^j_{b,r_2'}] \leq \frac{\mathbf{p}^t_a(i)}{\sqrt{\deg(i)}} \cdot \frac{\mathbf{p}^t_b(i)}{\sqrt{\deg(i)}} \cdot \frac{\mathbf{p}^t_a(j)}{\sqrt{\deg(j)}} \cdot \frac{\mathbf{p}^t_b(j)}{\sqrt{\deg(j)}}$. (This is an equality if $r_1 \neq r_1'$ and $r_2 \neq r_2'$. Otherwise, the expectation is zero.)

2. $i = j, \quad r_1 = r_1', \quad r_2 = r_2'$: $\mathbb{E}[X^i_{a,r_1} X^i_{b,r_2} X^j_{a,r_1'} X^j_{b,r_2'}] = \frac{\mathbf{p}^t_a(i)}{\sqrt{\deg(i)}} \cdot \frac{\mathbf{p}^t_b(i)}{\sqrt{\deg(i)}} \cdot \frac{1}{\sqrt{\deg(i)}} \cdot \frac{1}{\sqrt{\deg(i)}}$.

3. $i = j, \quad r_1 = r_1', \quad r_2 \neq r_2'$: $\mathbb{E}[X^i_{a,r_1} X^i_{b,r_2} X^j_{a,r_1'} X^j_{b,r_2'}] = \frac{\mathbf{p}^t_a(i)}{\sqrt{\deg(i)}} \cdot \frac{\mathbf{p}^t_b(i)}{\sqrt{\deg(i)}} \cdot \frac{1}{\sqrt{\deg(i)}} \cdot \frac{\mathbf{p}^t_b(i)}{\sqrt{\deg(i)}}$.

4. $i = j, \quad r_1 \neq r_1', \quad r_2 = r_2'$: $\mathbb{E}[X^i_{a,r_1} X^i_{b,r_2} X^j_{a,r_1'} X^j_{b,r_2'}] = \frac{\mathbf{p}^t_a(i)}{\sqrt{\deg(i)}} \cdot \frac{\mathbf{p}^t_b(i)}{\sqrt{\deg(i)}} \cdot \frac{\mathbf{p}^t_a(i)}{\sqrt{\deg(i)}} \cdot \frac{1}{\sqrt{\deg(i)}}$.

5. $i = j, \quad r_1 \neq r_1', \quad r_2 \neq r_2'$: $\mathbb{E}[X^i_{a,r_1} X^i_{b,r_2} X^j_{a,r_1'} X^j_{b,r_2'}] = \frac{\mathbf{p}^t_a(i)}{\sqrt{\deg(i)}} \cdot \frac{\mathbf{p}^t_b(i)}{\sqrt{\deg(i)}} \cdot \frac{\mathbf{p}^t_a(i)}{\sqrt{\deg(i)}} \cdot \frac{\mathbf{p}^t_b(i)}{\sqrt{\deg(i)}}$.

Thus we have,

$$\begin{aligned}
\mathbb{E}[Z_{a,b}^2] &= \frac{1}{R^4}\sum_{i\in V_G}\sum_{j\in V_G}\sum_{r_1=1}^{R}\sum_{r_2=1}^{R}\sum_{r_1'=1}^{R}\sum_{r_2'=1}^{R}\mathbb{E}[X_{a,r_1}^i X_{b,r_2}^i X_{a,r_1'}^j X_{b,r_2'}^j]\\
&\le \sum_{i\in V_G}\sum_{j\in V_G\setminus\{i\}}\frac{\mathbf{p}_a^t(i)\cdot\mathbf{p}_a^t(j)\cdot\mathbf{p}_b^t(i)\cdot\mathbf{p}_b^t(j)}{\deg(i)\cdot\deg(j)}+\sum_{i\in V_G}\frac{\mathbf{p}_a^t(i)^2\cdot\mathbf{p}_b^t(i)^2}{\deg(i)^2}\\
&+\frac{1}{R^2}\sum_{i\in V_G}\frac{\mathbf{p}_a^t(i)\cdot\mathbf{p}_b^t(i)}{\deg(i)^2}+\frac{1}{R}\sum_{i\in V_G}\frac{\mathbf{p}_a^t(i)\cdot\mathbf{p}_b^t(i)^2}{\deg(i)^2}+\frac{1}{R}\sum_{i\in V_G}\frac{\mathbf{p}_a^t(i)^2\cdot\mathbf{p}_b^t(i)}{\deg(i)^2}\\
&= \sum_{i,j\in V_G}\frac{\mathbf{p}_a^t(i)\cdot\mathbf{p}_a^t(j)\cdot\mathbf{p}_b^t(i)\cdot\mathbf{p}_b^t(j)}{\deg(i)\cdot\deg(j)}+\frac{1}{R^2}\sum_{i\in V_G}\frac{\mathbf{p}_a^t(i)\cdot\mathbf{p}_b^t(i)}{\deg(i)^2}\\
&+\frac{1}{R}\sum_{i\in V_G}\frac{\mathbf{p}_a^t(i)\cdot\mathbf{p}_b^t(i)\cdot(\mathbf{p}_a^t(i)+\mathbf{p}_b^t(i))}{\deg(i)^2}.
\end{aligned}$$

Therefore we get,

$$\begin{aligned}
\mathrm{Var}(\mathcal{G}_{a,b}) &= \mathbb{E}[\mathcal{G}_{a,b}^2]-\mathbb{E}[\mathcal{G}_{a,b}]^2\\
&\le \sum_{i,j\in V_G}\frac{\mathbf{p}_a^t(i)\cdot\mathbf{p}_a^t(j)\cdot\mathbf{p}_b^t(i)\cdot\mathbf{p}_b^t(j)}{\deg(i)\cdot\deg(j)}+\frac{1}{R^2}\sum_{i\in V_G}\frac{\mathbf{p}_a^t(i)\cdot\mathbf{p}_b^t(i)}{\deg(i)^2}\\
&+\frac{1}{R}\sum_{i\in V_G}\frac{\mathbf{p}_a^t(i)\cdot\mathbf{p}_b^t(i)\cdot(\mathbf{p}_a^t(i)+\mathbf{p}_b^t(i))}{\deg(i)^2}-\left(\sum_{i\in V_G}\frac{\mathbf{p}_a^t(i)\cdot\mathbf{p}_b^t(i)}{\deg(i)}\right)^2\\
&= \frac{1}{R^2}\sum_{i\in V_G}\frac{\mathbf{p}_a^t(i)\cdot\mathbf{p}_b^t(i)}{\deg(i)^2}+\frac{1}{R}\sum_{i\in V_G}\frac{\mathbf{p}_a^t(i)^2\cdot\mathbf{p}_b^t(i)}{\deg(i)^2}+\frac{1}{R}\sum_{i\in V_G}\frac{\mathbf{p}_a^t(i)\cdot\mathbf{p}_b^t(i)^2}{\deg(i)^2}\\
&\le \frac{1}{R^2}\sum_{i\in V_G}\frac{\mathbf{p}_a^t(i)}{\sqrt{\deg(i)}}\cdot\frac{\mathbf{p}_b^t(i)}{\sqrt{\deg(i)}}\\
&+\frac{1}{R}\sum_{i\in V_G}\left(\frac{\mathbf{p}_a^t(i)}{\sqrt{\deg(i)}}\right)^2\cdot\frac{\mathbf{p}_b^t(i)}{\sqrt{\deg(i)}}+\frac{1}{R}\sum_{i\in V_G}\frac{\mathbf{p}_a^t(i)}{\sqrt{\deg(i)}}\cdot\left(\frac{\mathbf{p}_b^t(i)}{\sqrt{\deg(i)}}\right)^2\\
&\le \frac{1}{R^2}||D^{-\frac{1}{2}}\mathbf{p}_a^t||_2\cdot||D^{-\frac{1}{2}}\mathbf{p}_b^t||_2+\frac{1}{R}||D^{-\frac{1}{2}}\mathbf{p}_a^t||_4^2\cdot||D^{-\frac{1}{2}}\mathbf{p}_b^t||_2+\frac{1}{R}||D^{-\frac{1}{2}}\mathbf{p}_a^t||_2\cdot||D^{-\frac{1}{2}}\mathbf{p}_b^t||_4^2\\
&\le \frac{1}{R^2}||D^{-\frac{1}{2}}\mathbf{p}_a^t||_2\cdot||D^{-\frac{1}{2}}\mathbf{p}_b^t||_2+\frac{1}{R}||D^{-\frac{1}{2}}\mathbf{p}_a^t||_2^2\cdot||D^{-\frac{1}{2}}\mathbf{p}_b^t||_2+\frac{1}{R}||D^{-\frac{1}{2}}\mathbf{p}_a^t||_2\cdot||D^{-\frac{1}{2}}\mathbf{p}_b^t||_2^2
\end{aligned}\tag{A.2}$$

Notice that all vertices in S are (σ,t)-good, therefore we get,

$$\mathrm{Var}(\mathcal{G}_{a,b})\le\frac{\sigma}{R^2}+\frac{2\sigma^{3/2}}{R}.$$

Then by Chebyshev's inequality, we get,

$$\Pr\left[|\mathscr{G}_{a,b}-\mathbb{E}[\mathscr{G}_{a,b}]|>\frac{\mu_{\mathrm{err}}}{s}\right]<\frac{\mathrm{Var}[\mathscr{G}_{a,b}]}{(\frac{\mu_{\mathrm{err}}}{s})^2}\leq\frac{s^2}{\mu_{\mathrm{err}}^2}\left(\frac{\sigma}{R^2}+\frac{2\sigma^{3/2}}{R}\right)\leq\frac{1}{50s^2},$$

where the last inequality follows by our choice of R. By the union bound, with probability at least $49/50$, we have for all $a, b \in S$,

$$|\mathscr{G}_{a,b}-((D^{-\frac{1}{2}}M^t)^T(D^{-\frac{1}{2}}M^t))_{a,b}|=|Z_{a,b}-\mathbb{E}[Z_{a,b}]|\leq\frac{\mu_{\mathrm{err}}}{s},$$

which implies $\|\mathscr{G}-(D^{-\frac{1}{2}}M^tS)^T(D^{-\frac{1}{2}}M^tS)\|_F\leq\mu_{\mathrm{err}}$. This, in turn, implies

$$|\mu_{k+1}(\mathscr{G})-\mu_{k+1}((D^{-\frac{1}{2}}M^tS)^T(D^{-\frac{1}{2}}M^tS))|\leq\mu_{\mathrm{err}},$$

due to Weyl's inequality and the fact that the Frobenius norm of a matrix bounds its maximum eigenvalue from above. □

Lemma 2.20 (restated). For all $0<\alpha<1$, and all $G=(V_G,E_G)$ which is $(k,\varphi_{\mathrm{in}})$-clusterable, there exists $V'_G\subseteq V_G$ with $\mathrm{vol}(V'_G)\geq(1-\alpha)\mathrm{vol}(V_G)$ such that for any $t\geq\frac{2\ln(\mathrm{vol}(V_G))}{\varphi_{\mathrm{in}}^2}$, every $u\in V'_G$ is $\left(\frac{2k}{\alpha\cdot\mathrm{vol}(V_G)},t\right)$-good.

Proof. Recall that we say that vertex u is (σ,t)*-good* if $\|D^{-\frac{1}{2}}\mathbf{p}_u^t\|_2^2\leq\sigma$. We can write $D^{-\frac{1}{2}}\mathbf{p}_u^t$ as

$$D^{-\frac{1}{2}}\mathbf{p}_u^t=D^{-\frac{1}{2}}M^t\mathbb{1}_u=D^{-\frac{1}{2}}(D^{\frac{1}{2}}\overline{M}^tD^{-\frac{1}{2}})\mathbb{1}_u=\overline{M}^tD^{-\frac{1}{2}}\mathbb{1}_u$$

Recall from section 3.2 that $1-\frac{\lambda_1}{2}\geq\cdots\geq1-\frac{\lambda_n}{2}$, are eigenvalues of $\overline{M}$, and $v_1,\ldots,v_n$ are the corresponding orthonormal eigenvectors. We write $D^{-\frac{1}{2}}\mathbb{1}_u$ in the eigenbasis of $\overline{M}$ as $D^{-\frac{1}{2}}\mathbb{1}_u=\sum_{i=1}^n\alpha_i(u)\cdot v_i$ where $\alpha_i(u)=(D^{-\frac{1}{2}}\mathbb{1}_u)^Tv_i=\frac{v_i(u)}{\sqrt{\deg(u)}}$. Therefore we get,

$$\begin{aligned}||D^{-\frac{1}{2}}\mathbf{p}_u^t||_2^2&=||\overline{M}^tD^{-\frac{1}{2}}\mathbb{1}_u||_2^2\\&=\sum_{i=1}^n\alpha_i(u)^2\left(1-\frac{\lambda_i}{2}\right)^{2t}\\&=\sum_{i=1}^k\alpha_i(u)^2\left(1-\frac{\lambda_i}{2}\right)^{2t}+\sum_{i=k+1}^n\alpha_i(u)^2\left(1-\frac{\lambda_i}{2}\right)^{2t}\\&\leq\sum_{i=1}^k\alpha_i(u)^2+\left(1-\frac{\lambda_{k+1}}{2}\right)^{2t}\sum_{i=k+1}^n\alpha_i(u)^2\\&\leq\sum_{i=1}^k\alpha_i(u)^2+\left(1-\frac{\varphi_{\mathrm{in}}^2}{4}\right)^{2t}.\end{aligned}$$

The last inequality follows from Lemma 3.1, and the fact that $\sum_{i=k+1}^n\alpha_i(u)^2\leq||v_i||_2^2\leq1$. We now bound $h(u):=\sum_{i=1}^k\alpha_i(u)^2$. Let $\mathscr{D}$ denote the degree distribution of G (i.e., $\mathscr{D}(v)=\frac{\deg(v)}{\mathrm{vol}(G)}$).

Observe that

$$\begin{aligned}\mathbb{E}_{\mathcal{D}}\left[h(u)\right] &= \sum_{u\in V_G}\frac{\deg(u)}{\mathrm{vol}(V_G)}\cdot\left(\sum_{i=1}^{k}\alpha_i(u)^2\right) = \sum_{u\in V_G}\frac{\deg(u)}{\mathrm{vol}(V_G)}\cdot\left(\sum_{i=1}^{k}\frac{v_i(u)^2}{\deg(u)}\right)\\ &= \frac{1}{\mathrm{vol}(V_G)}\sum_{i=1}^{k}\sum_{u\in V_G}v_i(u)^2 = \frac{1}{\mathrm{vol}(V_G)}\sum_{i=1}^{k}\|v_i\|_2^2 = \frac{k}{\mathrm{vol}(V_G)}\end{aligned}$$

Thus by Markov's inequality there exists a set $V'_G \subseteq V_G$ with $\mathrm{vol}(V'_G) \geq (1-\alpha)\mathrm{vol}(V_G)$ such that for any $u \in V'_G$,

$$h(u) \leq \frac{1}{\alpha}\cdot\frac{k}{\mathrm{vol}(V_G)}\ .$$

Thus if $t \geq \frac{2\ln(\mathrm{vol}(V_G))}{\varphi_{\mathrm{in}}^2}$ for any $u \in V'_G$ we have

$$\|D^{-\frac{1}{2}}\mathbf{p}_u^t\|_2^2 \leq \frac{k}{\alpha\cdot\mathrm{vol}(V_G)} + \left(1-\frac{\varphi_{\mathrm{in}}^2}{4}\right)^{2t} \leq \frac{2k}{\alpha\cdot\mathrm{vol}(V_G)},$$

therefore every $u \in V'_G$ is $\left(\frac{2k}{\alpha\cdot\mathrm{vol}(V_G)}, t\right)$-good. □

Lemma 2.18 (restated). Let $G = (V_G, E_G)$. Let $a \in V_G$, $\sigma > 0$, $0 < \delta < 1$, and $R \geq \frac{16\sqrt{\mathrm{vol}(G)}}{\delta}$. Let $t \geq 1$, and $\mathbf{p}_a^t$ be the probability distribution of the endpoints of a t-step random walk starting from a. There exists an algorithm, denoted by **ℓ_2^2-norm tester**(G, a, σ, R), that outputs accept if $\|D^{-\frac{1}{2}}\mathbf{p}_a^t\|_2^2 \leq \frac{\sigma}{4}$, and outputs reject if $\|D^{-\frac{1}{2}}\mathbf{p}_a^t\|_2^2 > \sigma$, with probability at least $1-\delta$. The running time of the tester is $O(R\cdot t)$.

Proof. The description of the algorithm **ℓ_2^2-norm tester**(G, a, σ, r) is simple:

1. Run $2R$ random walks of length t starting from a.
2. Let $X_{a,r}^i$ be a random variable which is $\frac{1}{\sqrt{\deg(i)}}$ if the r^{th} random walk starting from a, ends at vertex i, and 0 otherwise.
3. Let Z be a random variable given by $Z = \frac{1}{R^2}\sum_{i\in V_G}(\sum_{r_1=1}^{R}X_{a,r_1}^i)(\sum_{r_2=R+1}^{2R}X_{a,r_2}^i)$.
4. Reject if and only if $\|D^{-\frac{1}{2}}\mathbf{p}_a^t\|_2^2 > \frac{\sigma}{2}$.

   By equation A.1, and inequality A.2, in the proof of Lemma 2.19, we have $\mathbb{E}[Z] = \|D^{-\frac{1}{2}}\mathbf{p}_a^t\|_2^2$, and

$$\begin{aligned}\mathrm{Var}(Z) &\leq \frac{1}{R^2}\|D^{-\frac{1}{2}}\mathbf{p}_a^t\|_2\cdot\|D^{-\frac{1}{2}}\mathbf{p}_a^t\|_2 + \frac{1}{R}\|D^{-\frac{1}{2}}\mathbf{p}_a^t\|_2^2\cdot\|D^{-\frac{1}{2}}\mathbf{p}_a^t\|_2 + \frac{1}{R}\|D^{-\frac{1}{2}}\mathbf{p}_a^t\|_2\cdot\|D^{-\frac{1}{2}}\mathbf{p}_a^t\|_2^2\\ &\leq \frac{1}{R^2}\|D^{-\frac{1}{2}}\mathbf{p}_a^t\|_2^2 + \frac{2}{R}\|D^{-\frac{1}{2}}\mathbf{p}_a^t\|_2^3\\ &= \frac{1}{R^2}\mathbb{E}[Z] + \frac{2}{R}\mathbb{E}[Z]^{\frac{3}{2}}.\end{aligned}$$

Then by Chebyshev's inequality, we get,

$$\Pr\left[|Z-\mathbb{E}[Z]| > \frac{\mathbb{E}[Z]}{2}\right] < \frac{\mathrm{Var}[Z]}{(\frac{\mathbb{E}[Z]}{2})^2} \le \frac{\frac{4}{R^2}\mathbb{E}[Z] + \frac{8}{R}\mathbb{E}[Z]^{\frac{3}{2}}}{\mathbb{E}[Z]^2} = \frac{4}{R^2\cdot\mathbb{E}[Z]} + \frac{8}{R\cdot\mathbb{E}[Z]^{\frac{1}{2}}}.$$

Now Observe that $\mathbb{E}[Z] = \sum_{i=1}^{n}\frac{(\mathbf{p}_a^t(i))^2}{\deg(i)}$, is a convex funtion which is minimized when for all $1 \le i \ne j \le n$, $\frac{\mathbf{p}_a^t(i)}{\deg(i)} = \frac{\mathbf{p}_a^t(j)}{\deg(i)} = \frac{1}{\mathrm{vol}(V_G)}$. Thus we have

$$\mathbb{E}[Z] \ge \sum_{i=1}^{n}\frac{\left(\mathbf{p}_a^t(i)\right)^2}{\deg(i)} \ge \sum_{i=1}^{n}\frac{\left(\frac{\deg(i)}{\mathrm{vol}(V_G)}\right)^2}{\deg(i)} = \frac{1}{\mathrm{vol}(V_G)}.$$

Hence, we get,

$$\Pr\left[|Z-\mathbb{E}[Z]| > \frac{\mathbb{E}[Z]}{2}\right] \le \frac{4\cdot\mathrm{vol}(V_G)}{R^2} + \frac{8\cdot\mathrm{vol}(V_G)^{\frac{1}{2}}}{R} \le \delta.$$

The last inequality holds since $R \ge \frac{16\sqrt{\mathrm{vol}(G)}}{\delta}$.

Thus if $\|D^{-\frac{1}{2}}\mathbf{p}_a^t\|_2^2 \le \frac{\sigma}{4}$, then $\mathbb{E}[Z] \le \frac{\sigma}{4}$, and hence, with probability at least $1-\delta$, we have $Z \le \frac{\sigma}{4} + \frac{\sigma}{8} < \frac{\sigma}{2}$. And if $\|D^{-\frac{1}{2}}\mathbf{p}_a^t\|_2^2 \ge \sigma$, then with probability at least $1-\delta$, we have $Z \ge \frac{\mathbb{E}[Z]}{2} \ge \frac{\sigma}{2}$. Therefore, the tester outputs the correct anwer with probability at least $1-\delta$.

□

Bibliography

[Abb18] Emmanuel Abbe. Community detection and stochastic block models. *Foundations and Trends in Communications and Information Theory*, 14(1-2):1–162, 2018.

[ABM16] Kevin Aydin, MohammadHossein Bateni, and Vahab Mirrokni. Distributed balanced partitioning via linear embedding. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, pages 387–396. ACM, 2016.

[ACL08] Reid Andersen, Fan R. K. Chung, and Kevin J. Lang. Local partitioning for directed graphs using pagerank. *Internet Mathematics*, 5(1):3–22, 2008.

[AGPT16] Reid Andersen, Shayan Oveis Gharan, Yuval Peres, and Luca Trevisan. Almost optimal local graph clustering using evolving sets. *J. ACM*, 63(2):15:1–15:31, 2016.

[ALM13a] Zeyuan Allen Zhu, Silvio Lattanzi, and Vahab S. Mirrokni. A local algorithm for finding well-connected clusters. In *ICML*, pages 396–404, 2013.

[ALM13b] Zeyuan Allen Zhu, Silvio Lattanzi, and Vahab S. Mirrokni. A local algorithm for finding well-connected clusters. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, pages 396–404, 2013.

[ARVX12] Noga Alon, Ronitt Rubinfeld, Shai Vardi, and Ning Xie. Space-efficient local computation algorithms. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2012, Kyoto, Japan, January 17-19, 2012*, pages 1132–1139, 2012.

[AS12] Pranjal Awasthi and Or Sheffet. Improved spectral-norm bounds for clustering. In Anupam Gupta, Klaus Jansen, José D. P. Rolim, and Rocco A. Servedio, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques - 15th International Workshop, APPROX 2012, and 16th International Workshop, RANDOM 2012, Cambridge, MA, USA, August 15-17, 2012. Proceedings*, volume 7408 of *Lecture Notes in Computer Science*, pages 37–49. Springer, 2012.

[BJ06] Francis R. Bach and Michael I. Jordan. Learning spectral clustering, with application to speech separation. *J. Mach. Learn. Res.*, 7:1963–2001, 2006.

[BMD$^+$15] Dino Bellugi, David G Milledge, William E Dietrich, Jim A McKean, J Taylor Perron, Erik B Sudderth, and Brian Kazian. A spectral clustering search algorithm for predicting shallow landslide size and location. *Journal of Geophysical Research: Earth Surface*, 120(2):300–324, 2015.

[Bol80] Béla Bollobás. A probabilistic proof of an asymptotic formula for the number of labelled regular graphs. *Eur. J. Comb.*, 1(4):311–316, 1980.

[Bol88] Béla Bollobás. The isoperimetric number of random regular graphs. *Eur. J. Comb.*, 9(3):241–244, 1988.

[CGR$^+$14] Artur Czumaj, Oded Goldreich, Dana Ron, C. Seshadhri, Asaf Shapira, and Christian Sohler. Finding cycles and trees in sublinear time. *Random Struct. Algorithms*, 45(2):139–184, 2014.

[CKCLL$^+$13] Tsz Chiu Kwok, Lap Chi Lau, Yin Tat Lee, Shayan oveis gharan, and Luca Trevisan. Improved cheeger's inequality: Analysis of spectral partitioning algorithms through higher order spectral gap. *Proceedings of the Annual ACM Symposium on Theory of Computing*, 01 2013.

[CKK$^+$18] Ashish Chiplunkar, Michael Kapralov, Sanjeev Khanna, Aida Mousavifar, and Yuval Peres. Testing graph clusterability: Algorithms and lower bounds. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 497–508. IEEE, 2018.

[CPS15] Artur Czumaj, Pan Peng, and Christian Sohler. Testing cluster structure of graphs. In *STOC*, pages 723–732, 2015.

[CS10a] Artur Czumaj and Christian Sohler. Testing expansion in bounded-degree graphs. *Combinatorics, Probability & Computing*, 19(5-6):693–709, 2010.

[CS10b] Artur Czumaj and Christian Sohler. Testing expansion in bounded-degree graphs. *Combinatorics, Probability & Computing*, 19(5-6):693–709, 2010.

[Das16] Sanjoy Dasgupta. A cost function for similarity-based hierarchical clustering. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 118–127, 2016.

[DK] Chandler Davis and William Morton Kahan.

[DK70] Chandler Davis and William Morton Kahan. The rotation of eigenvectors by a perturbation. iii. *SIAM Journal on Numerical Analysis*, 7(1):1–46, 1970.

[DMS15] Amir Dembo, Andrea Montanari, and Subhabrata Sen. Extremal cuts of sparse random graphs. *CoRR*, abs/1503.03923, 2015.

[ELR18] Talya Eden, Reut Levi, and Dana Ron. Testing bounded arboricity. In *SODA*, pages 2081–2092, 2018.

[ELRS17] Talya Eden, Amit Levi, Dana Ron, and C. Seshadhri. Approximately counting triangles in sublinear time. *SIAM J. Comput.*, 46(5):1603–1646, 2017.

[ER18] Talya Eden and Will Rosenbaum. On sampling edges almost uniformly. In *SOSA*, pages 7:1–7:9, 2018.

[ERS17a] Talya Eden, Dana Ron, and C. Seshadhri. On approximating the number of \$k\$-cliques in sublinear time. *CoRR*, abs/1707.04858, 2017.

[ERS17b] Talya Eden, Dana Ron, and C. Seshadhri. Sublinear time estimation of degree distribution moments: The degeneracy connection. In *ICALP*, pages 7:1–7:13, 2017.

[GKL$^+$21] Grzegorz Gluch, Michael Kapralov, Silvio Lattanzi, Aida Mousavifar, and Christian Sohler. Spectral clustering oracles in sublinear time. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1598–1617. SIAM, 2021.

[GLMY11] Ullas Gargi, Wenjun Lu, Vahab S Mirrokni, and Sangho Yoon. Large-scale community detection on youtube for topic discovery and exploration. In *ICWSM*, 2011.

[GR99] Oded Goldreich and Dana Ron. A sublinear bipartiteness tester for bounded degree graphs. *Combinatorica*, 19(3):335–373, 1999.

[GR00] Oded Goldreich and Dana Ron. On testing expansion in bounded-degree graphs. *Electronic Colloquium on Computational Complexity (ECCC)*, 7(20), 2000.

[GR02] Oded Goldreich and Dana Ron. Property testing in bounded degree graphs. *Algorithmica*, 32(2):302–343, 2002.

[GR10] Mira Gonen and Dana Ron. On the benefits of adaptivity in property testing of dense graphs. *Algorithmica*, 58(4):811–830, 2010.

[GR11a] Oded Goldreich and Dana Ron. Algorithmic aspects of property testing in the dense graphs model. *SIAM J. Comput.*, 40(2):376–445, 2011.

[GR11b] Oded Goldreich and Dana Ron. On testing expansion in bounded-degree graphs. In Oded Goldreich, editor, *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation - In Collaboration with Lidor Avigad, Mihir Bellare, Zvika Brakerski, Shafi Goldwasser, Shai Halevi, Tali Kaufman, Leonid Levin, Noam Nisan, Dana Ron, Madhu Sudan, Luca Trevisan, Salil Vadhan, Avi Wigderson, David Zuckerman*, volume 6650 of *Lecture Notes in Computer Science*, pages 68–75. Springer, 2011.

[GS13] Lior Gishboliner and Asaf Shapira. Deterministic vs non-deterministic graph property testing. *Electronic Colloquium on Computational Complexity (ECCC)*, 20:59, 2013.

[GT14a] Shayan Oveis Gharan and Luca Trevisan. Partitioning into expanders. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 1256–1266. SIAM, 2014.

[GT14b] Shayan Oveis Gharan and Luca Trevisan. Partitioning into expanders. In *SODA*, pages 1256–1266, 2014.

[HJ90] Roger A. Horn and Charles R. Johnson. *Matrix analysis*. Cambridge University Press, 1990.

[HNH13] Stefan Heule, Marc Nunkesser, and Alexander Hall. Hyperloglog in practice: algorithmic engineering of a state of the art cardinality estimation algorithm. In *Proceedings of the 16th International Conference on Extending Database Technology*, pages 683–692. ACM, 2013.

[Jai10] Anil K Jain. Data clustering: 50 years beyond k-means. *Pattern recognition letters*, 31(8):651–666, 2010.

[KKSV17] Michael Kapralov, Sanjeev Khanna, Madhu Sudan, and Ameya Velingker. $1+\Omega(1)$-approximation to MAX-CUT requires linear space. In *SODA*, pages 1703–1722, 2017.

[KPS13] Satyen Kale, Yuval Peres, and C. Seshadhri. Noise tolerance of expanders and sublinear expansion reconstruction. *SIAM J. Comput.*, 42(1):305–323, 2013.

[KS08] Satyen Kale and C. Seshadhri. An expansion tester for bounded degree graphs. In Luca Aceto, Ivan Damgård, Leslie Ann Goldberg, Magnús M. Halldórsson, Anna Ingólfsdóttir, and Igor Walukiewicz, editors, *Automata, Languages and Programming, 35th International Colloquium, ICALP 2008, Reykjavik, Iceland, July 7-11, 2008, Proceedings, Part I: Tack A: Algorithms, Automata, Complexity, and Games*, volume 5125 of *Lecture Notes in Computer Science*, pages 527–538. Springer, 2008.

[KS11] Satyen Kale and C. Seshadhri. An expansion tester for bounded degree graphs. *SIAM J. Comput.*, 40(3):709–720, 2011.

[KVV04a] Ravi Kannan, Santosh Vempala, and Adrian Vetta. On clusterings: Good, bad and spectral. *J. ACM*, 51(3):497–515, 2004.

[KVV04b] Ravi Kannan, Santosh S. Vempala, and Adrian Vetta. On clusterings: Good, bad and spectral. *J. ACM*, 51(3):497–515, 2004.

[LGT14] James R Lee, Shayan Oveis Gharan, and Luca Trevisan. Multiway spectral partitioning and higher-order cheeger inequalities. *Journal of the ACM (JACM)*, 61(6):37, 2014.

[LRTV12] Anand Louis, Prasad Raghavendra, Prasad Tetali, and Santosh Vempala. Many sparse cuts via higher eigenvalues. In *Proceedings of the Forty-fourth Annual ACM Symposium on Theory of Computing*, STOC '12, pages 1131–1140, New York, NY, USA, 2012. ACM.

[LV13] László Lovász and Katalin Vesztergombi. Non-deterministic graph property testing. *Combinatorics, Probability & Computing*, 22(5):749–762, 2013.

[MMV12] Konstantin Makarychev, Yury Makarychev, and Aravindan Vijayaraghavan. Approximation algorithms for semi-random graph partitioning problems. *arXiv preprint arXiv:1205.2234*, 2012.

[NJW02] Andrew Y Ng, Michael I Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in neural information processing systems*, pages 849–856, 2002.

[NS10] Asaf Nachmias and Asaf Shapira. Testing the expansion of a graph. *Inf. Comput.*, 208(4):309–314, 2010.

[NS13] Ilan Newman and Christian Sohler. Every property of hyperfinite graphs is testable. *SIAM Journal on Computing*, 42(3):1095–1112, 2013.

[OA14] Lorenzo Orecchia and Zeyuan Allen Zhu. Flow-based algorithms for local graph clustering. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5-7, 2014*, pages 1267–1286, 2014.

[O'D14] Ryan O'Donnell. *Analysis of Boolean Functions*. Cambridge University Press, 2014.

[PCS06] Alberto Paccanaro, James A Casbon, and Mansoor AS Saqi. Spectral clustering of protein sequences. *Nucleic acids research*, 34(5):1571–1580, 2006.

[Pen20] Pan Peng. Robust clustering oracle and local reconstructor of cluster structure of graphs. In Shuchi Chawla, editor, *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, pages 2953–2972. SIAM, 2020.

[PS18] Pan Peng and Christian Sohler. Estimating graph parameters from random order streams. In *SODA*, pages 2449–2466, 2018.

[RAK$^+$16] David Rolnick, Kevin Aydin, Shahab Kamali, Vahab Mirrokni, and Amir Najmi. Geocuts: Geographic clustering using travel statistics. *arXiv preprint arXiv:1611.03780*, 2016.

[RTVX11] Ronitt Rubinfeld, Gil Tamir, Shai Vardi, and Ning Xie. Fast local computation algorithms. In *Proceedings of Innovations in Computer Science - ICS 2010, Tsinghua University, Beijing, China, January 7-9, 2011*, pages 223–238, 2011.

[Ses15] C Seshadhri. A simpler sublinear algorithm for approximating the triangle count. *arXiv preprint arXiv:1505.01927*, 2015.

[Sin16] Ali Kemal Sinop. How to round subspaces: A new spectral clustering algorithm. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 1832–1847, 2016.

[SM00] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *Departmental Papers (CIS)*, page 107, 2000.

[SSS93] Jeanette P. Schmidt, Alan Siegel, and Aravind Srinivasan. Chernoff-hoeffding bounds for applications with limited independence. In *Proceedings of the Fourth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '93, pages 331–340, Philadelphia, PA, USA, 1993. Society for Industrial and Applied Mathematics.

[ST14] Daniel A. Spielman and Shang-Hua Teng. Nearly linear time algorithms for preconditioning and solving symmetric, diagonally dominant linear systems. *SIAM J. Matrix Analysis Applications*, 35(3):835–885, 2014.

[Tod11] Alexis Akira Toda. Operator reverse monotonicity of the inverse. *The American Mathematical Monthly*, 118(1):82–83, 2011.

[Tro12] Joel A Tropp. User-friendly tail bounds for sums of random matrices. *Foundations of computational mathematics*, 12(4):389–434, 2012.

[VL07] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

[Yos10] Yuichi Yoshida. Lower bounds on query complexity for testing bounded-degree csps. *CoRR*, abs/1007.3292, 2010.

[Yos11] Yuichi Yoshida. Lower bounds on query complexity for testing bounded-degree csps. In *CCC*, pages 34–44, 2011.

[ZLM13] Zeyuan Allen Zhu, Silvio Lattanzi, and Vahab Mirrokni. A local algorithm for finding well-connected clusters. In *International Conference on Machine Learning*, pages 396–404. PMLR, 2013.

Aida Mousavifar

✉ aidasadat.mousavifar@epfl.ch
in https://www.linkedin.com/in/aida-mousavifar
Google Scholar

EPFL, 1015 Lausanne, Switzerland

27 Years old

Education

École Polytechnique Fédérale de Lausanne (EPFL), Switzerland [2016-Present]

Ph.D. in Computer Science
Theory of Computation Laboratory under supervision of Prof. Michael Kapralov
Recipient of Google PhD fellowship in Algorithms, Optimization and Markets
Thesis title: Designing sublinear algorithms for big data analysis
Research interests: graph algorithms, optimization, clustering, streaming, sketching, data analysis, machine learning

University of Tehran, Iran [2012-2016]

B.Sc. in Computer Engineering (Software)
Ranked 1st among all students GPA: 18.80/20 (equivalent to 4.0/4.0)
Thesis title: Designing approximation algorithms for optimization problems on graphs

Experience

Doctoral researcher under supervision of Prof. Michael Kapralov at EPFL [2016- Present]

- Designed a fast algorithm for understanding the **cluster structure of big graphs** in sublinear time. Joint work with Microsoft Research Redmond as part of the Swiss Joint Research Centre initiative with MSR. (FOCS'18)
- Designed a low-memory algorithm for streaming submodular maximization, that is widely used in **machine learning** and data mining. (**spotlight talk** in ICML'18)
- Designed a fast and low-memory **graph sketching** algorithm for spectral sparsification in **dynamic streams**. (SODA'20)
- Designed a local computation algorithm for graph clustering in sublinear time. (SODA'21)

Research / Software Engineer intern at Google Research under supervision of Dr. Silvio Lattanzi at Google Zurich [Summer 2020]

- Designed a sublinear **hierarchical clustering** algorithm.
- Developed a library (in C++) that supports clustering algorithms on dynamic graphs. (achieved a 40% gain in precision of cluster recovery in experiments on real datasets)

Research intern under supervision of Prof. Nisheeth Vishnoi at EPFL [Summer 2015]

- Developed a **game theoretical network model** inspired by social information networks. (WINE'17)

Honors and Awards

Recipient of Postdoc Mobility fellowship for pursuing postdoctoral at Stanford [2021]

Recipient of ETH-FDS postdoctoral fellow position from ETH Zurich [2021]

Recipient of Max Planck postdoctoral fellowship, Germany [2021]

Recipient of Google PhD fellowship in Algorithms, Optimization and Markets [2019]

Best research presentation award (1st Place) at the Computer Science Research Day at EPFL, (talk) [2019]

Ranked 5th in the Iranian National University Students **Olympiad in Computer Engineering**,
(Iran's premier computer engineering contest) [2015]

Recipient of Faculty of Engineering Award as the **1st ranked student** for four consecutive years in University of Tehran [2012-2016]

Exempted from MSc university entrance exam in Iran as an exceptionally talented student [2016]

Accepted in the second round (first 50 participants) in Iranian **National Olympiad in Informatics** [2011]

Semi-finalist in Iranian **National Olympiad in Mathematics** [2009, 2010, 2011]

Recipient of Distinguished Service Award as a president of committee of graduate student association at EPFL (EPIC) [2019]

Publications

1. G. Gluch, M. Kapralov, S. Lattanzi, A. Mousavifar, C. Sohler, "*Spectral Clustering Oracles in Sublinear Time*". In ACM-SIAM Symposium on Discrete Algorithms (SODA), 2021, Authors ordered alphabetically

2. M. Kapralov, A. Mousavifar, C. Musco, C. Musco, N. Nouri, A. Sidford, J. Tardos, "*Fast and Space Efficient Spectral Sparsification in Dynamic Streams*". In ACM-SIAM Symposium on Discrete Algorithms (SODA), 2020, Authors ordered alphabetically

3. A. Chiplunkar, M. Kapralov, S. Khanna, A. Mousavifar, Y. Peres, *"Testing Graph Clusterability: Algorithms and Lower Bounds*". In 59th Annual IEEE Symposium on Foundation of Computer (FOCS), 2018, Authors ordered alphabetically,

4. A. Norouzi Fard, J. Tarnawski, S. Mitrovic, A. Zandieh, A.Mousavifar, O. Svensson, "*Beyond 1/2-Approximation for Submodular Maximization on Massive Data Streams*". In 35th International Conference on Machine Learning (ICML), 2018, **spotlight talk**

5. Elisa Celis, A.Mousavifar, "*A Model for Social Network Formation: Efficiency, Stability and Dynamics*". Accepted for the poster session and short talk in 13th Conference on Web and Internet Economics (WINE), 2017

6. M. Kapralov, A. Mousavifar, C. Musco, C. Musco, N. Nouri, "*Faster Spectral Sparsification in Dynamic Streams*". 2019, Authors ordered alphabetically, (arXiv)

7. M. Kapralov, A. Kumar, S. Lattanzi, A. Mousavifar, "*Spectral Hierarchical Clustering Oracles in Sublinear Time*". 2021, Authors ordered alphabetically (manuscript)

8. M. Kapralov, A. Kumar, S. Lattanzi, A. Mousavifar, "*Spectral Hierarchical Clustering Oracles in Sublinear Time*". 2021, Authors ordered alphabetically (manuscript)

Teaching Experience

Teaching assistant in

▪ Advanced Algorithms	EPFL [Spring 2018, 2019]
▪ Algorithms	EPFL [Fall 2017, 2018, 2019, 2020]
▪ Probabilities and Statistics	EPFL [Fall 2016]
▪ Graph Theory (head TA)	University of Tehran [Spring 2016]
▪ Design and Analysis of Algorithms (head TA)	University of Tehran [Fall 2014, 2015]
▪ Advanced Programming	University of Tehran [Spring 2015]
▪ Database and Systems	University of Tehran [Spring 2015]
▪ Discrete Mathematics	University of Tehran [Spring 2014, 2015, 2016]

Teaching contestants of ACM-ICPC programming contest ACM Student Chapter, University of Tehran [2014]

Teaching contestants of Informatics and Mathematics Olympiad Farzanegan High School, [2013,2014]

Academic Services

- Reviewer for conferences: SODA'19, SODA'20, NeurIPS'20, ICML'20, PODS'19, STACS'19, ESA'18, FSTTCS'18
- President of committee of graduate student association at EPFL (EPIC) EPFL, [2019]
- Supervised a Bachelor semester project EPFL, [2018]
- Organized mock programming contests for contestants of ACM-ICPC Regional Contest University of Tehran, [2014, 2015]

Competitive programming

Bronze medal in the ACM ICPC Southwestern Europe Regional Contest, Porto [2016]

Silver medal in the 17th ACM ICPC Asia Regional Contest, Tehran [2015]

Silver medal in the 16th ACM ICPC Asia Regional Contest, Tehran [2014]

References

Prof. Michael Kapralov (michael.kapralov@epfl.ch), faculty at EPFL

- Association: Co-author and Ph.D. supervisor

Dr. Silvio Lattanzi (silviol@google.com), researcher at Google

- Association: Co-author, Google fellowship supervisor, and internship supervisor

Prof. Sanjeev Khanna (sanjeev@cis.upenn.edu), faculty at University of Pennsylvania

- Association: Co-author

Prof. Christian Sohler (sohler@uni-koeln.de), faculty at University of Cologne

- Association: Co-author

Technical Skills

Programming:	C / C++, Java, Python Verilog Hardware Description Language, Socket Programming,
Web development:	JavaScript, HTML, CSS, Ruby on Rails, jQuery, Doctrine, Codingither
Hardware Simulator:	ModelSim, Quartus II, PCB Design, Proteus, AVR, Microcontrollers

Selected Projects

Analysis of CERN datasets to recreate the process of discovering the Higgs particle by applying different **machine learning**, and pattern recognition techniques, such as classification, data reconstruction, feature processing and modelling

- ✓ using Python

Developed a movie **recommendation system** by applying different machine learning techniques such as neighbourhood models, matrix factorization, alternating least-squares, and blending

- ✓ using Python

Implemented a project organizer platform similar to Trello with graphical user interface

- ✓ using C++, C sockets, multi-threading

Developed a social network with the feature of community detection and friend suggestion, using B+ Tree and Treap data structures

- ✓ using Java

Provided a service to create homepages with multiple configurations [University of Tehran Summer of Code]

- ✓ using MVC Pattern, Ruby on Rails, Doctrine, Codingither, HTML, CSS, jQuery, and MySQL

Simulated different tasks of operating systems e.g. system calls, address translation, memory management, CPU scheduler and file system

- ✓ using Nach OS and Java

Linux kernel optimization, compiled Linux kernel code and implemented system calls, semaphores, and fair share scheduler

- ✓ using C and Qemu virtual machine

Simulated Ethernet switch, multi-cast IP and different routing algorithms

- ✓ using C socket programming

Implemented a pipeline MIPS processor (single-cycle and multi-cycle)

- ✓ using Verilog, Quartus

Languages

English (Fluent), French (Basic proficiency: A2 level), German (Basic proficiency: A2 level), Persian (Native)

Ce document a été imprimé au Centre d’impression EPFL,
imprimerie climatiquement neutre, certifiée myClimate.