SPECTRAL APPROACH TO MODERN ALGORITHM DESIGN

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Akash Kumar

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

May 2020

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF DISSERTATION APPROVAL

Dr. Saugata Basu, Chair

Department of Mathematics, Purdue University

Dr. Jeremiah Blocki

Department of Computer Science, Purdue University

Dr. Simina Branzei

Department of Computer Science, Purdue University

Dr. Petros Drineas

Department of Computer Science, Purdue University

Dr. Elena Grigorescu

Department of Computer Science, Purdue University

Dr. Hemanta Maji

Department of Computer Science, Purdue University

Approved by:

Dr. Clifton W. Bingham

Head of the Departmental Graduate Program

To my Parents and to my little nephew, Atharv

ACKNOWLEDGMENTS

First and foremost, I wish to express my sincere gratitude to my amazing advisor Saugata Basu. His drive for rigor and clarity have helped me mature as a researcher. I hope someday I have half as much clarity in my thoughts. Besides this, I am extremely thankful to him for giving me much needed intellectual freedom. For all this and quite a bit left out: thanks a ton Saugata!

I joined the PhD program with some trepidation especially since I came back to academia from the software industry. I am thankful to Elena Grigorescu for mentoring me in those initial years. To Saugata, Elena, Petros, Simina, Jeremiah and Maji – thanks all for taking out the time to serve on my committee especially when the COVID situation hit the world. A big thanks to Petros for helping me out countless times with his amazing advice and mentorship! No PhD journey can be fulfilling without fun collaborators. To this end, I am grateful to all of my collaborators over the course of my PhD: Clement Canonne, Siyao Guo, Elena Grigorescu, Karl Wimmer, Anand Louis, Madhur Tulsiani, Mrinalkanti Ghosh, C. Seshadhri, and Andrew Stolman. Thank you, I learnt a lot from each one of you.

I owe a huge intellectual debt to Madhur, Seshadhri and Anand in particular. Fortunately, I interned with them all! I thank Madhur for being like a second advisor! Thanks for allowing me to visit TTIC so frequently and for being willing to e ngage with me in stimulating conversations over the course of past several years. You debugged my understanding of almost everything so many times that it indirectly influenced every single fiber of the way I approach research. Thank you for being the center of my universe on all things academic! I visited Seshadhri at the Sandia National Labs in Summer 2014 and I was inspired by his fearlessness and his enthusiasm to dive headfirst into some really challenging open problems. I also admire his seamless ability to go back and forth across various levels of the abstraction hieararchy when attacking any problem. In the modern world, where "meme culture" has taken on strong roots, I think it will be fun to add one for beginning PhD students: be like Seshadhri! (I hope Sesh would not mind.) Anand and I go back a bit. We met each other when I was enrolled in a Masters Program at Georgia Tech. I visited Anand professionally in Summer 2017 at IISc. I enjoyed our conversations and learnt a fair bit from our brief but intense collaboration together with Madhur.

Among my collaborators, the person I discussed research the most with is without a doubt Andrew Stolman. Thank you Andrew! Closer to home at Purdue, I came in touch with some really remarkable people. Thanks in particular to the theory group here. Which includes Abhiram, GV, Abram, Samson, Negin, Peng, Mingyuan Wang, Minshen Zhu, Young-San Lin, Tamalika, Alex and Hai. A special thanks to my close friend Vikram Ravindra: I would not have survived this journey without numerous jokes you made (sometimes on me) and the deep insights you shared. Thank you in particular! Thanks again to Abhiram and Asish for their help during a particularly difficult time in my personal life. I also thank other friends I made at Purdue: Rohit, Asish, Kaushal, Mayank, Viplove and Aakash. Among my friends and mentors at Georgia Tech, I wish to acknowledge Dana Randall, Prasad Tetali and the late Robin Thomas for being awesome teachers. And Anand Louis, Arindam Khan, Nishad Kothari and Abhinav Shantanam for being amazing friends.

I am also immensely grateful to an entire generation of researchers/engineers before me whose dedicated efforts led us to great youtube channels like the one by Simons Institute, IAS etc. I am also thankful to Ryan O'Donnell, Ankur Moitra and Tim Roughgarden for posting their beautiful lectures online. A special thanks to Ryan for clarifying quite a few questions about SOS over the email. Also a shout out to Anupam Gupta for letting me use his random walk animation for part of my defense.

As a PhD student I have been in the rather fortunate position of being able to visit two other schools frequently: UC Santa Cruz and TTIC. Over the years I have made quite a few friends there and it would be amiss of me to not acknowledge them. Especially when both of these groups treat me as their own and joke that "I am a part time student at their university!" To this end, I thank Mrinalkanti Ghosh, Shashank Srivastava and Rachit Nimavat. Thanks for all the fun. Thanks especially to Mrinal and Shashank for letting me crash at their place on way more occassions than I can count. Thanks to Mrinal, Aaron Potechin and Fernando Granha for numerous discussions on Sum-of-Squares hierarchy among other things. I thank my friend Params Raman at UC Santa Cruz. Params and I shared identical career paths right from our Masters days. He also started off as a PhD student at Purdue and then moved over to UC Santa Cruz alongwith his advisor. His suggestions have always soothed me in my difficult times. I also would like to thank Shweta Jain at UC Santa Cruz for numerous discussions and especially for critiquing my defense slides.

Finally, I would like to thank my very dear friend Nishad Kothari for sharing with me his PhD journey and helping me prepare for what to expect. You know how much I value discussions with you. Thanks a lot for that and I hope to count on that in the future as well. On a somewhat unusual note, I am also immensely thankful to Martin Gardner for writing "The Colossal book of mathematics" which had a huge impact on me in terms of developing my mathematical appetite.

Besides this, I also wish to sincerely thank my friends: Abhinav Rai, Abhinav Shantanam, Ambica Pundir, Rohit Moghe, Amar Bajpai and Suraj and Saurabh Pratap. It goes without saying, I am immensely thankful to Ayush Kumar and Priyanjali Vashisth. Among so so many other fun things, thanks a lot to you for being parents to my little nephew, Atharv. To you, dear Atharv and to my parents, I dedicate this thesis.

TABLE OF CONTENTS

Page

LIST OF TABLES

ABSTRACT

Kumar, Akash PhD, Purdue University, May 2020. Spectral Approach to Modern Algorithm Design. Major Professor: Saugata Basu.

Spectral Methods have had huge influence of modern algorithm design. For algorithmic problems on graphs, this is done by using a deep connection between random walks and the powers of various natural matrices associated with the graph. The major contribution of this thesis initiates attempts to recover algorithmic results in Graph Minor Theory via spectral methods.

We make progress towards this goal by exploring these questions in the Property Testing Model for bounded degree graphs. Our main contributions are

- The first result gives an almost query optimal one-sided tester for the property of H-minor-freeness. Benjamini-Schramm-Shapira (STOC 2008) conjectured that for fixed H, this can be done in time $\tilde{O}(\sqrt{n})$. Our algorithm solves this in time $n^{1/2+o(1)}$ which nearly resolves this upto $n^{o(1)}$ factors.
- BSS also conjectured that in the two-sided model, H-minor-freeness can be tested in time $poly(1/\varepsilon)$. We resolve this conjecture in the affirmative.
- Lastly, in a previous work on the two-sided-question above, Hassidim-Kelner-Nguyen-Onak (FOCS 2009) introduced a tool they call *partition oracle*. They conjectured that partition oracles could be implemented in time $poly(1/\varepsilon)$ and gave an implementation which took $\exp(poly(1/\varepsilon))$ time. In this work, we resolve this conjecture and produce such an oracle.

Additionally, this work also presents an algorithm which can recover a planted 3-coloring in a graph with some random like properties and suggests some future research directions alongside.

1 THE POWER OF SPECTRAL APPROACHES IN ALGORITHM DESIGN

1.1 Introduction

Graphs have been a rich source of problems in discrete mathematics and algorithms for several decades. Spectral approaches have been the mainstay of modern graph algorithms for quite sometime. For details, see [1, 2] and the references therein. One of the most intriguing avenues where spectral methods find use in graph algorithms concern graph partitioning problems [1, 3–5]. This thesis explores spectral methods on two fronts: First, it makes an effort to improve the quality of partition given by (spectral graph theory) toolkit for some special family of graphs; and second it also adds more ammo to this toolkit for solving novel graph theoretic problems.

Using spectral techniques, jointly with Seshadhri and Stolman, we designed the first (almost query optimal) algorithm for finding minors in bounded degree graphs in sublinear time [6]. In another joint work with the same authors [7], we showed that in the so-called *two-sided-testing* model, minor-freeness in bounded degree graphs is *efficiently testable*. Earlier, it was known to be *testable* but not efficiently testable. These works are covered in Chapter 2 and Chapter 3 respectively. Chapter 4 builds up on these works and further refines our spectral partitioning tools to give the first polynomial time *partition oracles* for bounded degree minor-free graphs. *All three of these questions were open for at least a decade* and interestingly spectral techniques led to results that eluded the previous combinatorial approaches.

In another direction, spectral techniques also appear powerful for understanding planted problems. Notably, [8] used spectral methods to find an outrageously large planted clique (of size $\Omega(\sqrt{n})$) in a n-vertex graph. In another work, [9] used spectral methods to recover the color classes in a 3-colorable graph which comes from a nice random family of graphs with a planted 3 coloring. In [10], jointly with Anand

Louis and Madhur Tulsiani, we gave spectral algorithms to recover a 3-coloring when the graph has some *pseudorandom or random-like* properties and admits a planted 3-coloring (which is again *pseudorandom*). Recent works have shown that the guarantees given by spectral methods can also be achieved or improved upon by using the *sum-of-squares or SoS* approach [11] (which can also be thought of as "higher order spectral methods"). [1] This brings us to another topic we recently started working on: polynomial optimization using the SoS method. This is briefly discussed in §1.3

Unlike the case with planted problems, for worst case problems, it appears that the sum-of-squares approach is indeed more powerful than spectral methods. In an ongoing work, we are trying to understand (with my collaborators) the problem of optimizing polynomials over the sphere using the SoS approach. In particular, we are exploring a conjecture which essentially postulates that a huge class of randomly distributed homogeneous polynomials (with degree d in n variables) can be approximated to within a $n^{d/4-1/2}$ factor. Apriori, it was conceivable that random polynomials cannot be approximated to within a factor which is $o(n^{d/2-1})$ (see [13, 14]). §1.3.3 has more discussion on this and suggests some directions to explore this further.

1.2 Organization of the Thesis

One of the huge success stories of modern graph theory is the celebrated Graph Minor Theorem of Robertson and Seymour [15, 16]. Unfortunately, huge chunks of this literature have not been ported to the toolkit of the modern algorithm designer. Together with C. Seshadhri and Andrew Stolman, our research attempts to bridge this gap. In particular, under the assumption that the input graph G is sufficiently far from being H-minor-free, (one must delete εdn edges to get a H-minor-free graph) we give an algorithm that finds an H-minor in G in time $n^{1/2+o(1)}$. This almost resolves a decade old conjecture by Benjamini, Schramm and Shapira [17]. Our algorithm uses random walks which is a major point of departure from previous works (which used

[1] we note that in a curious twist, [12] brings us full circle by showing that for numerous planted problems the guarantees given by SoS can also be met by more nuanced spectral methods.

a combinatorial approach). One notes that it is rather easy using spectral arguments to show that there is $\widetilde{O}(\sqrt{n})$ random walk based algorithm for finding minors in expanders. Letting $|V(H)| = r$, fix some sufficiently small $\delta > 0$ and set a parameter $\ell = n^{\delta}$ (the length of the walk). Let us make a definition: call a vertex $s \in V(G)$ *leaky* if random walks of length ℓ from s mix (and thus have a small norm – at most ℓ^{-10r^2}. Suppose the graph G has at least n/ℓ leaky vertices. Our key contribution is that a simple random walk based algorithm which we used in the expanding case (with some important modifications) can still find a K_r-minor (and thus, an H-minor). In the other case, when the number of leaky vertices is super small, at most n/ℓ, our analysis shows that G admits a *hyperfinite decomposition*. That is, G can then be partitioned into several tiny pieces many of which contain an H-minor. Moreover, our algorithm can find a superset of any piece by performing random walks in G. Putting all of this together, we are able to find minors with a number of queries tiny bit bigger than the bound conjectured by BSS. This is further explored in Chapter 2.

Chapter 3 considers a followup to this work [7], where we attack the *two-sided* version of this problem. This involves distinguishing graphs which are H-minor-free from those graphs which are ε-far from being H-minor-free as opposed to one-sided version above (which involves finding the H-minor). This problem was first studied by [17] who gave an algorithm runs in time triply exponential in $1/\varepsilon$ and returns the correct verdict with probability $2/3$. A series of improvements followed [18,19] got the runtime down to $1/\varepsilon^{O(\log(1/\varepsilon))}$. [7] uses spectral arguments and gets the running time down to $poly(1/\varepsilon)$. The key here is to estimate the number of leaky vertices (which can be done with very few queries). And if the number of leaky vertices is large, then the estimation process can straight up reject the graph. Otherwise, if this number is much smaller, we show that the graph again admits a hyperfinite decomposition. Moreover, just like earlier these pieces can be discovered by performing random walks.

Chapter 4 considers the problem of constructing a $poly(1/\varepsilon)$ time partition oracle for minor-free graphs. [18] introduced the notion of partition oracles for two-sided-testing H-minor-freeness. These oracles are local procedures which output the connected component a vertex lies in with no knowledge of the explicit global partition. This is challenging as one must maintain consistency across queries. The final construction of [18] is an intricate recursive procedure that requires $\exp O(d/\varepsilon)$ queries. This was improved by [19] who gave an algorithm that makes $\exp O(\log^2 d/\varepsilon)$ queries. Constructing a partition oracle which makes $poly(1/\varepsilon)$ queries has been a huge open question. Building up on the tools introduced in [6,7] we give a (random walk based) global algorithm which can be used to partition the graph. We show how this global procedure can be locally simulated by using the "diffusion-like" view of random walk on graphs. As of the writing of this thesis, this work (jointly with Seshadhri and Stolman) is currently in submission to FOCS 2020 and it gives the first construction of partition oracles for minor-free graphs which run provably in time $poly(1/\varepsilon)$ settling a question open for a little more than a decade.

Spectral methods are really versatile and can be used to attack a wide variety of algorithmic problems. Chapter 5 explores this versatility by using spectral methods for the graph coloring problem. This chapter reports the partial progress towards obtaining subexponential time algorithms for coloring a 3-colorable graph with $O(n^\delta)$ colors. [2] The starting point of this chapter is the seminal result of [9] which considers the planted problem of recovering a 3-coloring in a special family of random graphs. It turns out, that this random family has some spectrally nice properties which can be used to show that a simple spectral clustering heuristic successfully colors expanding graphs. Chapter 5 shows that the essentially the same heuristic with the subspace enumeration approach from [4,20] continues to recover "pseudorandom" 3-colorings in graphs which are "pseudorandom" and do not enjoy the spectral properties needed in the analysis of [9]. We make an important comment about this result.

To do this, let us introduce some terminology. We call a graph G partially 3-colorable if there is a set $S \subseteq V(G)$ with $|S| \geq (1-\alpha)n$ such that $G[S]$ admits a

[2] The exact statement of the conjecture appears in §1.3

pseudorandom proper 3-coloring. Our algorithm has the key additional property that it can correctly color $1-O(\alpha)$ fraction of the vertices and leave the remaining vertices uncolored. §1.3 explains why is this desirable in more detail.

1.3 Future Directions

The results described in the thesis lead to a number of interesting questions. We collected a few of those in §1.2. Below, I present a list of representative problems I would like to explore in the near future.

1.3.1 Future Directions: Problems related to minor freeness

Motivated by the power of spectral techniques for minor-freeness testing problems, we are interested in understanding to what extent can spectral arguments be used to recover algorithmic results which rely on the theory of graph minors. Below, I mention an assortment of some of these problems.

An alternate proof of graph minor theorem? For a bounded degree graph G, our work seems to suggest that the echoes of G being H-minor-free are visible in the random walk behavior from numerous nodes in G. If the random walks from more than a mere handful vertices mix rapidly over their support, G cannot be minor-free. Thus, one can attempt to characterize H-minor-freeness by looking at the rate of decay of random walk vectors.

Near-Linear Time algorithms for Balanced Separators Over the past several years, some of the deepest advances in Graph Minor Theory have been used to recover a balanced separator in H-minor-free graphs [21–23]. However, getting a $\widetilde{O}(n)$ time algorithm which returns a separator of size $\widetilde{O}(\sqrt{n})$ remains elusive. It would be interesting to see how good a separator can we recover using our random walk based decomposition algorithms.

Other problems concerning graph minors There are numerous other problems concerning minor-free graphs where I would like to see if spectral methods can say something interesting. One of the major open questions in this area is whether there exists an $o(n^2)$ algorithm which decides whether or not an input graph G is H-minor-free where $|V(H)| = O(1)$. The current best algorithm due to [24] runs in quadratic time and it again uses the Graph Minor Theory toolkit.

1.3.2 Future Directions: Spectral Algorithms for Partial Colorings

As described in §1.2, one of our long term goals is to show that we can color a 3-colorable graph in subexponential time (at least in the bi-criteria setting below). To this end, we have the following conjecture.

Conjecture for Subexponential time coloring algorithms for expanders Fix $\delta, \varepsilon > 0$. Then there exits an algorithm which when given an expanding graph G as input, runs in time $\exp(n^\delta) \cdot \left(\frac{1}{\varepsilon}\right)^{1/\delta}$ and returns a coloring χ such that

- No edges are violated under χ.
- At most εn vertices are left uncolored.

Approaching Partial 3-coloring It is also of interest to explore the partial 3-coloring problem that we considered in [10]. In particular, we would like to understand how small can we make $\alpha > 0$ such that there exists a polynomial time algorithm which, on input a $(1-\varepsilon)$ partially 3-colorable graph, returns a $(1 - O(\varepsilon))$ partial coloring with n^α colors? Some partial progress on these questions was made recently in [25].

1.3.3 Future Directions: Approximation algorithms using SDP hierarchies

Finally, we turn to the SDP sum-of-squares connection mentioned earlier in §1.1. As noted there, SDP's and spectral approaches seem to be equally powerful for planted problems. For worst case problems, SDP's do seem more powerful. Sum-of-Squares approach for optimization problems consists of solving a semidefinite program at the end of the day where the size of the SDP depends on the degree d of the SoS relaxation. To understand the power and limitations of degree d-SOS better, a classical problem often considered is polynomial optimization over the unit sphere [13]. Degree d polynomial optimization can be thought of as capturing *higher order spectral methods using SOS*. The algorithmic problem considered seeks to find $\max f(x) s.t. \|x\|_2 = 1$ where $f \in \mathbb{R}[X_1, X_2, \ldots, X_n]$ and $degree(f) = d$. [13] earlier showed that there exists a distribution μ supported on homogeneous degree d polynomials such that with high probability for $f \sim \mu$, $SoS(f)/OPT(f) \geq n^{d/4-1/2}$. It was shown in [14] that, *for any* polynomial f, we have $SOS(f)/OPT(f) \leq n^{d/2-1}$ and thus there is a quadratic gap between the best approximation a degree d SOS can achieve and the known lower bound on the quality of approximation. [26] introduced another distribution (which we denote by $\prod$) polynomials coming from which have some nicer properties. Namely, for large d and the number of variables n held fixed, [27] showed that the number of connected components of the zero set of f has many more connected components than what we typically see in a random polynomial from μ (the distribution considered in [13]). Naively, this leads to the conjecture that perhaps one could "hide" the maximum better from degree d SOS by using the random family $\prod$. We give a report of our ongoing work on this problem and have not included any chapter on this. We managed to show that with very high probability, a random polynomial $f \sim \prod$ satisfies that $max_{x \in S^{n-1}} f(x) \leq O(\sqrt{n})$. It remains to be seen what approximation factor can SOS achieve for polynomials coming from this family. Thus, we also ask

Approximating polynomials on the sphere Let $f \sim \prod$ where $\prod$ is the distribution supported over homogeneous degree d polynomials defined in [26]. We would like to understand what is the best approximation factor to the maximum value of f over the unit sphere that can be achieved using degree d SOS.

2 FINDING FORBIDDEN MINORS IN SUBLINEAR TIME

Deciding if an n-vertex graph G is planar is a classic algorithmic problem solvable in linear time [28]. The Kuratowski-Wagner theorem asserts that any non-planar graph must contain a K_5 or $K_{3,3}$-minor [29, 30]. Thus, certifying non-planarity is equivalent to producing such a minor, which can be done in linear time. Can we beat the linear time bound if we knew that G was "sufficiently" non-planar?

Assume random access to an adjacency list representation of a bounded degree graph, G. Suppose, for some constant $\varepsilon > 0$, one had to remove εn edges from G to make it planar. Can one find a forbidden (K_5 or $K_{3,3}$) minor in $o(n)$ time? It is natural to ask this question for any property expressible through forbidden minors. By the famous Robertson-Seymour graph minor theorem [31], any graph property $\mathcal{P}$ that is closed under taking minors can be expressed by a finite list of forbidden minors. We desire sublinear time algorithms to find a forbidden minor in any G that requires εn edge deletions to make it have $\mathcal{P}$.

This problem was first posed by Benjamini-Schramm-Shapira [32] in the context of property testing on bounded degree graphs. We follow the model of property testing on bounded degree graphs as defined by Goldreich-Ron [33]. Fix a degree bound d. Consider $G = (V, E)$, where $V = [n]$, and G is represented by an adjacency list. We have random access to the list through *neighbor queries*. There is an oracle that, given $v \in V$ and $i \in [d]$, returns the ith neighbor of v (if no neighbor exists, it returns $\perp$).

Given any property, $\mathcal{P}$, of graphs with degree bound d, the distance of G to $\mathcal{P}$ is defined to be the minimum number of edge additions/removals required to make G have $\mathcal{P}$ divided by dn. This ensures that the distance is in $[0, 1]$. We say that G is ε-far from $\mathcal{P}$ if the distance to $\mathcal{P}$ is more than ε.

A property tester for $\mathcal{P}$ is a randomized procedure that takes as input (query access to) G and a proximity parameter $\varepsilon > 0$. If $G \in \mathcal{P}$, the tester must accept with probability at least $2/3$. If G is ε-far from $\mathcal{P}$, the tester must reject with probability at least $2/3$. A one-sided tester must accept $G \in \mathcal{P}$ with probability 1, and thus must provide a certificate of rejection.

We are interested in properties expressible through *forbidden minors*. Fix a finite graph H. The property $\mathcal{P}_H$ of *H-minor-freeness* is the set of graphs that do not contain H as a minor. Observe that one-sided testers for $\mathcal{P}_H$ have a special significance since they must produce an H-minor whenever they reject. One can cast one-sided property testers for $\mathcal{P}_H$ as sublinear time procedures that find forbidden minors. Our main theorem follows.

Theorem 2.0.1 *Fix a finite graph H with $|V(H)| = r$ and arbitrarily small $\delta > 0$. Let $\mathcal{P}_H$ be the property of H-minor-freeness. There is a randomized algorithm that takes as input (oracle access to) a graph G with maximum degree d, and a parameter $\varepsilon > 0$. Its running time is $dn^{1/2+O(\delta r^2)} + d\varepsilon^{-2\exp(2/\delta)/\delta}$. If G is ε-far from $\mathcal{P}_H$, then, with probability $> 2/3$, the algorithm outputs an H-minor in G.*

Equivalently, there exists a one-sided property tester for $\mathcal{P}_H$ with the above running time.

The graph minor theorem of Robertson and Seymour [31] asserts the following. Consider any property $\mathcal{Q}$ that is closed under taking minors. There is a finite list $\boldsymbol{H}$ of graphs such that $G \in \mathcal{Q}$ iff G is H-minor-free for all $H \in \boldsymbol{H}$. If G is ε-far from $\mathcal{Q}$, then G is $\Omega(\varepsilon)$-far from $\mathcal{P}_H$ for some $H \in \boldsymbol{H}$. Thus, a direct corollary of Theorem 2.0.1 is the following.

Corollary 2.0.2 *Let $\mathcal{Q}$ be any minor-closed property of graphs with degree bound d. For any $\delta > 0$, there is a one-sided property tester for $\mathcal{Q}$ with running time $O(dn^{1/2+\delta} + d\varepsilon^{-2\exp(2/\delta)/\delta})$.*

In the following discussion, we suppress dependences on ε and n^δ by $O^*(\cdot)$ (where $\delta > 0$ is arbitrarily small). Previously, the only graphs H for which an analogue of

Theorem 2.0.1 was known are the following: $O^*(1)$ time for H being a forest, $O^*(\sqrt{n})$ for H being a cycle [34], and $O^*(n^{2/3})$ for H being $K_{2,k}$, the $(k \times 2)$-grid, and the k-circus [35]. No sublinear time bound was known for planarity.

Corollary 2.0.2 implies that properties such as planarity, series-parallel graphs, embeddability in bounded genus surfaces, and bounded treewidth are all one-sided testable in $O^*(\sqrt{n})$ time.

We note a particularly pleasing application of Theorem 2.0.1. Suppose a bounded degree graph, G, has more than $(3+\varepsilon)n$ edges. Then it is guaranteed to be ε-far from being planar, and thus, there is an algorithm to find a forbidden minor in G in $O^*(\sqrt{n})$ time. Since all minor-closed properties have constant average degree bounds, analogous statements can be made for all such properties.

2.0.1 Related work

Graph minor theory is a deep topic, and we refer the reader to Chapter 12 of Diestel's book [36] and Lovász's survey [37]. For our purposes, we use as a blackbox a polynomial time algorithm that finds fixed minors in a graph. A result of Kawarabayashi-Kobayashi-Reed provides an $O(n^2)$ time algorithm [24].

Property testing on graphs is an immensely rich area of study, and we refer the reader to Goldreich's recent textbook for more details [38]. There is a significant difference between the theory of property testing for dense graphs and that of bounded degree graphs. For the former, there is a complete characterization of properties (one-sided, non-adaptive) testable in query complexity independent of graph size. There is a deep connection between property testing and the Szemeredi regularity lemma [39]. Property testing for bounded degree graphs is much less understood. This study was initiated by Goldreich-Ron, and the first results focused on connectivity properties [33]. Czumaj-Sohler-Shapira proved that hereditary properties of non-expanding graphs are testable [40]. A breakthrough result of Benjamini-Schramm-Shapira (henceforth BSS) proved that all minor-closed (more generally, hyperfinite) properties are two-sided testable in constant time. The dependence on ε was subsequently improved by Hassidim et al, using the concept of local partitioning oracles [41]. A result of Levi-Ron [42] significantly simplified and improved this analysis, to get a final query complexity quasi-polynomial in $1/\varepsilon$. Indeed, it is a major open question to get polynomial dependence on $1/\varepsilon$ for two-sided planarity testers. Towards this goal, Ito and Yoshida give such a bound for testing outerplanarity [43], or Edelman et al generalize for bounded treewidth graphs [44].

In contrast to dense graph testing, there is a significant jump in complexity for one-sided testers. BSS first raised the question of one-sided testers for minor-closed properties (especially planarity) and conjectured that the bound is $O(\sqrt{n})$. Czumaj et al [34] made the first step by giving an $\widetilde{O}(\sqrt{n})$ one-sided tester for the property of being C_k-minor-free [34]. For $k = 3$, this is precisely the class of forests. This tester is obtained by a reduction to a much older result of Goldreich-Ron for one-sided bipartiteness testing for bounded degree graphs [45] (the results in Czumaj et al are obtained by black-box applications of this result). Czumaj et al adapt the one-sided $\Omega(\sqrt{n})$ lower bound for bipartiteness and show an $\Omega(\sqrt{n})$ lower bound for one-sided testers for H-minor-freeness when H has a cycle [34]. This is complemented with a constant time tester for H-minor-freeness when H is a forest.

Recently, Fichtenberger-Levi-Vasudev-Wötzel give an $\widetilde{O}(n^{2/3})$ tester for H-minor-freeness when H is one of the following graphs: $K_{2,k}$, the $(k \times 2)$-grid or the k-circus graph (a wheel where spokes have two edges) [35]. This subsumes the properties of outerplanarity and cactus graphs. This result uses a different, more combinatorial (as opposed to random walk based) approach than Czumaj et al.

The use of random walks in property testing was pioneered by Goldreich-Ron [45] and was then (naturally) used in testing expansion properties and clustering structure [46–51]. Our approach is inspired by the Goldreich-Ron analysis, and we discuss more in the next section. A number of previous results have used random walks for routing in expanders [52,53]. We use techniques from Kale-Seshadhri-Peres to analyze random walks on projected Markov Chains [50]. We also employ the local partitioning methods of Spielman-Teng [54], which is in turn derived from the Lovász-Simonovits analysis technique [3].

2.1 Main Ideas

We give an overview of the proof strategy and discuss the various moving parts of the proof. Assume that G is a d-regular graph. It is instructive to understand the method of Goldreich-Ron (henceforth GR) for one-side bipartiteness testing [45]. The basic idea to perform $O(\sqrt{n})$ random walks of $\mathrm{poly}(\log n)$ length from a uar vertex s. An odd cycle is discovered when two walks end at the same vertex v, through path of differing parity (of length).

The GR analysis first considers the case when G is an expander (and ε-far from bipartite). In this case, the walks from s reach the stationary distribution. One can use a standard collision argument to show that $O(\sqrt{n})$ suffice to hit the same vertex v twice, with different parity paths. The deep insight is that any graph G can be decomposed into pieces where the algorithm works, and each piece P has a small cut to $\overline{P}$. This has connections with decomposing a graph into expander-like pieces [55, 56]. Famously, the Arora-Barak-Steurer algorithm [4] for unique games basically proves such a statement. We note that GR does *not* decompose into expanders, but rather into pieces where the expander analysis goes through. So, one might hope to analyze the algorithm by its behavior on each component. Unfortunately, the algorithm cannot produce the decomposition; it can only walk in G and hope that performing random walks in G suffice to simulate the procedure within P. This is extremely challenging, and is precisely what GR achieve (this is the bulk of the analysis). The main lemma produces a decomposition into such pieces, such that for each piece P, there exists $s \in P$ wherein short random walks (in G) from s reach all vertices in P with sufficient probability. One can think of this a simulation argument: we would like to simulate the random walk algorithm running only on P, through random walks in G.

The challenge of general minors: With planarity in mind, let us focus on finding K_5 minors. It is highly unlikely that random walks from a single vertex will find a such a minor. Intuitively, we would need to find 5 different vertices, launch random walks from all of them and hope these walks will produce a minor. Thus, we would need to simulate a much more complex procedure than the (odd) cycle finder of GR. Most significantly, we need to understand the random walks behavior from multiple sources within P simultaneously. The GR analysis actually constructs the pieces P by a local partitioning looking at the random walk distribution from a single vertex. There is no guarantee on random walk behavior from other vertices in P.

There is a more significant challenge from arbitrary minors. The simulation does not say anything about the specific structure of the paths generated. It only deals with the probability of reaching v from s by a random walk in G when v and s are in the same piece. For bipartiteness, as long as we find two paths of differing parity, we are done. They may intersect each other arbitrarily. For finding a K_5 minor, the actual intersection matter. We would need paths between all pairs of 5 seed vertices to be "disjoint enough" to give a K_5 minor. This appears extremely difficult using the GR analysis. Even if we did understand the random walk behavior (in G) from all vertices in P, we have little control over their behavior when they leave P. (Based on the parameters, the walks leave P with high probability.) They may intersect arbitrarily, and thus destroy any minor structure.

2.1.1 When do random walks find minors?

Inspired by GR, let us start with an algorithm to find a K_5 minor in an expander G (variants of these ideas were present in a result of Kleinberg-Rubinfeld that expanders contain an H-minor for any H with $n/\text{poly}(\log n)$ edges [53]). Let ℓ denote the mixing time. Pick u.a.r. a vertex, s, and launch 5 random walks each of length ℓ to reach $v_1, v_2, \ldots, v_5$. From each v_i, launch $\sqrt{n}$ random walks each of length ℓ. With high probability, a walk from v_i and a walk from v_j will "collide" (end at the same vertex).

We can collect these collisions to get paths between all v_i, v_j, and one can, with some effort, show that these form a K_5-minor.

Our main insight is to show that this algorithm, with minor modifications, works even when random walks have extremely slow mixing properties. When the random walks mix even more slowly than the requisite bound, we can essentially perform local partitioning to pull out very small (n^δ for arbitrarily small $\delta > 0$) pieces that have low conductance cuts. We can simply query all edges in this piece and run a planarity test.

There is a parameter $\delta > 0$ that can be set to an arbitrarily small constant. Let us set the random walk length ℓ to n^δ, and let $\mathbf{p}_{s,\ell}$ be the random walk distribution after ℓ steps from s. Our proof splits into two cases, where $\alpha = c\delta$ for explicit constant $c > 1$:

- Case 1 (the leaky case): For at least εn vertices s, $\|\mathbf{p}_{s,\ell}\|_2^2 \leq 1/n^\alpha$.
- Case 2 (the trapped case): For at least $(1-\varepsilon)n$ vertices s,$\|\mathbf{p}_{s,\ell}\|_2^2 > 1/n^\alpha$.

In the leaky case, random walks are hardly mixing by any standard of convergence. We are merely requiring that a random walk of length n^δ (roughly speaking) spreads to a set of size $n^{c\delta}$.

We prove that, in the leaky case, the procedure described in the first paragraph succeeds in finding a K_5 with high probability. We give an outline of this proof strategy.

Let us assume that $\mathbf{p}_{v,\ell/2} = \mathbf{p}_{v,\ell}$ (so ℓ-length walks have "stabilized"). Let us make a slight modification to the algorithm. We pick $v_1, \dots, v_5$ as before, with ℓ-length random walks from s. We will perform $O(\sqrt{n})$ $\ell/2$ length random walks from each v_i to produce the K_5 minor. By symmetry of the random walks, the probability that a single walk from v_i and one from v_j collide (to produce a path) is exactly $\mathbf{p}_{v_i,\ell/2} \cdot \mathbf{p}_{v_j,\ell/2}$. Thus, we would like these dot products to be large. By the symmetry of the random walk, the probability of an ℓ-length random walk starting from s and ending at v is $\mathbf{p}_{s,\ell/2} \cdot \mathbf{p}_{v,\ell/2}$. In other words, the entries of $\mathbf{p}_{s,\ell}$ are precisely these dot products, and $\|\mathbf{p}_{s,\ell}\|_2^2 = \sum_{v \in V} (\mathbf{p}_{s,\ell/2} \cdot \mathbf{p}_{v,\ell/2})^2 = \mathbf{E}_{v \sim \mathbf{p}_{s,\ell/2}}[\mathbf{p}_{s,\ell/2} \cdot \mathbf{p}_{v,\ell/2}]$. Since $\mathbf{p}_{s,\ell/2} = \mathbf{p}_{s,\ell}$, we rewrite to get $\mathbf{p}_{s,\ell/2} \cdot \mathbf{p}_{s,\ell/2} = \mathbf{E}_{v \sim \mathbf{p}_{s,\ell/2}}[\mathbf{p}_{s,\ell/2} \cdot \mathbf{p}_{v,\ell/2}]$.

Think of the dot products as correlations between distributions. We are saying that the average correlation (over some distribution on vertices) of $\mathbf{p}_{v,\ell/2}$ with $\mathbf{p}_{s,\ell/2}$ is exactly the self-correlation of $\mathbf{p}_{s,\ell/2}$. If the distributions by and large had low ℓ_2-norm (as in the leaky case), we might hope that these distributions are reasonably correlated with each other. Indeed, this is what we prove. Under some conditions, we show that $\mathbf{E}_{v_i, v_j \sim \mathbf{p}_{s,\ell/2}}[\mathbf{p}_{v_i,\ell/2} \cdot \mathbf{p}_{v_j,\ell/2}]$ can be lower bounded, where $\mathbf{p}_{s,\ell/2}$ is exactly the distribution the algorithm picks the v_i and v_j from. This is evidence that $\ell/2$-length random walks will connect the v_i's through collisions.

There are four difficulties in increasing order of worry.

1. We only have a lower bound of the average $\mathbf{p}_{v_i,\ell/2} \cdot \mathbf{p}_{v_j,\ell/2}$. We would need bounds for all (or most) pairs to produce a minor.
2. $\mathbf{p}_{v,\ell}$ might be very different from $\mathbf{p}_{v,\ell/2}$.
3. The expected number of collisions between walks from v_i and v_j is controlled by the dot product above, but the variance (which really controls the probability of getting a collision) can be large. There are instances where the dot product is high, but the collision probability is extremely low.
4. There is no guarantee that these paths will produce a minor since we do not have any obvious constraints on the intermediate vertices in the path.

The first problem is surmounted by a technical trick. It turns out to be cleaner to analyze the probability of getting a biclique minor. So, we perform 50 random walks from s to get sets $A = \{a_1, a_2, \ldots, a_{25}\}$ and an analogous B. We launch $\ell/2$-length random walks from each vertex in $A \cup B$. The average lower bound on the dot product suffices to get a lower bound on the probability of getting a $K_{25,25}$-minor, which contains a K_5-minor.

For the second problem, what we can show is that the weaker bound of $\|\mathbf{p}_{v,\ell}\|_2 = \Omega(n^{-\delta}\|\mathbf{p}_{v,\ell/2}\|_2)$ suffices. We could try to search for some value of ℓ where this happens. If there was no (small) value of ℓ where this bound held, then it suggest that $\|\mathbf{p}_{v,n^\delta}\|_2$

is extremely small (say $\Theta(1/n)$). This kind of reasoning is detailed more in the next subsection.

The third problem requires bounds on the variance, or higher norms, of $\mathbf{p}_{v,\ell/2}$. Unfortunately, there appears be no handle on these. At a high level, our idea is to truncate $\mathbf{p}_{v,\ell/2}$ by ignoring large entries. This truncated vector is not a probability vector any more, but we can hope to redo the analysis for such vectors.

Now for the fourth problem. Naturally, if the vertices $v_1, \ldots, v_5$ are close to each other, we do not expect to get a minor by connecting them. Suppose they were sufficiently "spread out", One could hope that the paths connecting the v_i, v_j pairs would only intersect "near" the v_i. The portion of the paths nears the v_i's could be contracted to get a K_5-minor. We can roughly quantify how far the v_i's will be by the variance of $\mathbf{p}_{v,\ell/2}$. Thus, the third and fourth problem are coupled.

2.1.2 R-returning walks

The main technical contribution of our work is in defining R-returning walks. These are walks that periodically return to a given set of vertices, R. A careful analysis of these walks provides the tools to handle the various problems discussed above.

Fix ℓ as before. Formally, an R-returning walk of length $j\ell$ (for $j \in \mathbb{N}$) is a walk that encounters R at every $i\ell$ step $\forall i \in [j]$. While random walk distributions can have poor variance, we can carefully choose R to ensure that the distribution of R-returning walks is well-behaved. We will quantify this as approximate "support uniformity" (being approximatedly uniform on the support).

In the leaky case, there is some (large) set, R, such that $\forall s \in R$, $\|\mathbf{p}_{s,\ell/2}\|_2^2 \leq 1/n^{\alpha}$. Let $\mathbf{p}_{[R],s,\ell}$ be the random walk distribution restricted to R. Suppose for some $s \in R$, $\|\mathbf{p}_{[R],s,\ell}\|_2^2 \geq 1/n^{\alpha+\delta}$. Observe that each entry in $\mathbf{p}_{[R],s,\ell}$ is $\mathbf{p}_{s,\ell/2} \cdot \mathbf{p}_{v,\ell/2}$, for $s, v \in R$. By Cauchy-Schwartz, this is at most $1/n^{\alpha}$. For any distribution $\mathbf{v}$, the condition $\|\mathbf{v}\|_2^2 = \|\mathbf{v}\|_\infty$ is equivalent to support uniformity. Thus, $\mathbf{p}_{[R],s,\ell}$ is approximately support uniform, up to n^δ deviations. The math discussed in the previous section goes through for any such s. In other words, if the random walk algorithm started from s, it succeeds in finding a K_5 minor.

Suppose only a negligible fraction of vertices satisfied this condition, and so our algorithm would not actually find such a vertex. Let us remove all these vertices from R (abusing notation, let R be the resulting set). Now, $\forall s \in R$, $\|\mathbf{p}_{[R],s,\ell}\|_2^2 \leq 1/n^{\alpha+\delta}$. So, the bound on the l_2-norm has fallen by an n^δ factor. What does $\mathbf{p}_{[R],s,\ell} \cdot \mathbf{p}_{[R],v,\ell}$ signify? This is the probability of a 2ℓ-length random walk starting from s, ending at v, and encountering R at the ℓth step. This is an R-returning walk of length 2ℓ. Let $\mathbf{q}_{[R],s,2\ell}$ denote the vector of R-returning walk probabilities. Suppose for some s, $\|\mathbf{q}_{[R],s,2\ell}\|_2^2 \geq 1/n^{\alpha+2\delta}$. By Cauchy-Schwartz, $\|\mathbf{q}_{[R],s,2\ell}\|_\infty \leq 1/n^{\alpha+\delta}$, implying that $\mathbf{q}_{[R],s,2\ell}$ is approximately support uniform. Again, the math of the previous section goes through for such an s.

We remove all vertices that have this property, and end up with R such that $\forall s \in R$, $\|\mathbf{q}_{[R],s,2\ell}\|_2^2 \leq 1/n^{\alpha+2\delta}$. Observe that $\mathbf{q}_{[R],s,2\ell} \cdot \mathbf{q}_{[R],v,2\ell}$ is a probability of a 4ℓ R-returning walk. We then iterate this argument.

In general, this argument goes through phases. In the ith phase, we find all $s \in R$ that satisfy $\|\mathbf{q}_{[R],s,2^i\ell}\|_2^2 \geq 1/n^{\alpha+i\delta}$. We show that the random walk procedure of the previous section (with some modifications) finds a K_5-minor starting from such vertices. We remove all such vertices from R, increment i and continue the argument. The vertices removed at the ith phase are called the *ith stratum*, and we refer to this entire process as stratification. Intuitively, for vertices in the ith stratum, the R-returning (for the setting of R at that phase) walk probabilities roughly form a uniform distribution of support $n^{\alpha+i\delta}$. Thus, for vertices in higher strata, the random walks are spreading to larger sets.

There is a major problem. The $\mathbf{q}$ vectors are *not* distributions, and the vast majority of walks are not R-returning. Indeed, the reduction in norm as we increase strata might simply be an artifact of the lower probability of a longer R-returning walk (note that the walks lengths are increasing exponentially in the phase number). We prove a spectral lemma asserting that this is not the case. As long as R is sufficiently large, the probabilities of R-returning walks are sufficiently high. Unfortunately, these probabilities (must) decrease exponentially in the number of returns. In the ith phase, the walk length is $2^i\ell$ and it must return to R 2^i times. Here is where the n^δ decay in l_2-norm condition saves us. After $1/\delta$ phases, the $\|\mathbf{q}_{[R],s,2^i\ell}\|_2^2$ is basically $1/n$. The spectral lemma tells us that if R is still large, the probability that a $2^{1/\delta}\ell$ length walk is R-returning is sufficiently large. Thus, the norm cannot decrease, and almost all vertices end up in the very next stratum. If R was small, then there is an earlier stratum containing $\Omega(\delta\varepsilon n)$ vertices. Regardless of the case, there exists a $i \leq 1/\delta + O(1)$ such that the ith stratum contains $\Omega(\delta\varepsilon n)$ vertices. For all these vertices, the random walk algorithm to find minors succeeds with non-trivial probability.

2.1.3 The trapped case: local partitioning to the rescue

In this case, for almost all vertices $\|\mathbf{p}_{s,\ell}\|_2^2 \geq 1/n^\alpha$. The proofs of the (contrapositive of the) Cheeger inequality basically imply the existence of a set of low condutance cut P_s "around" s. By local partitioning methods such as those of Spielman-Teng and Anderson-Chung-Lang [54, 57], we can actually find P_s in roughly n^α time. We expect our graph to basically decompose into $O(n^\alpha)$ sized components with few edges between them. Our algorithm can simply find these pieces P_s and run a planarity test on them. We refer to this as the *local search* procedure.

While the intuition is correct, the analysis is difficult. The main problem is that actual partitioning of the graph (into small components connected by low conductance cuts) is fundamentally iterative. It starts by finding a low conductance set P_{s_1}, then finding a low conductance set P_{s_2} in $\overline{P_{s_1}}$, then P_{s_3} in $\overline{P_{s_1} \cup P_{s_2}}$, and so on. In general, this requires conditions on the random walk behavior inside $\overline{\bigcup_{j<i} P_{s_j}}$. On the other hand, our algorithm and the trapped case condition only refer to random

walk behavior in all of G. Furthermore, $\overline{\bigcup_{j<i} P_{s_j}}$ can be as small as $\Theta(\varepsilon n)$, and so we do expect the random walk behavior to be quite different.

The GR bipartiteness analysis surmounts this problem and performs such a decomposition, but their parameters do not work for us. Starting from a source vertex s, their analysis discovers P_s such that probabilities of reaching any vertex in P_s (from s) is roughly uniform *and* smaller than $1/\sqrt{n}$. On the other hand, we would like to discover all of P_s in $n^{O(\delta)}$ time so that we can run a full planarity test.

We employ a collection of tools, and use the methods of Kale-Peres-Seshadhri to analyze "projected" Markov Chains [50]. In the analysis above, we have some set S ($\overline{\bigcup_{j<i} P_{s_j}}$) and want to find a low conductance set P completely contained in S. Moreover, we wish to discover P using random walks in G. We construct a Markov chain, M_S, with vertex set S, and include new transitions that correspond to walks in G whose intermediate vertices are not in S. Each such transition has an associated "cost," corresponding to the actual length in G. (GR also have a similar idea, although their Markov chain introduces extra vertices to track the length of the walk in G. This makes the analysis somewhat unwieldy, since low conductance cuts in M_S may include these extra vertices.)

Using bounds on the return time of random walks, we have relationships between the average length of a walk in G whose endpoints are in S and the corresponding length when "projected" to M_S. On average, an ℓ-length walk in G with endpoints in S corresponds to an $\ell|S|/n$-length walk in M_S. Roughly speaking, we hope that for many vertices s, an $\ell|S|/n$-length walk in M_S is trapped in a set of size n^α.

We employ the Lovász-Simonovits curve technique to produce a low conductance cut P_s in M_S [3]. We can guarantee that all vertices in P_s are reachable with roughly $n^{-\alpha}$ probability from s through $\ell|S|/n$-length random walks in M_S. Using the average length correspondence between walks in M_S to G, we can make a similar statement in G - albeit with a longer length. We basically iterate over this entire argument to produce the decomposition into low conductance pieces.

In our analysis, we use the stratification itself to (implicitly) distinguish between the leaky and trapped case. Stratification peels the graph into $1/\delta + O(1)$ strata. If a vertex s lies in a stratum numbered at least some fixed constant b, we can show that the algorithm finds a K_r-minor with s as the starting vertex. Thus, if at least (say) $n^{1-\delta}$ vertices lie in stratum b or higher, we are done. If s is in a low stratum, we have a lower bound on the random walks norm. This allows for local partitioning around s.

2.2 The algorithm

We are given a bounded degree graph $G = (V, E)$, with max degree d. We assume that $V = [n]$. We follow the standard adjacency list model of Goldreich-Ron for (random) access to the graph. This model allows an algorithm to sample u.a.r. vertices and perform *edge queries*. Given a pair $(v, i) \in [n] \times [d]$, the output of an edge query is the ith neighbor of v according to the adjacency list ordering. If the degree of v is smaller than i, the output is $\perp$.

In the algorithm, the phrase "random walk" refers to a lazy random walk on G. Given a current vertex, v, with probability $1/2$, the walk remains at v. With probability $1/2$, the procedure generates u.a.r. $i \in [d]$. It performs the edge query for (v, i). If the output is $\perp$, the walk remains at v, otherwise the walk visits the output vertex. This is a symmetric, ergodic Markov chain with a uniform stationary distribution.

Our main procedure `FindMinor`(G, ε, H), tries to find a H-minor in G. We prove that it succeeds with high probability if G is ε-far from being H-minor-free. There are three subroutines:

- `LocalSearch`(s): This procedure perform a small number of short random walks to find the piece described in §2.1.3. This produces a small subgraph of G, where an exact H-minor finding algorithm is used.

- `FindPath`(u, v, k, i): This procedure tries to find a path from u to v. The parameter i decides the length of the walk, and the procedure performs k walks from u and v. If any pair of these walks collide, this path is output.

- `FindBiclique`(s): This is the main procedure mostly as described in §2.1.1. It attempts to find a sufficiently large biclique minor. First, it generates seed sets A and B by performing random walks from s. Then, it calls `FindPath` on all pairs in $A \times B$.

We fix a collection of parameters.

- δ: An arbitrarily small constant.
- r: The number of vertices in H.
- ℓ: The random walk length. This will be $n^{5\delta}$.
- $\varepsilon_{\mathsf{CUTOFF}}$: $\varepsilon_{\mathsf{CUTOFF}} = n^{\frac{-\delta}{exp(2/\delta)}}$. If $\varepsilon < \varepsilon_{\mathsf{CUTOFF}}$, the algorithm just queries the whole graph.
- `KKR`(F, H): This refers to an exact H-minor finding process (in F). For concreteness, we use the quadratic time procedure of Kawarabayashi-Kobayashi-Reed [24].

`FindMinor`(G, ε, H)

1. If $\varepsilon < \varepsilon_{\mathsf{CUTOFF}}$, query all of G, and output `KKR`(G, H)
2. Else
   (a) Repeat $\varepsilon^{-2} n^{35\delta r^2}$ times:
      i. Pick uar $s \in V$
      ii. Call `LocalSearch`(s) and `FindBiclique`(s).

`LocalSearch`(s)

1. Initialize set $B = \emptyset$.
2. For $h = 1, \dots, n^{7\delta r^4}$:
   (a) Perform $\varepsilon^{-1} n^{30\delta r^4}$ independent random walks of length h from s. Add all destination vertices to B.
3. Determine $G[B]$, the subgraph induced by B.
4. Run `KKR`$(G[B], H)$. If it returns an H-minor, output that and terminate.

`FindBiclique`(s)

1. For $i = 5r^4, \dots, 1/\delta + 4$:
   (a) Perform $2r^2$ independent random walks of length $2^{i+1}\ell$ from s. Let the destinations of the first r^2 walks be multiset A, and the destinations of the remaining walks be B.
   (b) For each $a \in A$, $b \in B$:
      i. Run `FindPath`$(a, b, n^{\delta(i+18)/2}, i)$
   (c) If all calls to `FindPath` return a path, then let the collection of paths be the subgraph F. Run `KKR`(F, H). If it returns an H-minor, output that and terminate.

`FindPath`(u, v, k, i)

1. Perform k random walks of length $2^i\ell$ from u and v.
2. If a walk from u and v terminate at the same vertex, return these paths. (Otherwise, return nothing.)

Theorem 2.2.1 *If G is ε-far from being H-minor-free, then* `FindMinor`(G, ε, H) *finds an H-minor of G with probability at least $2/3$. Furthermore,* `FindMinor` *has a running time of $dn^{1/2+O(\delta r^2)} + d\varepsilon^{-2\exp(2/\delta)/\delta}$.*

The query complexity is fairly easy to compute. The total queries made in the `LocalSearch` calls is $dn^{O(\delta r^4)}$. The main work happens in the calls of `FindPath`, within `FindBiclique`. Observe that k is set to $n^{\delta(i+18)/2}$, where $i \leq 1/\delta + 4$. This leads to the

$\sqrt{n}$ in the final complexity. (In general, a setting of $\delta < 1/\log(\varepsilon^{-1}\log\log n)$ suffices for an $n^{1/2+o(1)}$ running time.)

Outline: There are a number of moving parts in the proof, which we relegate to their own subsections. We first develop the notion of R-returning walks and the stratification process in §2.3. In §2.4, we use these techniques to prove that `FindBiclique` discovers a sufficiently large biclique-minor in the leaky case. In §2.5, we prove a local partitioning lemma that will be used to handle the trapped case. Finally, in §2.6, we put the tools together to complete the proof of Theorem 2.2.1.

2.3 Returning walks and stratification

We introduce the concept of R-returning random walks for any $R \subseteq V$. These definitions are with respect to a fixed length ℓ.

Definition 2.3.1 *For any set of vertices R, $s \in R$, $u \in R$, and $i \in \mathbb{N}$, we define the* R-returning probability *as follows. We denote by $q^{(i)}_{[R],s}(u)$ the probability that a $2^i\ell$-length random walk from s ends at u, and encounters a vertex in S at every $j\ell^{th}$ step, for all $1 \leq j \leq 2^i$. The R-returning probability vector, denoted by $\boldsymbol{q}^{(i)}_{[R],s}$, is the $|R|$-dimensional vector of returning probabilities.*

Proposition 2.3.1 $q^{(i+1)}_{[R],s}(u) = \boldsymbol{q}^{(i)}_{[R],s} \cdot \boldsymbol{q}^{(i)}_{[R],u}$

Proof We use the symmetry of (returning) random walks in G.

$$q^{(i+1)}_{[R],s}(u) = \sum_{w \in S} q^{(i)}_{[R],s}(w) q^{(i)}_{[R],w}(u) = \sum_{w \in R} q^{(u)}_{[R],s}(w) q^{(i)}_{[R],u}(w) = \boldsymbol{q}^{(i)}_{[R],s} \cdot \boldsymbol{q}^{(i)}_{[R],u}$$

■

Let M be the transition matrix of the lazy random walk on G. Let $\mathbb{P}_R$ be the $n \times |R|$ matrix on R, where each column is the unit vector for some $s \in R$. For any set U, we use $\mathbf{1}_U$ for the indicator vector on U. If no subscript is given, it is the all ones vector, for the appropriate dimension.

Proposition 2.3.2 $\boldsymbol{q}^{(i)}_{[R],s} = (\mathbb{P}_R^T M^\ell \mathbb{P}_R)^{2^i} \mathbf{1}_s$

Now for a critical lemma. We can lower bound the total probability of an R-returning random walk. If R contains at least a β-fraction of vertices, the average R-returning walk probability, for t returns, is at least β^t.

Lemma 2.3.1 $|R|^{-1} \sum_{s \in R} \|\boldsymbol{q}^{(i)}_{[R],s}\|_1 \geq (|R|/n)^{2^i}$

Proof We will express $\sum_{s \in R} \|\boldsymbol{q}^{(i)}_{[R],s}\|_1 = \mathbf{1}^T(\mathbb{P}_R^T M^\ell \mathbb{P}_R)^{2^i}\mathbf{1}$. Let us first prove the lemma for $i = 0$. Note $\sum_{s \in R} \|\boldsymbol{q}^{(0)}_{[R],s}\|_1 = \mathbf{1}_R^T M^\ell \mathbf{1}_R = ((M^T)^{\ell/2}\mathbf{1}_R)^T(M^{\ell/2}\mathbf{1}_R) = \|M^{\ell/2}\mathbf{1}_R\|_2^2$. Since $M^{\ell/2}$ is a stochastic matrix, $\|M^{\ell/2}\mathbf{1}_R\|_1 = \|\mathbf{1}_R\|_1 = |R|$. By a standard norm inequality, $\|M^{\ell/2}\mathbf{1}_R\|_2^2 \geq \|M^{\ell/2}\mathbf{1}_R\|_1^2/n = |R|^2/n$. This completes the proof for $i = 0$.

Let $N = \mathbb{P}_R^T M^\ell \mathbb{P}_R$, which is a symmetric matrix. We have just proven that $\mathbf{1}^T N \mathbf{1} \geq |R|^2/n$. Let the eigenvalues of N be $1 \geq \lambda_1 \geq \lambda_2 \dots \lambda_{|R|}$, with corresponding eigenvectors $\mathbf{u_1}, \mathbf{u_2}, \dots, \mathbf{u_s}$. We can express $\mathbf{1} = \sum_{k \leq |R|} \alpha_k \mathbf{u_k}$, where $\sum_k \alpha_k^2 = |R|$. Observe that $N^{2^i}\mathbf{1} = \sum_{k \leq |R|} \alpha_k \lambda_k^{2^i} \mathbf{u_k}$

Let $\mu_k = \alpha_k^2 / \sum_j \alpha_j^2$, noting that $\sum_k \mu_k = 1$. We apply Jensen's inequality below.

$$\frac{\mathbf{1}^T N^{2^i} \mathbf{1}}{|R|} = \frac{\sum_k \alpha_k^2 \lambda_k^{2^i}}{\sum_j \alpha_j^2} = \sum_k \mu_k \lambda_k^{2^i} \geq (\sum_k \mu_k \lambda_k)^{2^i}$$

For $i = 0$, we already proved that $\mathbf{1}^T N \mathbf{1}/|R| = \sum_k \mu_k \lambda_k \geq |R|/n$. We plug this bound to complete the proof for general i. ■

2.3.1 Stratification

Stratification results in a collection of disjoint sets of vertices denoted by $S_0, S_1, \dots$ which are called *strata*. The corresponding *residue* sets denoted by $R_0, R_1, \dots$. The zeroth residue R_0 is initialized before stratification and subsequent residues are defined by the recurrence $R_i = R_0 \setminus \bigcup_{j<i} S_j$. The definitions and claims may seem technical, and the proofs are mostly norm manipulations. But these provide the tools to analyze our main algorithm.

Definition 2.3.2 *Suppose R_i has been constructed. A vertex $s \in R_i$ is placed in S_i if $\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_2^2 \geq 1/n^{\delta i}$.*

We have an upper bound for the length of R_i-returning walk vectors.

Claim 2.3.2 *For all $s \in R_i$ and $1 \leq j \leq i$, $\|\boldsymbol{q}^{(j)}_{[R_i],s}\|_2^2 \leq 1/n^{\delta(j-1)}$.*

Proof Suppose $\exists j \leq i, \|\boldsymbol{q}^{(j)}_{[R_i],s}\|_2^2 > 1/n^{\delta(j-1)}$. By assumption, $s \in R_i \subseteq R_{j-1}$. An R_i-returning walk from s is also an R_{j-1}-returning walk. Thus, every entry of $\boldsymbol{q}^{(j)}_{[R_{j-1}],s}$ is at least that of $\boldsymbol{q}^{(j)}_{[R_i],s}$. So $\|\boldsymbol{q}^{(j)}_{[R_{j-1}],s}\|_2^2 \geq \|\boldsymbol{q}^{(j)}_{[R_i],s}\|_2^2 > 1/n^{\delta(j-1)}$. This implies that $s \in S_{j-1}$ or an earlier stratum, contradicting the assumption that $s \in R_i$. ■

We prove an ℓ_∞ bound on the returning probability vectors. Note that we allow j to be $i+1$ in the following bound.

Claim 2.3.3 *For all $s \in R_i$ and $2 \leq j \leq i+1$, $\|\boldsymbol{q}^{(j)}_{[R_i],s}\|_\infty \leq 1/n^{\delta(j-2)}$.*

Proof By Prop. 2.3.1, for any $v \in R_i$, $q^{(j)}_{[R_i],s}(v) = \boldsymbol{q}^{(j-1)}_{[R_i],s} \cdot \boldsymbol{q}^{(j-1)}_{[R_i],v}$. Note that $1 \leq j-1 \leq i$. By Cauchy-Schwartz and Claim 2.3.2, $q^{(j)}_{[R_i],s}(v) \leq 1/n^{\delta(j-2)}$. ■

As a consequence of these bounds, we are able to bound the amount of probability mass retained by R_i-returning walks.

Claim 2.3.4 *For all $s \in S_i$, $||\boldsymbol{q}^{(i+1)}_{[R_i],s}||_1 \geq n^{-\delta}$.*

Proof Since $s \in S_i$, $||\boldsymbol{q}^{(i+1)}_{[R_i],s}||_2^2 \geq n^{-i\delta}$, and by Claim 2.3.3, $||\boldsymbol{q}^{(i+1)}_{[R_i],s}||_\infty \leq n^{-\delta(i-1)}$. Since, $||\boldsymbol{q}^{(i+1)}_{[R_i],s}||_2^2 \leq ||\boldsymbol{q}^{(i+1)}_{[R_i],s}||_1 ||\boldsymbol{q}^{(i+1)}_{[R_i],s}||_\infty$, we conclude $||\boldsymbol{q}^{(i+1)}_{[R_i],s}||_1 \geq n^{-i\delta} n^{\delta(i-1)} = n^{-\delta}$. ■

We prove that most vertices lie in "early" strata.

Lemma 2.3.5 *Suppose $\varepsilon \geq \varepsilon_{\mathsf{CUTOFF}}$. At most $\varepsilon n/\log n$ vertices are in $R_{1/\delta+3}$.*

Proof We prove by contradiction. Suppose that $R_{1/\delta+3}$ has at least $\varepsilon n/\log n$ vertices. The previous residue, $R_{1/\delta+2}$, is only bigger and thus $|R_{1/\delta+2}| \geq \varepsilon n/\log n$ as well. By Lemma 2.3.1,

$$|R_{1/\delta+2}|^{-1} \sum_{s \in R_{1/\delta+2}} \|\boldsymbol{q}^{(1/\delta+3)}_{[R_{1/\delta+2}],s}\|_1 \geq \left(\frac{\varepsilon}{\log n}\right)^{2^{1/\delta+3}}. \tag{2.1}$$

By averaging and a standard l_1-l_2 norm inequality,

$$||\boldsymbol{q}^{(1/\delta+3)}_{[R_{1/\delta+2}],s}\|_2^2 \geq n^{-1}\left(\frac{\varepsilon}{\log n}\right)^{2^{1/\delta+4}}. \tag{2.2}$$

By assumption, $\varepsilon \geq \varepsilon_{\mathsf{CUTOFF}} \geq n^{-\delta/\exp(1/\delta)}$. For sufficiently small δ, $\delta/\exp(1/\delta) < 2\delta/2^{1/\delta+4}$. Thus, $\varepsilon \geq (\log n)n^{-2\delta/(2^{1/\delta+4})}$. Plugging into the RHS of the previous equation, $||\boldsymbol{q}^{(1/\delta+3)}_{[R_{1/\delta+2}],s}\|_2^2 \geq 1/n^{1+2\delta} = 1/n^{\delta(1/\delta+2)}$. This implies that $v \in S_{1/\delta+2}$ - a contradiction. ■

2.3.2 The correlation lemma

The following lemma is an important tool in our analysis. Here is an intuitive explanation. Fix some $s \in S_i$. By Prop. 2.3.1, the probability $q^{(i+1)}_{[R_i],s}(v)$ is the correlation between the vectors $\boldsymbol{q}^{(i)}_{[R_i],s}$ and $\boldsymbol{q}^{(i)}_{[R_i],v}$. If many of these probabilities are large, then there are many v such that $\boldsymbol{q}^{(i)}_{[R_i],v}$ is correlated with $\boldsymbol{q}^{(i)}_{[R_i],s}$. We then expect many of these vectors are correlated among themselves.

Definition 2.3.3 *For $s \in R_i$, the distribution $\mathcal{D}_{s,i}$ has support R_i, and the probability of $u \in R_i$ is $\hat{q}^{(i+1)}_{[R_i],s}(v) = q^{(i+1)}_{[R_i],s}(v)/\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1$.*

Lemma 2.3.6 *Fix arbitrary $s \in R_i$.*

$$\mathbf{E}_{u_1,u_2 \sim \mathcal{D}_{s,i}}[\boldsymbol{q}^{(i)}_{[R_i],u_1} \cdot \boldsymbol{q}^{(i)}_{[R_i],u_2}] \geq \frac{1}{\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1^2} \cdot \frac{\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_2^4}{\|\boldsymbol{q}^{(i)}_{[R_i],s}\|_2^2}$$

Proof

$$\begin{aligned}
&\mathbf{E}_{u_1,u_2\sim\mathcal{D}_{s,i}}[\boldsymbol{q}^{(i)}_{[R_i],u_1}\cdot\boldsymbol{q}^{(i)}_{[R_i],u_2}] && (2.3)\\
=&\sum_{u_1,u_2\in R_i}\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1^{-2}q^{(i+1)}_{[R_i],s}(u_1)q^{(i+1)}_{[R_i],s}(u_2)\boldsymbol{q}^{(i)}_{[R_i],u_1}\cdot\boldsymbol{q}^{(i)}_{[R_i],u_2} && (2.4)\\
=&\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1^{-2}\sum_{u_1,u_2\in R_i}(\boldsymbol{q}^{(i)}_{[R_i],s}\cdot\boldsymbol{q}^{(i)}_{[R_i],u_1})(\boldsymbol{q}^{(i)}_{[R_i],s}\cdot\boldsymbol{q}^{(i)}_{[R_i],u_2})(\boldsymbol{q}^{(i)}_{[R_i],u_1}\cdot\boldsymbol{q}^{(i)}_{[R_i],u_2}) \quad (\text{Prop. 2.3.1}) && \\
& && (2.5)\\
=&\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1^{-2}\sum_{u_1,u_2\in R_i}(\boldsymbol{q}^{(i)}_{[R_i],s}\cdot\boldsymbol{q}^{(i)}_{[R_i],u_1})(\boldsymbol{q}^{(i)}_{[R_i],s}\cdot\boldsymbol{q}^{(i)}_{[R_i],u_2})\sum_{w\in R_i}q^{(i)}_{[R_i],u_1}(w)q^{(i)}_{[R_i],u_2}(w)) && (2.6)\\
=&\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1^{-2}\sum_{w\in R_i}\sum_{u_1,u_2\in R_i}[(\boldsymbol{q}^{(i)}_{[R_i],s}\cdot\boldsymbol{q}^{(i)}_{[R_i],u_1})q^{(i)}_{[R_i],u_1}(w)][(\boldsymbol{q}^{(i)}_{[R_i],s}\cdot\boldsymbol{q}^{(i)}_{[R_i],u_2})q^{(i)}_{[R_i],u_2}(w)] && (2.7)\\
=&\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1^{-2}\sum_{w\in R_i}\left[\sum_{u\in R_i}(\boldsymbol{q}^{(i)}_{[R_i],s}\cdot\boldsymbol{q}^{(i)}_{[R_i],u})q^{(i)}_{[R_i],u}(w)\right]^2 && (2.8)
\end{aligned}$$

We now write out $\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_2^2 = \sum_{u\in R_i} q^{(i+1)}_{[R_i],s}(u)^2 = \sum_{u\in R_i}(\boldsymbol{q}^{(i)}_{[R_i],s}\cdot\boldsymbol{q}^{(i)}_{[R_i],u})^2$, by Prop. 2.3.1. We expand further below. The only inequality is Cauchy-Schwartz.

$$\begin{aligned}
\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_2^2 &= \sum_{u\in R_i}(\boldsymbol{q}^{(i)}_{[R_i],s}\cdot\boldsymbol{q}^{(i)}_{[R_i],u})\sum_{w\in R_i}q^{(i)}_{[R_i],s}(w)q^{(i)}_{[R_i],u}(w) && (2.9)\\
&= \sum_{w\in R_i}q^{(i)}_{[R_i],s}(w)\Big[\sum_{u\in R_i}(\boldsymbol{q}^{(i)}_{[R_i],s}\cdot\boldsymbol{q}^{(i)}_{[R_i],u})q^{(i)}_{[R_i],u}(w)\Big] && (2.10)\\
&\leq \sqrt{\sum_{w\in R_i}q^{(i)}_{[R_i],s}(w)^2}\sqrt{\sum_{w\in R_i}\Big[\sum_{u\in R_i}(\boldsymbol{q}^{(i)}_{[R_i],s}\cdot\boldsymbol{q}^{(i)}_{[R_i],u})q^{(i)}_{[R_i],u}(w)\Big]^2} && (2.11)\\
&= \|\boldsymbol{q}^{(i)}_{[R_i],s}\|_2\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1\sqrt{\mathbf{E}_{u_1,u_2\sim\mathcal{D}_{s_i}}[\boldsymbol{q}^{(i)}_{[R_i],u_1}\cdot\boldsymbol{q}^{(i)}_{[R_i],u_2}]} && (2.12)
\end{aligned}$$

The last step above follows by (2.8). We rearrange and take squares to complete the proof. ■

We can apply previous norm bounds to get an explicit lower bound. To see the significance of the following lemma, note that by Claim 2.3.2 and Cauchy-Schwartz, $\forall u_1,u_2\in R_i$, $\boldsymbol{q}^{(i)}_{[R_i],u_1}\cdot\boldsymbol{q}^{(i)}_{[R_i],u_2}\leq 1/n^{\delta(i-1)}$ (fairly close to the lower bound below).

Lemma 2.3.7 *Fix arbitrary $s\in S_i$.*

$$\mathbf{E}_{u_1,u_2\sim\mathcal{D}_{s,i}}[\boldsymbol{q}^{(i)}_{[R_i],u_1}\cdot\boldsymbol{q}^{(i)}_{[R_i],u_2}]\geq 1/n^{\delta(i+1)}$$

Proof By Lemma 2.3.6, the LHS is at least $\frac{1}{\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1^2} \cdot \frac{\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_2^4}{\|\boldsymbol{q}^{(i)}_{[R_i],s}\|_2^2}$. Note that $\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1 \leq 1$. By Definition 2.3.2, $\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_2^2 \geq 1/n^{\delta i}$. Since $s \in S_i \subseteq R_i$, by Claim 2.3.2, $\|\boldsymbol{q}^{(i)}_{[R_i],s}\|_2^2 \leq 1/n^{\delta(i-1)}$. ■

2.4 Analysis of `FindBiclique`

This is the central theorem of our analysis. It shows that the `FindBiclique`(s) procedure discovers a K_{r^2,r^2} minor with non-trivial probability when s is in a sufficiently high stratum.

Theorem 2.4.1 *Suppose $s \in S_i$, for $5r^2 \leq i \leq 1/\delta + 3$. The probability that the paths discovered in* `FindBiclique`(s) *contain a K_{r^2,r^2} minor is at least $n^{-4\delta r^4}$.*

Theorem 2.4.1 is proved in §2.4.5. Towards the proof, we will need multiple tools. In §2.4.1, we perform a standard calculation to bound the success probability of `FindPath`. In §2.4.2, we use this bound to show that the sets A and B sampled by `FindBiclique` are successfully connected by paths as discovered by `FindPath`. In §2.4.3, we argue that the intersections of these paths is "well-behaved" enough to induce a K_{r^2,r^2} minor.

We note that the $\sqrt{n}$ in the final running time comes from the calls to `FindPath` in `FindBiclique`.

2.4.1 The procedure `FindPath`

For convenience, we reproduce the procedure `FindPath`. It is a relatively straightforward application of a birthday paradox argument for bidirectional path finding.

`FindPath`(u, v, k, i)

1. Perform k random walks of length $2^i\ell$ from u and v.
2. If a walk from u and v terminate at the same vertex, return these paths.

Lemma 2.4.2 *Let c be a sufficiently large constant. Consider $u, v \in R_i$. Suppose there exist $\alpha \leq \beta$ such that* $\max(\|\boldsymbol{q}^{(i)}_{[R_i],u}\|_2^2, \|\boldsymbol{q}^{(i)}_{[R_i],v}\|_2^2) \leq 1/n^\alpha$ *and* $\boldsymbol{q}^{(i)}_{[R_i],u} \cdot \boldsymbol{q}^{(i)}_{[R_i],v} \geq 1/2n^\beta$. *Then, with $k \geq cn^{\beta/2+4(\beta-\alpha)}$,* `FindPath`$(u, v, k, i)$ *returns an R_i-returning path of length $2^{i+1}\ell$ with probability $\geq 2/3$.*

Proof First, define $W = \{w | q^{(i)}_{[R_i],u}(w)/q^{(i)}_{[R_i],v}(w) \in [1/(8n^{\beta-\alpha}), 8n^{\beta-\alpha}]\}$.

$$\sum_{w \notin W} q^{(i)}_{[R_i],u}(w) q^{(i)}_{[R_i],v}(w) \leq (8n^{\beta-\alpha})^{-1} \sum_{w \notin W} \max(q^{(i)}_{[R_i],u}(w), q^{(i)}_{[R_i],v}(w))^2$$
$$\leq (8n^{\beta-\alpha})^{-1}(\|\boldsymbol{q}^{(i)}_{[R_i],u}\|_2^2 + \|\boldsymbol{q}^{(i)}_{[R_i],v}\|_2^2) \leq 1/4n^\beta$$

Therefore, $\sum_{w \in W} q^{(i)}_{[R_i],u}(w) q^{(i)}_{[R_i],v}(w) \geq 1/2n^\beta$.

For $a, b \leq k$, let $X_{a,b}$ be the indicator for the following event: the ath $2^i\ell$-length random walk from u is an R_i-returning walk that ends at some $w \in W$, and the bth random walk from v is also R_i-returning, ending at the same w. Let $X = \sum_{a,b \leq k} X_{a,b}$. Observe that the probability that `FindPath`(u, v, k, i) returns a path is at least $\Pr[X > 0]$.

We bound $\mathbf{E}[\sum_{a,b \leq k} X_{a,b}] = k^2 \sum_{w \in W} q^{(i)}_{[R_i],u}(w) q^{(i)}_{[R_i],v}(w) \geq k^2/4n^\beta \geq (c^2/4)n^{4(\beta-\alpha)}$. Let us now bound the variance. First, let us expand out the expected square.

$$\mathbf{E}[(\sum_{a,b} X_{a,b})^2] = \Bigg(\sum_{a,b} \mathbf{E}[X_{a,b}^2] + 2\sum_{a \neq a', b} \mathbf{E}[X_{a,b}X_{a',b}] + 2\sum_{a, b \neq b'} \mathbf{E}[X_{a,b}X_{a',b}]$$
$$+2\sum_{a \neq a', b \neq b'} \mathbf{E}[X_{a,b}X_{a',b'}]\Bigg) \tag{2.13}$$

Observe that $X_{a,b}^2 = X_{a,b}$. Furthermore, for $a \neq a', b \neq b'$, by independence of the walks, $\mathbf{E}[X_{a,b}X_{a',b'}] = \mathbf{E}[X_{a,b}]\mathbf{E}[X_{a',b'}]$. (This term will cancel out in the variance.) By symmetry, $\sum_{a \neq a', b} \mathbf{E}[X_{a,b}X_{a',b}] \leq k^3 \mathbf{E}[X_{1,1}X_{2,1}]$ (and analogously for the third term in (2.13)). Plugging these in and expanding out the $\mathbf{E}[X]^2$,

$$\mathbf{var}[X] \leq \mathbf{E}[X] + 2k^3\mathbf{E}[X_{1,1}X_{2,1}] + 2k^3\mathbf{E}[X_{1,1}X_{1,2}]$$

Note that $X_{1,1}X_{2,1} = 1$ when the first and second walks from u end at the same vertex where the first walk from v ends. So, we see $\mathbf{E}[X_{1,1}X_{2,1}] = \sum_{w \in W} q^{(i)}_{[R_i],u}(w)^2 q^{(i)}_{[R_i],v}(w)$. Since $w \in W$, $q^{(i)}_{[R_i],v}(w) \leq 8n^{\beta-\alpha} q^{(i)}_{[R_i],u}(w)$. Plugging this bound in,

$$\begin{aligned}
2k^3\mathbf{E}[X_{1,1}X_{2,1}] &\le 16k^3 n^{\beta-\alpha}\sum_{w\in W} q^{(i)}_{[R_i],u}(w)^3 \\
&\le 16k^3 n^{\beta-\alpha}[\sum_{w\in W} q^{(i)}_{[R_i],u}(w)^2]^{3/2} \qquad (l_2\text{-}l_3 \text{ norm inequality}) \\
&= 16n^{\beta-\alpha}[k^2\sum_{w\in W} q^{(i)}_{[R_i],u}(w)^2]^{3/2}
\end{aligned}$$

We can apply the bound $q^{(i)}_{[R_i],u}(w) \le 8n^{\beta-\alpha} q^{(i)}_{[R_i],v}(w)$.

$$\begin{aligned}
2k^3\mathbf{E}[X_{1,1}X_{2,1}] &\le 16n^{\beta-\alpha}[k^2\sum_{w\in W} q^{(i)}_{[R_i],u}(w) q^{(i)}_{[R_i],v}(w)\cdot 8n^{\beta-\alpha}]^{3/2} \\
&\le 512 n^{5(\beta-\alpha)/2}[k^2\sum_{w\in W} q^{(i)}_{[R_i],u}(w) q^{(i)}_{[R_i],v}(w)]^{3/2} \qquad (2.14)
\end{aligned}$$

Note that $k^2\sum_{w\in W} q^{(i)}_{[R_i],u}(w) q^{(i)}_{[R_i],v}(w)]^{3/2}$ is exactly $\mathbf{E}[X]$. We have previously bounded $\mathbf{E}[X] \ge (c^2/4)n^{8(\beta-\alpha)}$. Thus, $512n^{5(\beta-\alpha)/2} \le \mathbf{E}[X]^{1/2}/(c/100)$. Applying the bounds in (2.14), we deduce that

$$2k^3\mathbf{E}[X_{1,1}X_{2,1}] \le (\mathbf{E}[X]^{1/2}/(c/100))(\mathbf{E}[X]^{3/2}) = \mathbf{E}[X]^2/(c/100).$$

We get an identical bound for $2k^3\mathbf{E}[X_{1,1}X_{1,2}]$. Putting it all together, we can prove that $\mathbf{var}[X] \le 4\mathbf{E}[X]^2/c'$, for $c' = \Theta(c)$. An application of Chebyshev proves that $\Pr[X > 0] > 2/3$.

■

2.4.2 The procedure `FindBiclique`

For convenience, we reproduce `FindBiclique`.

`FindBiclique`(s)

1. For $i = 5r^4, \ldots, 1/\delta + 4$:
   (a) Perform $2r^2$ independent random walks of length $2^{i+1}\ell$ from s. Let the destinations of the first r^2 walks be multiset A, and the destinations of the remaining walks be B.
   (b) For each $a \in A$, $b \in B$:
      i. Run `FindPath`$(a, b, n^{\delta(i+18)/2}, i)$
   (c) If all calls to `FindPath` return a path, then let the collection of paths be the subgraph F. Run `KKR`(F, H). If it returns an H-minor, output that and terminate.

Lemma 2.4.3 *Suppose $s \in S_i$, for some $i \leq 1/\delta + 4$. Condition on the event that $A, B \subseteq R_i$, during the ith iteration in* `FindBiclique`(s)*. With probability $(4n^{2\delta})^{-r^4}$, the calls to* `FindPath` *output paths from every $a \in A$ to every $b \in B$, where each path is an R_i-returning walk of length $2^{i+1}\ell$.*

Proof The probability that a $2^{i+1}\ell$-length random walk from s ends at u is at least $q^{(i+1)}_{[S_i],s}(u) = \hat{q}^{(i+1)}_{[R_i],s}(u)\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1$. In the rest of the proof, let $t = |A| = |B| = r^2$ denote the common size of the multisets A and B. For any $a, b \in V$, let $\tau_{a,b}$ be the probability that `FindPath`$(a, b, n^{\delta(i+18)/2}, i)$ succeeds in finding an R_i-returning walk between a and b (of length $2^{i+1}\ell$). The probability of success for `FindBiclique`(s) conditioned on $A, B \subseteq R_i$ is at least

$$
\begin{aligned}
&\sum_{A \in R_i^r} \sum_{B \in R_i^r} \prod_{a \in A} \hat{q}^{(i+1)}_{[R_i],s}(a) \prod_{b \in B} \hat{q}^{(i+1)}_{[R_i],s}(b) \tau_{a,b} \\
&= \sum_{A \in R_i^r} \sum_{B \in R_i^r} \prod_{a \in A} \hat{q}^{(i+1)}_{[R_i],s}(a) \Big(\prod_{b \in B} \hat{q}^{(i+1)}_{[R_i],s}(b) \Big) \Big(\prod_{b \in B} \tau_{a,b} \Big) \\
&= \sum_{B \in R_i^r} \Big(\prod_{b \in B} \hat{q}^{(i+1)}_{[R_i],s}(b) \Big) \sum_{A \in R_i^r} \prod_{a \in A} \Big[\hat{q}^{(i+1)}_{[R_i],s}(a) \Big(\prod_{b \in B} \tau_{a,b} \Big) \Big] \\
&= \sum_{B \in R_i^r} \prod_{b \in B} \hat{q}^{(i+1)}_{[R_i],s}(b) \Big(\sum_{a \in R_i} \hat{q}^{(i+1)}_{[R_i],s}(a) \prod_{b \in B} \tau_{a,b} \Big)^r.
\end{aligned}
$$

Observe that $\prod_{b\in B} \hat{q}^{(i+1)}_{[R_i],s}(b)$ is a probability distribution over R_i^r. By Jensen, we lower bound.

$$\sum_{B\in R_i^r}\prod_{b\in B}\hat{q}^{(i+1)}_{[R_i],s}(b)\Big(\sum_{a\in R_i}\hat{q}^{(i+1)}_{[R_i],s}(a)\prod_{b\in B}\tau_{a,b}\Big)^r \geq \Big[\sum_{B\in R_i^r}\Big(\prod_{b\in B}\hat{q}^{(i+1)}_{[R_i],s}(b)\Big)\sum_{a\in R_i}\hat{q}^{(i+1)}_{[R_i],s}(a)\prod_{b\in B}\tau_{a,b}\Big]^r$$

Note that we can write

$$\begin{aligned}\Big[\sum_{B\in R_i^r}\Big(\prod_{b\in B}\hat{q}^{(i+1)}_{[R_i],s}(b)\Big)\sum_{a\in R_i}\hat{q}^{(i+1)}_{[R_i],s}(a)\prod_{b\in B}\tau_{a,b}\Big]^r \\ = \Big[\sum_{a\in R_i}\sum_{B\in R_i^r}\hat{q}^{(i+1)}_{[R_i],s}(a)\Big(\prod_{b\in B}\hat{q}^{(i+1)}_{[R_i],s}(b)\Big)\Big(\prod_{b\in B}\tau_{a,b}\Big)\Big]^r\end{aligned}$$

Thus, we can manipulate the latter further.

$$\begin{aligned}\Big[\sum_{a\in R_i}\hat{q}^{(i+1)}_{[R_i],s}(a)\sum_{B\in R_i^r}\prod_{b\in B}\hat{q}^{(i+1)}_{[R_i],s}(b)\tau_{a,b}\Big]^r &= \Big[\sum_{a\in R_i}\hat{q}^{(i+1)}_{[R_i],s}(a)\big(\sum_{b\in R_i}\hat{q}^{(i+1)}_{[R_i],s}(b)\tau_{a,b}\big)^r\Big]^r \\ &\geq \Big[\sum_{a\in R_i}\sum_{b\in R_i}\hat{q}^{(i+1)}_{[R_i],s}(a)\hat{q}^{(i+1)}_{[R_i],s}(b)\tau_{a,b}\Big]^{r^2} \\ &= \Big[\mathbf{E}_{a,b\sim\mathcal{D}_{s,i}}[\tau_{a,b}]\Big]^{r^2}. \end{aligned} \tag{2.15}$$

where the inequality follows by Jensen. Towards lower bounding $\tau_{a,b}$, we first lower bound $\boldsymbol{q}^{(i)}_{[R_i],a}\cdot\boldsymbol{q}^{(i)}_{[R_i],b}$. By Lemma 2.3.7, $\mathbf{E}_{a,b}[\boldsymbol{q}^{(i)}_{[R_i],a}\cdot\boldsymbol{q}^{(i)}_{[R_i],b}] \geq 1/n^{\delta(i+1)}$. Applying Cauchy-Schwartz, $\boldsymbol{q}^{(i)}_{[R_i],a}\cdot\boldsymbol{q}^{(i)}_{[R_i],b} \leq 1/n^{\delta(i-1)}$. Let p be the probability (over a,b) that $\boldsymbol{q}^{(i)}_{[R_i],a}\cdot\boldsymbol{q}^{(i)}_{[R_i],b} \geq 1/2n^{\delta(i+1)}$.

$$1/n^{\delta(i+1)} \leq \mathbf{E}_{a,b}[\boldsymbol{q}^{(i)}_{[R_i],a}\cdot\boldsymbol{q}^{(i)}_{[R_i],b}] \leq (1-p)/2n^{\delta(i+1)} + p/n^{\delta(i-1)}$$

Thus, $p \geq 1/2n^{2\delta}$.

By Claim 2.3.2, for every $a\in R_i$, $\|\boldsymbol{q}^{(i)}_{[R_i],a}\|_2^2 \leq 1/n^{\delta(i-1)}$ (similarly for $b\in R_i$). Suppose $\boldsymbol{q}^{(i)}_{[R_i],a}\cdot\boldsymbol{q}^{(i)}_{[R_i],b} \geq 1/2n^{\delta(i+1)}$. Let us apply Lemma 2.4.2, with $\alpha=\delta(i-1)$ and $\beta=\delta(i+1)$. The number of paths taken in `FindPath` (the value k) is $n^{\delta(i+18)/2}$. Note that $\delta(i+18)/2 > \delta(i+1)/2+8\delta = \beta/2+4(\alpha-\beta)$. By Lemma 2.4.2, in this case, $\tau \geq 1/2$. As argued in the previous paragraph, this will happen with probability $1/2n^{2\delta}$ (over the choice of $a,b\sim\mathcal{D}_{s,i}$). We plug in (2.15) and deduce that the probability of success is at least $(1/4n^{2\delta})^{r^4}$. ∎

2.4.3 Criteria for `FindBiclique` to reveal a minor

Fix $s \in S_i$, as in Lemma 2.4.3. This lemma only asserts that all pairs in $A \times B$ are connected by `FindBiclique` (with non-trivial probability). We need to argue that these paths will actually induce a K_{r^2,r^2}-minor.

As in Lemma 2.4.3, let us focus on the ith iteration within `FindBiclique`, and condition on $A, B \in R_i$. For every $a \in A, b \in B$, we call `FindPath`$(a, b, n^{\delta(i+18)/2}, i)$. Within each such call, a set of walks is performed from both a and b, with the hope of connecting a to b. We use a, a' (resp. b, b') to refer to elements in A (resp. B).

- Let us use $\boldsymbol{W}_a^b$ to refer to the set of walks from a performed in the call to `FindPath`$(a, b, n^{\delta(i+18)/2}, i)$ that are R_i-returning. We stress that these walks do not necessarily end at b, and come from a distribution independent of b (but we wish to track the specific call of `FindPath` where these walks were performed). Note that $\boldsymbol{W}_b^a$ is the set of R_i-returning walks starting from b performed in the same call.

We use $\boldsymbol{W}_a$ to denote the set of all vertices in $\bigcup_{b \in B} \boldsymbol{W}_a^b$.

- Let $P_{a,b}$ be a single path from a to b discovered by `FindPath`$(a, b, n^{\delta(i+18)/2}, i)$, that consists of a walk in $\boldsymbol{W}_a^b$ and a walk $\boldsymbol{W}_b^a$ that end at the same vertex. If there are many possible such paths, pick the lexicographically least.

Note that any of the paths/sets described above could be empty. We will think of paths as sequences, rather than sets, since the order in which the path is constructed is relevant. For any path, P, we use $P(t)$ to denote the tth element in the sequence. We use $P(\geq t)$ to denote the sequence of elements with index at least t. When we refer to intersections of paths being empty/non-empty, we refer to sets induced by the corresponding sequences.

For $s \in S_i$, conditioned on $A, B \subseteq R_i$, Lemma 2.4.3 gives a lower bound on $\Pr[\bigcap_{a \in A, b \in B} P_{a,b} \neq \emptyset]$. We will define some *bad* events that interfere with minor structure.

Recall that A and B are multisets (it is convenient to think of them as sequences). The same vertex may appear multiple times in $A \cup B$, but we think of each occurrence as a distinct multiset element. Therefore, equality refers to vertex at the same index in A (or B). By definition, elements in A are disjoint from B.

Definition 2.4.1 *The following events are referred to as* bad events *of Type 1, 2, or 3. We set* $\tau = 2^{i-1}\ell$.

1. $\exists a, b, c \in A \cup B$, $c \neq a, b$, *such that* $\boldsymbol{W}_c \cap P_{a,b} \neq \emptyset$.

2. $\exists a, b, b'$ *(all distinct) such that* $\exists W \in \boldsymbol{W}_a^b$ *where* $W(\geq \tau) \cap P_{a,b'} \neq \emptyset$. *(Or,* $\exists a, a' \in A, b \in B$, *all distinct, such that* $\exists W \in \boldsymbol{W}_b^a$ *where* $W(\geq \tau) \cap P_{a',b} \neq \emptyset$.*)*

3. $\exists a, b, W_a \in \boldsymbol{W}_a^b, W_b \in \boldsymbol{W}_b^a$ *such that* W_a, W_b *end at the same vertex and* $\exists t_1, t_2$ *such that* $\min(t_1, t_2) \leq \tau$ *and* $W_a(t_1) = W_b(t_2)$.

For clarity, let us express the above bad events in plain English. Note that τ is the index of the midpoint of the walks, so it splits walks into halves.

1. A walk from $c \in A \cup B$ intersects $P_{a,b}$, where $c \neq a, b$.
2. The second half of a walk in $\boldsymbol{W}_a^b$ (which starts from a) intersects $P_{a,b'}$ for $b \neq b'$.
3. A walk in $\boldsymbol{W}_a^b$ and a walk in $\boldsymbol{W}_b^a$ intersect twice. Note that this is a pair of walks, one from a and the other from b. The first intersection is in the first half of either of the walks. The walks also end at the same vertex.

Claim 2.4.4 *If all* $P_{a,b}$ *sets are non-empty and there is no bad event, then* $\bigcup_{a,b} P_{a,b}$ *contains a* K_{r^2,r^2}*-minor.*

Proof The $P_{a,b}$'s may not form simple paths, and it will be convenient to "clean them up". Each $P_{a,b}$ is formed by $W_a \in \boldsymbol{W}_a^b$ and $W_b \in \boldsymbol{W}_b^a$ that end at the same vertex. Since there is no Type 3 bad event, $W_a(\leq \tau)$ is disjoint from W_b (and vice versa). Therefore (by removing self-intersections and loops), we can construct a simple path from a to b with the following (vertex) disjoint contiguous simple paths: $Q_{a,b} \subseteq W_a(\leq \tau)$, $\widehat{P_{a,b}} \subseteq W_a(\geq \tau) \cup W_b(\geq \tau)$, and $Q_{b,a} \subseteq W_b(\leq \tau)$.

In each bullet below, we first make a statement about the disjointness of these various sets. The proof follows immediately. We consider $a, a' \in A$ and $b, b' \in B$, where the elements in A (or B) might be equal.

- If $a \neq a'$, $Q_{a,b} \cap Q_{a',b'} = \emptyset$. If $b \neq b'$, $Q_{b,a} \cap Q_{b',a'} = \emptyset$.

Consider the first statement. (Note that we allow $b = b'$.) Observe that $Q_{a,b} \subseteq \boldsymbol{W}_a$ and $Q_{a',b'} \subseteq P_{a',b'}$. So $\boldsymbol{W}_a \cap P_{a',b'} \neq \emptyset$, implying a Type 1 bad event. The second statement has an analogous proof.

- $Q_{a,b} \cap Q_{b',a'} = \emptyset$.

If $a = a', b = b'$, then this holds by the argument in the first paragraph (no Type 3 bad events). Suppose $a \neq a'$. Then (as before), $Q_{a,b} \subseteq \boldsymbol{W}_a$ and $Q_{b',a'} \subseteq P_{a',b'}$. Since no Type 1 bad events occur, $\boldsymbol{W}_a \cap P_{a',b'} = \emptyset$. The case $b \neq b'$ is analogous.

- If $a \neq a'$ or $b \neq b'$, $\widehat{P_{a,b}} \cap P_{a',b'} = \emptyset$.

Wlog, assume $a \neq a'$. Note that $\widehat{P_{a,b}} \subseteq W_a(\geq \tau) \cup W_b(\geq \tau)$, where $W_a \in \boldsymbol{W}_a^b$ and $W_b \in \boldsymbol{W}_b^a$. If $W_a(\geq \tau) \cap P_{a',b'} \neq \emptyset$, then $\boldsymbol{W}_a \cap P_{a',b'} \neq \emptyset$ (a Type 1 bad event). Suppose $W_b(\geq \tau) \cap P_{a',b'} \neq \emptyset$. If $b \neq b'$, this is Type 1 bad event. So suppose $b = b'$, so $W_b(\geq \tau) \cap P_{a',b} \neq \emptyset$. Since $W_b \in \boldsymbol{W}_b^a$ (for $a \neq a'$), this is Type 2 bad event.

We construct the minor. Let $C(a) = \bigcup_{b \in B} Q_{a,b}$ and $C(b) = \bigcup_{a \in A} Q_{b,a}$. Each $C(a), C(b)$ forms a connected subgraph. By the disjointness properties of the $Q_{a,b}$ sets, all the $C(a), C(b)$ sets/subgraphs are vertex disjoint. Note that $\widehat{P_{a,b}}$ is disjoint from all other $P_{a',b'}$ paths *and* all the $C(a), C(b)$ sets. (We construct $P_{a,b}$ to be disjoint from $Q_{a,b}$ and $Q_{b,a}$ in the first paragraph. Every other $Q_{a',b'}$ is contained in $P_{a',b'}$.) Thus, we have disjoint paths from each $C(a)$ to $C(b)$, which gives a K_{r^2,r^2}-minor. ■

2.4.4 The probabilities of bad events

In this section, we bound the probability of bad events, as detailed in Definition 2.4.1. As before, we fix $s \in S_i$ and condition on $A \cup B \subseteq R_i$.

We require some technical definitions of random walk probabilities.

Definition 2.4.2 *Let $\sigma_{s,S,t}(v)$ be the probability of a walk from s to v of length t being S-returning. (We allow $\ell \nmid t$, and require that the walk encounters S at every $j\ell$th step, for $j \leq \lfloor t/\ell \rfloor$.)*

We use $\boldsymbol{\sigma}_{s,S,t}$ to denote the vector of these probabilities. More generally, given any distribution vector $\boldsymbol{x}$ on V, $\boldsymbol{\sigma}_{\boldsymbol{x},S,t}$ denotes the vector of S-returning walk probabilities at time t.

We stress that this is not a conditional probability. Note that if $t = 2^i\ell$, then $\boldsymbol{\sigma}_{s,S,t} = \boldsymbol{q}^{(i)}_{[S],s}$. We show some simple propositions on these vectors. Let $\mathbb{I}_S$ denote the $n \times n$ matrix that preserves all coordinates in S and zeroes out other coordinates.

Proposition 2.4.1 *The vector $\boldsymbol{\sigma}_{\boldsymbol{x},S,t}$ evolves according to the following recurrence. Firstly, $\boldsymbol{\sigma}_{\boldsymbol{x},S,0} = \boldsymbol{x}$. For $t \geq 1$ such that $\ell \nmid t$, $\boldsymbol{\sigma}_{\boldsymbol{x},S,t} = M\boldsymbol{\sigma}_{\boldsymbol{x},S,t-1}$. For $t \geq 1$ such that $\ell \mid t$, $\boldsymbol{\sigma}_{\boldsymbol{x},S,t} = \mathbb{I}_S M\boldsymbol{\sigma}_{\boldsymbol{x},S,t-1}$*

Proposition 2.4.2 *For all $\boldsymbol{x}$ and all $t \geq 1$, $\|\boldsymbol{\sigma}_{\boldsymbol{x},S,t}\|_\infty \leq \|\boldsymbol{\sigma}_{\boldsymbol{x},S,t-1}\|_\infty$.*

Proof Since M is a symmetric random walk matrix, it computes the "new" value at a vertex by averaging the values of the neighbors (and itself). This can never increase the maximum value. Furthermore, $\mathbb{I}_S$ only zeroes out some coordinates. This proves the proposition. ■

In what follows, we fix the walk length to $2^i\ell$. To reduce clutter, we drop notational dependencies on this length.

Definition 2.4.3 *The distribution of $2^i\ell$-length walks from u is denoted $\mathcal{W}_u$. For any walk W, $W_u(t)$ denotes the tth vertex of the walk.*

The Boolean predicate $\rho(W_u)$ is true if W_u is R_i-returning.

Recall that $\mathcal{D}_{s,i}$ is the distribution with support R_i, where the probability of $u \in R_i$ is $\hat{q}^{(i+1)}_{[R_i],s}(v)/\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1$ (Definition 2.3.3). Conditioned on $a \in R_i$, this is precisely the distribution that the elements of the sets A and B are drawn from. Refer to `FindBiclique`, where $A \cup B$ are the destinations of $2^{i+1}\ell$-length random walks from s. Since i is fixed, we will simply write this as $\mathcal{D}_s$.

Claim 2.4.5 *For any $F \subseteq V$:*

1.

$$\Pr_{a\sim\mathcal{D}_s, W_a\sim\mathcal{W}_a}[\rho(W_a)\wedge W_a\cap F\neq\emptyset]\leq 2^i\ell|F|/(n^{\delta(i-1)}\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1).$$

2. For any $a\in R_i$,

$$\Pr_{W_a\sim\mathcal{W}_a}[\exists t\geq\tau \mid \rho(W_a)\wedge W_a(t)\in F]\leq 2^i\ell|F|/n^{\delta(i-2)}$$

Proof We prove the first part. Let $\boldsymbol{x}$ be the probability vector corresponding to $\mathcal{D}_s$. So $\|\boldsymbol{x}\|_\infty=\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_\infty/\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1$. By Prop. 2.4.2, $\forall t\geq 1$, $\|\boldsymbol{\sigma}_{\boldsymbol{x},R_i,t}\|_\infty\leq\|\boldsymbol{x}\|_\infty$. sing Claim 2.3.3, this is at most $1/(n^{\delta(i-1)}\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1)$. We union bound over F and the walk length.

$$\begin{aligned}\Pr_{a\sim\mathcal{D}_s,W_a\sim\mathcal{W}_a}[\rho(W_a)\wedge W_a\cap F\neq\emptyset]&\leq\sum_{t\leq 2^i\ell}\sum_{v\in F}\Pr_{a\sim\mathcal{D}_s,W_a\sim\mathcal{W}_a}[\rho(W_a)\wedge W_a(t)=v]\\&\leq\sum_{t\leq 2^i\ell}\sum_{v\in F}\|\boldsymbol{x}\|_\infty\leq 2^i\ell|F|/(n^{\delta(i-1)}\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1)\end{aligned}$$

Now for the second part. By the union bound, the probability is bounded above by

$$\sum_{t\geq 2^{i-1}\ell}\sum_{u\in F}\Pr_{W_a\sim\mathcal{W}_a}[\rho(W_a)\wedge W_a(t)=u]\leq\sum_{t\geq 2^{i-1}\ell}\sum_{u\in F}\|\boldsymbol{\sigma}_{a,R_i,t}\|_\infty \tag{2.16}$$

By Prop. 2.4.2, the infinity norm is bounded above by $\|\boldsymbol{\sigma}_{a,R_i,2^{i-1}\ell}\|_\infty=\|\boldsymbol{q}^{(i-1)}_{[R_i],a}\|_\infty$. By Claim 2.3.3, the latter is at most $1/n^{\delta(i-2)}$. Plugging in (2.16), we get an upper bound of $2^{i-1}\ell|F|/n^{\delta(i-2)}$. ■

Claim 2.4.6 *For any $a\in R_i$,*

$$\begin{aligned}\Pr\big[\rho(W_a)\wedge\rho(W_b)\wedge W_a(2^i\ell)=W_b(2^i\ell)\wedge&\\ \exists t_a,t_b,\min(t_a,t_b)\leq\tau,W_a(t_a)=W_b(t_b))\big]&\leq\frac{2^{2i}\ell^2}{(n^{\delta(2i-2)}\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1)}\end{aligned}$$

Here, the probability is taken over the following joint distribution: $b\sim\mathcal{D}_s, W_a\sim\mathcal{W}_a, W_b\sim\mathcal{W}_b$

Proof Let us write out the main event in English. We fix an arbitrary a, and pick $b\sim\mathcal{D}_s$. We perform R_i-returning walks of length $2^i\ell$ from both a and b. We are bounding the probability that the "initial half" (less than $2^{i-1}\ell$ steps) of one of the walks intersects with the other, and subsequently, both walks end at the same vertex.

To that end, let us define two vertices w_1, w_2. We want to bound the probability of that both walks first encounter w_1, and then end at w_2. It is be very useful to treat the latter part simply as two walks from w_1, where one of them is at least of length $2^{i-1}\ell$. Note that w_1 might not be in R_i.

Let $Z_{a,t}$ be the random variable denoting the tth vertex of a random walk from a. Let us also define R_i-returning walks with an offset g, starting from w. Basically, such a walk starts from w (that may not be in R_i) and performs g steps to end up in R_i. Subsequently, it behaves as an R_i-returning walk. Observe that the second parts of the walks are R_i-returning walks from w_1, with offsets of $\ell - [t_a(\text{mod } \ell)]$, $\ell - [t_b(\text{mod } \ell)]$. Let $Y_{w,t}$ be the random variable denoting the tth vertex of an R_i-returning walk from w, with the offset $\ell - [t(\text{mod } \ell)]$. We use primed versions for independent such variables.

Let us fix values for t_a, t_b such that $\min(t_a, t_b) \leq \tau = 2^{i-1}\ell$. (We will eventually union bound over all such values.) The probability we wish to bound is the following. We use independence of the walks to split the probabilities. There are four independent walks under consideration: one from a, one from b, and two from w.

$$\sum_{w_1 \in V} \sum_{w_2 \in V} \Pr_{\substack{b \sim \mathcal{D}_s \\ \mathcal{W}_a, \mathcal{W}_b, \mathcal{W}_{w_1}}} [Z_{a,t_a} = w_1 \wedge Z_{b,t_b} = w_1 \wedge Y_{w_1, 2^i\ell - t_a} = w_2 \wedge Y'_{w_1, 2^i\ell - t_b} = w_2]$$
$$= \sum_{w_1 \in V} \sum_{w_2 \in V} \Pr_{\mathcal{W}_a}[Z_{a,t_a} = w_1] \Pr_{\substack{b \sim \mathcal{D}_s \\ \mathcal{W}_b}}[Z_{b,t_b} = w_1] \Pr_{\mathcal{W}_{w_1}}[Y_{w_1, 2^i\ell - t_a} = w_2] \Pr_{\mathcal{W}_{w_1}}[Y_{w_1, 2^i\ell - t_b} = w_2] \tag{2.17}$$

Consider $\Pr_{b \sim \mathcal{D}_s, \mathcal{W}_b}[Z_{b,t_b} = w_1]$. This is exactly the w_1th entry in $\boldsymbol{\sigma}_{\boldsymbol{x}, \mathbb{R}_i, t_b}$ where $\boldsymbol{x}$ is the distribution given by $\mathcal{D}_s$. By Prop. 2.4.2, this is at most $\|\boldsymbol{x}\|_\infty$, which is at most $1/(n^{\delta(i-1)} \|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1)$ (as argued in the second pat of the proof of Claim 2.4.5).

Since $\min(t_a, t_b) \leq \tau$, at least one of $2^i\ell - t_a$ or $2^i\ell - t_b$ is at least $2^{i-1}\ell$. Thus, one of $\Pr_{\mathcal{W}_{w_1}}[Y_{w_1, 2^i\ell - t_a} = w_2]$ or $\Pr_{\mathcal{W}_{w_1}}[Y_{w_1, 2^i\ell - t_b} = w_2]$ refers to a walk of length at least $2^{i-1}\ell$. Let us bound $\Pr_{\mathcal{W}_{w_1}}[Y_{w_1,t} = w_2]$ for $t \geq 2^i\ell$. We can break such a walk into two parts: the first $\ell - [t(\text{mod } \ell)]$ steps lead to some $v \in R_i$, and the second part is an R_i-returning walk of length at least $2^i\ell$ from v to w. Recall that $p_{x,d}(y)$ is the standard random walk probability of starting from x and ending at y after d steps. For some $t' \geq 2^i\ell$,

$$\begin{aligned}
\Pr_{\mathcal{W}_{w_1}}[Y_{w_1,t} = w_2] &= \sum_{v \in R_i} p_{w_1,\ell-[t(\text{mod } \ell)]}(v)\sigma_{v,R_i,t'}(w_2) \\
&\leq \sum_{v \in R_i} p_{w_1,\ell-[t(\text{mod } \ell)]}(v)\|\boldsymbol{\sigma}_{v,R_i,t'}\|_\infty \\
&\leq \sum_{v \in R_i} p_{w_1,\ell-[t(\text{mod } \ell)]}(v)\|\boldsymbol{q}^{(i)}_{[R_i],v}\|_\infty \\
&\leq \sum_{v \in R_i} p_{w_1,\ell-[t(\text{mod } \ell)]}(v)n^{-\delta(i-1)} \\
&= n^{-\delta(i-1)}.
\end{aligned}$$

Plugging these bounds in (2.17), for fixed t_a, t_b, there exists $t \in \{2^i\ell - t_a, 2^i\ell - t_b\}$ such that the probability of the main event is at most

$$\begin{aligned}
&\frac{1}{n^{\delta(i-1)}\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1} \cdot \frac{1}{n^{\delta(i-1)}} \sum_{w_1 \in V} \sum_{w_2 \in V} \Pr_{\mathcal{W}_a}[Z_{a,t_a} = w_1] \Pr_{\mathcal{W}_{w_1}}[Y_{w_1,t} = w_2] \\
&\leq \frac{1}{n^{\delta(2i-2)}\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1} \sum_{w_1 \in V} \Pr_{\mathcal{W}_a}[Z_{a,t_a} = w_1] \sum_{w_2 \in V} \Pr_{\mathcal{W}_{w_1}}[Y_{w_1,t} = w_2] = \frac{1}{n^{\delta(2i-2)}\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1}
\end{aligned}$$

A union bound over all pairs of t_a, t_b completes the proof. ■

We now bound the total probability of bad events. Most of the technical work is already done in the previous lemmas; we only need to perform some union bounds.

Lemma 2.4.7 *Conditioned on $A \cup B \subseteq R_i$, the total probability of bad events is at most*

$$\frac{2^{2i+4}r^8n^{30\delta}}{n^{\delta i/2}\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1} \tag{2.18}$$

Proof We bound the bad events by type. Recall that $\ell = n^{5\delta}$.

Type 1: $\exists a, b, c \in A \cup B$, $c \neq a, b$, such that $\boldsymbol{W}_c \cap P_{a,b} \neq \emptyset$.

Fix a choice of $a \in A, b \in B$. Conditioned in $A \cup B \subseteq R_i$, any $c \neq a, b$ is drawn from $\mathcal{D}_s$. In Claim 2.4.5, set $F = P_{a,b}$. By the first part of Claim 2.4.5, the probability that a single walk drawn from $\mathcal{W}_c$ is R_i-returning and intersects $P_{a,b}$ is at most $2^i\ell(2^{i+1}\ell)/n^{\delta(i-1)}\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1$. The set $\boldsymbol{W}_c$ consists of at most $r^2 n^{\delta(i+18)/2}$ such walks. We union bound over all these walks, and all r^4 choices of a, b, and plug in $\ell = n^{5\delta}$ to get an upper bound of

$$\frac{2^{2i+1}\ell^2 r^6 n^{\delta(i+18)/2}}{n^{\delta(i-1)}\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1} = \frac{2^{2i+1} r^6 n^{20\delta}}{n^{\delta i/2}\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1}$$

Type 2: $\exists a, b, b'$ (all distinct) such that $\exists W \in \boldsymbol{W}^b_a$ where $W(\geq \tau) \cap P_{a,b'} \neq \emptyset$. (Or, $\exists a, a' \in A, b \in B$ with analogous conditions.)

Fix a, b, b'. Set $F = P_{a,b'}$ in Claim 2.4.5. By the second part of Claim 2.4.5, the probability that a single walk from $\mathcal{W}_a$ is R_i-returning and intersects F at step $\geq \tau$ is at most $2^i\ell(2^{i+1}\ell)/n^{\delta(i-2)}$. We union bound over all the $r^2 n^{\delta(i+18)/2}$ walks in $\boldsymbol{W}_a$ and all r^6 choices of a, b, b'. (We also union bound over choosing b, b' or a, a'.) The upper bound is $2^{2i+1} r^6 n^{21\delta}/n^{\delta i/2}$.

Type 3: $\exists a, b, W_a \in \boldsymbol{W}^b_a, W_b \in \boldsymbol{W}^a_b$ such that W_a, W_b end at the same vertex and $\exists t_1, t_2$ such that $\min(t_1, t_2) \leq \tau$ and $W_a(t_1) = W_b(t_2)$.

This case is qualitatively different. We will take a union bound over *pairs* of walks, and require the stronger bound of Claim 2.4.6.

Fix $a \in A$. Observe that $b \sim \mathcal{D}_s$. For a single walk $W_a \sim \mathcal{W}_a$ and a single walk $W_b \sim \mathcal{W}_b$, the probability of a Type 3 bad event is bounded by Claim 2.4.6. The upper bound is $2^{2i}\ell^2/(n^{\delta(2i-2)}\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1)$. We union bound over the $r^4 n^{\delta(i+18)}$ *pairs* of walks from a and b, and then over the r^4 choices of a, b. The final bound is:

$$\frac{2^{2i} r^4 \ell^2 n^{\delta(i+18)}}{n^{\delta(2i-2)}\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1} = \frac{2^{2i} r^4 n^{30\delta}}{n^{\delta i}\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1}$$

We complete the proof by taking a union bound over the three types. Note that $\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1 \leq 1$, so we can upper bound the probability of each type of bad event by $\frac{2^{2i+1} r^8 n^{30\delta}}{n^{\delta i/2}\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1}$. ∎

2.4.5 Proof of Theorem 2.4.1

Proof Fix $s \in S_i$. Let $\mathcal{C}$ be the event that $A \cup B \subseteq R_i$, let $\mathcal{E}$ be the event $\bigcap_{a\in A, b\in B} P_{a,b} \neq \emptyset$, and let $\mathcal{F}$ be the union of bad events. By Claim 2.4.4, the probability that `FindBiclique`(s) find a minor is at least $\Pr[\mathcal{E} \cap \overline{\mathcal{F}}]$. We lower bound as follows: $\Pr[\mathcal{E} \cap \overline{\mathcal{F}}] \geq \Pr[\mathcal{C}] \Pr[\mathcal{E} \cap \overline{\mathcal{F}}|\mathcal{C}] \geq \Pr[\mathcal{C}](\Pr[\mathcal{E}|\mathcal{C}] - \Pr[\mathcal{F}|\mathcal{C}])$.

Note that $\Pr[\mathcal{C}] = \|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1^{2r^2}$. By Claim 2.3.4, $\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1 \geq n^{-\delta}$, so $\Pr[\mathcal{C}] \geq n^{-2\delta r^2}$. Lemma 2.4.3 provides a lower bound for $\Pr[\mathcal{E}|\mathcal{C}]$, and Lemma 2.4.7 provides an upper bound for $\Pr[\mathcal{F}|\mathcal{C}]$. We plug these bounds in below.

$$\Pr[\mathcal{E}|\mathcal{C}] - \Pr[\mathcal{F}|\mathcal{C}] \;\; \geq \;\; \frac{1}{(4n^{2\delta})^{r^4}} - \frac{2^{2i+4} r^8 n^{30\delta}}{n^{\delta i/2}\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1} \tag{2.19}$$

Observe how the positive term is independent of i, while the negative term decays exponentially in i. This is crucial to argue that for a sufficiently large (constant) i, the lower bound is non-trivial.

When $i \geq 5r^4$, $n^{i\delta/2} \geq n^{2\delta r^4 + \delta r^4/2} \geq n^{2\delta r^4 + 40\delta}$ (note that, r, the number of vertices in H, is at least 3). By Claim 2.3.4, $\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1 \geq n^{-\delta}$. Thus, for sufficiently large n, $\Pr[\mathcal{F}|\mathcal{C}] \leq 1/(2(4n^{2\delta})^{r^4})$. Putting it all together, the probability of finding a K_{r^2,r_2}-minor is at least $n^{-4\delta r^4}$. ■

2.5 Local partitioning in the trapped case

Theorem 2.4.1 tells us that if there are $\Omega(n^{1-\delta})$ vertices in strata numbered $5r^4$ and above, then `FindMinor` finds a biclique minor with high probability. We deal with the case when most vertices lie in low strata, i.e, random walks from most vertices are trapped in a very small subset.

We will argue that (almost) all vertices in low strata can be partitioned into "pieces", such that each piece is a low conductance cut, and (a superset of) each piece can be found by performing random walks in G. If `FindMinor` fails to find a minor, this lemma can be iteratively applied to make G H-minor-free by removing few edges (this argument is given in §2.6).

We use $p_{s,t}(v)$ to denote the probability that at t length random walk from s ends at v.

Lemma 2.5.1 *Let $\alpha \geq n^{-\delta/2}$. Consider some subset $S \subseteq V$ and $i \in \mathbb{N}$ such that $\forall s \in S, \|\boldsymbol{q}^{(i)}_{[S],s}\|_2^2 \leq 1/n^{\delta(i-1)}$. Define $S' \subseteq S$ to be $\{s | s \in S$ and $\|\boldsymbol{q}^{(i+1)}_{[S],s}\|_2^2 \geq 1/n^{\delta i}\}$.*

Suppose $|S'| \geq \alpha n$. Then, there is a subset $\widetilde{S} \subseteq S'$, $|\widetilde{S}| \geq \alpha n/8$ such that for $\forall s \in \widetilde{S}$: there exists a subset $P_s \subseteq S$ where

- *$E(P_s, S \setminus P_s) \leq 2n^{-\delta/4} d|P_s|$*
- *$\forall v \in P_s$, $\exists t \leq 160 n^{\delta(i+7)}/\alpha$ such that $p_{s,t}(v) \geq \alpha/n^{\delta(2i+14)}$.*

The aim of this section is to prove this lemma. Henceforth, we will assume that S, S' are as defined in the lemma.

Using the norm bounds, we show that for every vertex $s \in S'$, there is a large set of destination vertices that are all reached with high probability through random walks of length $2^{i+1}\ell$.

Claim 2.5.2 *For every $s \in S'$, there exists a set $U_s \subseteq S$, $|U_s| \geq n^{\delta(i-2)}$, such that $\forall u \in U_s$, $p_{s,2^{i+1}\ell}(u) \geq 1/2n^{\delta i}$.*

Proof By Prop. 2.3.1, for any $u \in U$, $q^{(i+1)}_{[S],s}(u) = \boldsymbol{q}^{(i)}_{[S],s} \cdot \boldsymbol{q}^{(i)}_{[S],u}$. By the property of S and Cauchy-Schwartz, $q^{(i+1)}_{[S],s}(u) \leq 1/n^{\delta(i-1)}$.

Since $s \in S'$, $\sum_{u \in S} q^{(i+1)}_{[S],s}(u)^2 \geq 1/n^{\delta i}$. Let us simply define U_s to be $\{u | u \in S, q^{(i+1)}_{[S],s}(u) \geq 1/2n^{\delta i}\}$. Note that $p_{s,2^{i+1}\ell}(u) \geq q^{(i+1)}_{[S],s}(u)$.

$$\begin{aligned} 1/n^{\delta i} \leq \sum_{u \in S} q^{(i+1)}_{[S],s}(u)^2 &= \sum_{u \in U_s} q^{(i+1)}_{[S],s}(u)^2 + \sum_{u \notin U_s} q^{(i+1)}_{[S],s}(u)^2 \\ &\leq |U_s|/n^{2\delta(i-1)} + (1/2n^{\delta i}) \sum_{u \notin U_s} q^{(i+1)}_{[S],s}(u) \\ &\leq |U_s|/n^{2\delta(i-1)} + 1/2n^{\delta i} \end{aligned}$$

We rearrange to bound the size of U_s. ■

2.5.1 Local partitioning on the projected Markov chain

We define the "projection" of the random walk onto the set S. This uses a construction of [50]. We define a Markov chain M_S over the set S. We retain all transitions from the original random walk on G that are within S, and we denote these by $e^{(1)}_{u,v}$ for every u to v transition in the random walk on G. Additionally, for every $u, v \in S$ and $t \geq 2$, we add a transition $e^{(t)}_{u,v}$. The probability of this transition is equal to the total probability of t-length walks in G from u to v, where all internal vertices in the walk lie outside S.

Since G is irreducible and the stationary mass on S is non-zero, all walks eventually reach S. Thus the outgoing transition probabilities from each v in M_S sum to 1, and hence M_S is a valid Markov chain. Furthermore, by the symmetry of the original random walk, $e^{(t)}_{u,v} = e^{(t)}_{v,u}$. Therefore the transition matrix of M_S remains symmetric, and the stationary distribution is uniform on S.

For a transition $e^{(t)}_{u,v}$ in M_S, we define the length of this transition to be t. For clarity, we use "hops" to denote the length of a walk in M_S, and retain "length" for walks in G. The length of an h hop random walk in M_S is defined to be the sum of the lengths of the transitions it takes. We note that these ideas come from the work of Kale-Peres-Seshadhri to analyze random walks in noisy expanders [50].

We use $\boldsymbol{\tau}_{s,h}$ to denote the distribution of the h-hop walk from s, and $\tau_{s,h}(v)$ to denote the corresponding probability of reaching v. We use $\mathcal{W}_h$ to denote the distribution of h-hop walks starting from the uniform distribution in S.

We state Kac's formula (Corollary 24 in Chapter 2 of [58], restated).

Lemma 2.5.3 *(Kac's formula) The expected return time (in G) to S of a random walk starting from S is reciprocal of the fractional stationary mass of S, ie $n/|S|$.*

The following is a direct corollary.

Lemma 2.5.4 $\mathbf{E}_{W \sim \mathcal{W}_h}[\textit{length of } W] = hn/|S|$

Proof Since the walk starts at the stationary distribution, it remains in this distribution at all hops. By linearity of expectation, it suffices to get the expected length for the first hop (and multiply with h). This is precisely expected return time to S, if we performed random walks in G. By Kac's formula above, the expected return time to S equals the reciprocal of the stationary mass of S, which is just $n/|S|$. ■

The next lemma is an analogue of Claim 2.5.2 for M_S. Recall that $\ell = n^{5\delta}$.

Lemma 2.5.5 *There exists a subset $S'' \subseteq S'$, $|S''| \geq |S'|/2$, such that $\forall s \in S''$, $\|\boldsymbol{\tau}_{s,n^\delta}\|_\infty \geq 1/n^{\delta(i+6)}$.*

Proof Define event $\mathcal{E}_{s,v,h}$ as follows. The event $\mathcal{E}_{s,v,h}$ occurs when an h-hop random walk from s has length $2^{i+1}\ell$ and ends at v. Observe that $p_{s,2^{i+1}\ell}(v) = \sum_{h \leq 2^{i+1}\ell} \Pr[\mathcal{E}_{s,v,h}]$ (because the number of hops is always at most the length). Since $\boldsymbol{\tau}_{s,h}$ is a random walk vector in a symmetric Markov Chain, the infinity norm is non-increasing in h. Thus, it suffices to find a subset $S'' \subseteq S'$, $|S''| \geq |S'|/2$ such that $\forall s \in S''$, $\exists v \in S, h \geq n^\delta$, $\Pr[\mathcal{E}_{s,v,h}] \geq 1/n^{\delta(i+6)}$.

We define U_s as given in Claim 2.5.2. For all $v \in U_s$, by Claim 2.5.2, $p_{s,2^{i+1}\ell}(v) \geq 1/2n^{\delta i}$. Therefore, for all $v \in U_s$,

$$\sum_{h \leq 2^{i+1}\ell} \Pr[\mathcal{E}_{s,v,h}] \geq 1/2n^{\delta i} \tag{2.20}$$

We will construct S'' by finding s where for some $v \in U_s$, $\sum_{h \leq n^\delta} \Pr[\mathcal{E}_{s,v,h}]$ is sufficiently small.

For any h,

$$\frac{1}{|S|} \sum_{s \in S'} \sum_{v \in U_s} \Pr[\mathcal{E}_{s,v,h}](2^{i+1}\ell) \leq \mathbf{E}_{W \sim \mathcal{W}_h}[\text{length of } W] = hn/|S|$$

Suppose $h \leq 2^{i+1}\ell/n^{4\delta}$. (This is true for all $h \leq n^\delta$). Then $\sum_{s \in S'} \sum_{v \in U_s} \Pr[\mathcal{E}_{s,v,h}] \leq n^{1-4\delta}$, and $\sum_{h \leq n^\delta} \sum_{s \in S'} \sum_{v \in U_s} \Pr[\mathcal{E}_{s,v,h}] \leq n^{1-3\delta}$.

We rearrange to get

$$\sum_{s \in S'} \sum_{v \in U_s} \sum_{h \leq n^\delta} \Pr[\mathcal{E}_{s,v,h}] \leq n^{1-3\delta}$$

By the Markov bound, there is a set $S'' \subseteq S'$, $|S''| \geq |S'|/2$ such that for all $s \in S''$, $\sum_{v \in U_s} \sum_{h \leq n^\delta} \Pr[\mathcal{E}_{s,v,h}] \leq 2n^{1-3\delta}/|S'|$. By averaging, $\forall s \in S''$, $\exists v \in U_s$, such that $\sum_{h \leq n^\delta} \Pr[\mathcal{E}_{s,v,h}] \leq 2n^{1-3\delta}/(|S'| \cdot |U_s|)$. By the assumptions of Lemma 2.5.1, $|S'| \geq \alpha n \geq n^{1-\delta/2}$. Claim 2.5.2 bounds $|U_s| \geq n^{\delta(i-2)}$. Plugging these in,

$$\sum_{h \leq n^\delta} \Pr[\mathcal{E}_{s,v,h}] \leq \frac{2n^{1-3\delta}}{n^{1-\delta/2} n^{\delta(i-2)}} \leq \frac{2}{n^{\delta(i+1/2)}} \tag{2.21}$$

Subtracting this bound from (2.20), $\sum_{h \in [n^\delta, 2^{i+1}\ell]} \Pr[\mathcal{E}_{s,v,h}] \geq 1/4n^{\delta i}$. By averaging, for some $h \in [n^\delta, 2^{i+1}\ell]$, $\Pr[\mathcal{E}_{s,v,h}] \geq 1/(2^{i+3} n^{\delta i} \ell) \geq 1/n^{\delta(i+6)}$. This completes the proof. ■

We perform local partitioning on M_S, starting with arbitrary $s \in S''$. We apply the Lovász-Simonovits curve technique. (The definitions are originally from [3]. Refer to Lecture 7 of Spielman's notes [59] as well as Section 2 in Spielman-Teng [54].) This requires a series of definitions.

- Ordering of states at time t: At time t, let us order the vertices in M_S as $v_1^{(t)}, v_2^{(t)}, \ldots$ such that $\tau_{s,t}(v_1^{(t)}) \geq \tau_{s,t}(v_2^{(t)}) \ldots$, breaking ties by vertex id.

- The LS curve h_t: We define a function $h_t : [0, |S|] \to [0, 1]$ as follows. For every $k \in [|S|]$, set $h_t(k) = \sum_{j \leq k} [\tau_{s,t}(v_j^{(t)}) - 1/|S|]$. (Set $h_t(0) = 0$.) For every $x \in (k, k+1)$, we linearly interpolate to construct $h(x)$. Alternately, $h_t(x) = \max_{\vec{w} \in [0,1]^{|S|}, \|\vec{w}\|_1 = x} \sum_{v \in S} [\tau_{s,t}(v) - 1/n] w_i$.

- Level sets: For $k \in [0, |S|]$, we define the (k, t)-level set,

  $$L_{k,t} = \{v_1^{(t)}, v_2^{(t)}, \ldots, v_k^{(t)}\}.$$

  The *minimum probability* of $L_{k,t}$ denotes $\tau_{s,t}(v_k^{(t)})$.

- Conductance: for some $T \subseteq S$ we define the conductance of T in M_S to be

  $$\Phi(T) = \frac{\sum_{\substack{u \in T \\ v \in S \setminus T}} \tau_{u,1}(v)}{\min(|T|, |S \setminus T|)}$$

The main lemma of Lovász-Simonovits is the following (Lemma 1.4 of [3]).

Lemma 2.5.6 *For all k and all t,*

$$h_t(k) \leq \frac{1}{2}[h_{t-1}(k - 2\min(k, n-k)\Phi(L_{k,t})) + h_{t-1}(k + 2\min(k, n-k)\Phi(L_{k,t}))]$$

The typical use of the Lovász-Simonovitz technique is to argue about rapid mixing when all conductances (or conductances of sufficiently large sets) are lower bounded. We consider a scenario in which only sets with minimum probability at least (say) p have high conductance. In this case, we can guarantee that the largest probability will converge to p.

Lemma 2.5.7 *Suppose the following holds. For all $t' \leq t$, if the minimum probability of $L_{k,t'}$ is at least $1/10n^{\delta(i+6)}$, then $\Phi(L_{k,t'}) \geq n^{-\delta/4}$, Then, $\forall x \in [0, n]$, $h_t(x) \leq \sqrt{x}(1 - n^{-\delta/2}/4)^t + x/10n^{\delta(i+6)}$.*

Proof Notice that it suffices to show this claim for integral values of x since h_t is concave. To begin with, note that if $x = k \geq n^{\delta(i+6)}$, then the RHS is at least 1. Thus the bound is trivially true. Let us assume that $k < n^{\delta(i+6)} < n/2$. We proceed by induction over t and split into two cases based on the conductance of level sets.

Suppose k is such that $\Phi(L_{k,t}) \geq n^{-\delta/4}$. By Lemma 2.5.6 and concavity of h, we have the following at $x = k$

$$\begin{aligned}
h_t(k) &\leq \frac{1}{2}\left(h_{t-1}(k(1 - 2n^{-\delta/4})) + h_{t-1}(k(1 + 2n^{-\delta/4}))\right) && (2.22)\\
&\leq \frac{1}{2}\Bigg(\sqrt{k(1 - 2n^{-\delta/4})}(1 - n^{-\delta/2}/4)^{t-1} \\
&\qquad + \sqrt{k(1 + 2n^{-\delta/4})}(1 - n^{-\delta/2}/4)^{t-1} + \frac{2k}{10n^{\delta(i+6)}}\Bigg) && (2.23)\\
&\leq \frac{1}{2}\left(\sqrt{k}(1 - 2n^{-\delta/4})^{t-1}(\sqrt{1 - 2n^{-\delta/4}} + \sqrt{1 + 2n^{-\delta/4}}) + \frac{2k}{10n^{\delta(i+6)}}\right) && (2.24)\\
&\leq \sqrt{k}(1 - n^{\delta/2}/2)^t + k/n^{\delta(i+6)} && (2.25)
\end{aligned}$$

For the last inequality we use the bound $\left(\sqrt{1+z} + \sqrt{1-z}\right)/2 \leq 1 - z^2/8$.

Now, consider the case where k is such that $\Phi(L_{k,t}) \leq n^{-\delta/4}$. By assumption, it must be that $L_{k,t'}$ must have minimum probability less than $1/10n^{\delta(i+6)}$. Let k' be the largest integer less than k such that $\Phi(L_{k',t}) \geq n^{-\delta/4}$. By the previous case, $h_t(k') \leq \sqrt{k'}(1 - n^{\delta/2}/2)^t + k/n^{\delta(i+6)}$. Using this and the concavity of h_t, we get

$$h_t(k) \leq h_t(k') + (k - k')/10n^{\delta(i+6)} \tag{2.26}$$

$$\leq \sqrt{k'}(1 - n^{-\delta/2}/2)^t + k'/10n^{\delta(i+6)} + (k - k')/10n^{\delta(i+6)} \tag{2.27}$$

$$\leq \sqrt{k}(1 - n^{-\delta/2}/2)^t + k/10n^{\delta(i+6)} \tag{2.28}$$

■

2.5.2 Proof of Lemma 2.5.1

Proof Define S'' as given in Lemma 2.5.5. For any $s \in S''$, $\|\boldsymbol{\tau}_{s,n^\delta}\|_\infty \geq 1/n^{\delta(i+6)}$. By the definition of the LS curve, $h_{n^\delta}(1) \geq 1/n^{\delta(i+6)}$. Suppose (for contradiction's sake) all level sets for $t \leq n^\delta$ with minimum probability at least $1/10n^{\delta(i+6)}$ have conductance at least $n^{-\delta/4}$. By Lemma 2.5.7, $h_{n^\delta}(1) \leq (1 - n^{-\delta/2}/4)^{n^\delta} + 1/10n^{\delta(i+6)}$ $< 1/n^{\delta(i+6)}$. This contradicts the bound obtained by Lemma 2.5.5.

Thus, for every $s \in S''$, there exists some level set for $t_s \leq n^\delta$ with minimum probability at least $1/10n^{\delta(i+6)}$ and conductance $< n^{-\delta/4}$. Let us call this level set P_s. We also use the fact that $|P_s| < |S|/2$. By the construction of M_S, we have,

$$\Phi(P_s) \geq \frac{\sum_{\substack{x \in P_s \\ y \in S \setminus P_s}} \tau_{x,1}(y)}{\min(|P_s|, |S \setminus P_s|)} = \frac{E(P_s, S \setminus P_s)}{2d|P_s|}$$

The first inequality follows because we restrict the numerator to length one transitions in the Markov Chain M_S (which correspond to edges in G). Rearranging, we get $E(P_s, S \setminus P_s) \leq n^{-\delta/4}(2d|P_s|)$.

For all $s \in S''$ and $v \in P_s$, $\tau_{s,n^\delta}(v) \geq 1/10n^{\delta(i+6)}$. Set $L = 160n^{\delta(i+7)}/\alpha$. Let $\widetilde{S}$ be the subset of S'' such that $\forall s \in \widetilde{S}$, P_s is such that $\forall v \in P_s$, $\sum_{l \leq L} p_{s,l}(v) \geq 1/20n^{\delta(i+6)}$. By averaging, $\exists l \leq L$ such that $p_{s,l}(v) \geq \alpha/n^{\delta(2i+14)}$.

We have seen that $\widetilde{S}$ satisfies the two desired properties: for all $s \in \widetilde{S}$ $E(P_s, S \setminus P_s) \leq 2n^{-\delta/4}d|P_s|/\alpha$ and for all $v \in P_s$, $\exists t \leq 160n^{\delta(i+7)}$ such that $p_{s,t}(v) \geq \alpha/n^{\delta(2i+14)}$.

It only remains to prove a lower bound on size, or alternately, an upper bound on $|S'' \setminus \widetilde{S}|$.

Consider any $s \in S'' \setminus \widetilde{S}$. There exists some $v_s \in P_s$ such that $\tau_{s,n^\delta}(v_s) \geq 1/10n^{\delta(i+6)}$ but $\sum_{l \leq L} p_{s,l}(v_s) < 1/20n^{\delta(i+6)}$. Let us use $\hat{p}_{s,l}(v_s)$ to denote the probability of reaching v_s from s in an l-length walk that makes n^δ hops. Observe that

$$\tau_{s,n^\delta}(v_s) = \sum_{l \geq n^\delta} \hat{p}_{s,l}(v_s) = \sum_{l=n^\delta}^{L} \hat{p}_{s,l}(v_s) + \sum_{l>L} \hat{p}_{s,l}(v_s) \leq \sum_{l=n^\delta}^{L} p_{s,l}(v_s) + \sum_{l>L} \hat{p}_{s,l}(v_s) \tag{2.29}$$

$$< 1/20n^{\delta(i+6)} + \sum_{l>L} \hat{p}_{s,l}(v_s) \tag{2.30}$$

The last inequality follows from the fact that $s \in S'' \setminus \widetilde{S}$, and hence $\sum_{l=n^\delta}^{L} p_{s,l}(v_s) < 1/20n^{\delta(i+6)}$. Since $\tau_{s,n^\delta}(v_s) \geq 1/10n^{\delta(i+6)}$, it follows that $\sum_{l>L} \hat{p}_{s,l}(v_s) > 1/20n^{\delta(i+6)}$. Thus,

$$\frac{1}{|S|} \sum_{s \in S'' \setminus \widetilde{S}} \sum_{l>L} \hat{p}_{s,l}(v_s) L > \frac{|S'' \setminus \widetilde{S}| \cdot L}{|S| 20n^{\delta(i+6)}} = \frac{160\alpha^{-1} n^{\delta(i+7)} \cdot |S'' \setminus \widetilde{S}|}{20|S|n^{\delta(i+6)}} = \frac{8n^\delta |S'' \setminus \widetilde{S}|}{\alpha |S|} \tag{2.31}$$

By Lemma 2.5.4,

$$\frac{1}{|S|} \sum_{s \in S'' \setminus \widetilde{S}} \sum_{l>L} \hat{p}_{s,l}(v_s) L \leq \mathbf{E}_{W \sim \mathcal{W}_n^\delta}[\text{length of } W] = \frac{n^{1+\delta}}{|S|} \tag{2.32}$$

Combining the above, $|S'' \setminus \widetilde{S}| \leq \alpha n/8$. By Lemma 2.5.5, $|S''| \geq |S'|/2 \geq \alpha n/2$, yielding the bound $|\widetilde{S}| \geq \alpha n/4$.

■

2.6 Wrapping it all up: the proof of Theorem 2.2.1

We have all the tools required to complete the proof of Theorem 2.2.1. Our aim is to show that if `FindMinor`(G, ε, H) outputs an H-minor with probability $< 2/3$, then G is ε-close to being H-minor-free. Henceforth in this section, we will simply assume the "if" condition.

The following decomposition procedure is used by the proof. We set parameter $\alpha = \varepsilon/(50r^4 \log n)$.

`Decompose`(G)

1. Initialize $S = V$ and $\mathcal{P} = \emptyset$.
2. For $i = 1, \ldots, 5r^4$:
   (a) Assign $S' := \left\{ s \in S : ||\boldsymbol{q}_{[S],s}^{(i+1)}||_2^2 \geq 1/n^{\delta i} \right\}$
   (b) While $|S'| \geq \alpha n$:
      i. Choose arbitrary $s \in S''$, and let P_s be as in Lemma 2.5.1.
      ii. Add P_s to $\mathcal{P}$ and assign $S := S \setminus P_s$
      iii. Assign $S' := \left\{ s \in S : ||\boldsymbol{q}_{[S],s}^{(i+1)}||_2^2 \geq 1/n^{\delta i} \right\}$
   (c) Assign $S := S \setminus S'$
   (d) Assign $X_i := S'$
3. Let $X = \bigcup_i X_i$.
4. Output the partition $\mathcal{P}, X, S$

The procedure `Decompose` repeatedly employs Lemma 2.5.1 for values of $i \leq 5r^4$. In the ith iteration, eventually $|S'|$ becomes too small for Lemma 2.5.1. Then, S' is moved (from S) to an "excess" set X_i, and the next iteration begins. `Decompose` ends with a partition $\mathcal{P}, X, S$ where each set in $\mathcal{P}$ is a low conductance cut, X is fairly small, and `FindBiclique` succeeds with high probability on every vertex in S.

This is formalized in the next lemma.

Lemma 2.6.1 *Assume $\varepsilon > \varepsilon_{\mathsf{CUTOFF}}$. Suppose* `FindMinor`$(G, \varepsilon, H)$ *outputs an H-minor with probability $< 2/3$. Then, the output of* `Decompose` *satisfies the following conditions.*

- $|X| \leq \varepsilon n/10$.
- $|S| \leq \varepsilon n/10$.
- $\forall P_s \in \mathcal{P}, v \in P_s$, $\exists t \leq 160 n^{6\delta r^4}/\alpha$ *such that* $p_{s,t}(v) \geq \frac{\alpha}{n^{11\delta r^2}}$.
- *There are at most $\varepsilon n/10$ edges that go between different P_s sets.*

Proof Consider the X_i's formed by `Decompose`. Each of these has size at most $\alpha n = \varepsilon n/50r^4 \log n$, and there are at most $5r^4$ of these. Clearly, their union has size at most $\varepsilon n/10$.

The third condition holds directly from Lemma 2.5.1. Consider the number of edges that go between P_s and the rest of S, when P_s was constructed (in `Decompose`). By Lemma 2.5.1 again, the number of these edges is at most

$$2n^{-\delta/4}d|P_s|/\alpha = 40r^4(\log n)\varepsilon^{-1}n^{-\delta/4}d|P_s|.$$

Note that $\varepsilon > \varepsilon_{\mathsf{CUTOFF}}$. For sufficiently small constant δ, the number of edges between P_s and $S \setminus P_s$ (at the time of removal) is at most $\varepsilon|P_s|/10$. The total number of such edges is at most $\varepsilon n/10$ (since P_s are all disjoint).

Suppose, for contradiction's sake, that $|S| > \varepsilon n/10$. Consider the stratification process with $R_0 = S$. By construction of S, $\forall s \in S$, $||\boldsymbol{q}^{(5r^4+1)}_{[S],s}|| \leq 1/n^{5\delta r^4}$. Thus, all of these vertices will lie in strata numbered $5r^4$ or above. Since $\varepsilon > \varepsilon_{\mathsf{CUTOFF}}$, by Lemma 2.3.5, at most $\varepsilon n/\log n$ vertices are in strata numbered more than $1/\delta + 3$. By Theorem 2.4.1, for at least $\varepsilon n/10 - \varepsilon n/\log n \geq \varepsilon n/20$ vertices, the probability that the paths discovered by `FindBiclique`(s) contain a K_{r^2,r^2}-minor is at least $n^{-4\delta r^4}$. Since a K_{r^2,r^2} minor contains an H-minor, the algorithm (in this situation) will succeed in finding an H-minor.

All in all, this implies that the probability that a single call to `FindBiclique` finds an H minor is at least $n^{-5\delta r^4}$. Since `FindMinor` makes $n^{20\delta r^4}$ calls to `FindBiclique`, an H-minor is found with probability at least $5/6$. This is a contradiction, and we conclude that $|S| \leq \varepsilon n/10$.

■

And now, we can prove the correctness guarantee of `FindMinor`.

Claim 2.6.2 *Suppose* `FindMinor`(G, ε, H) *outputs an* H*-minor with probability* $<$ $2/3$. *Then* G *is* ε*-close to being* H*-minor-free.*

Proof If $\varepsilon \leq \varepsilon_{\mathsf{CUTOFF}}$, then `FindMinor` runs an exact procedure. So the claim is clearly true. Henceforth, assume $\varepsilon > \varepsilon_{\mathsf{CUTOFF}}$. Apply Lemma 2.6.1 to partition V into $\mathcal{P}, X, S$.

Call $s \in V$ bad if there is a corresponding $P_s \in \mathcal{P}$ and P_s induces an H-minor. By Lemma 2.6.1, for all $v \in P_s$, $\exists t \leq 160n^{6\delta r^4}/\alpha$ such that $p_{s,t}(v) \geq \alpha/n^{11\delta r^4}$. Note that $160n^{6\delta r^4}/\alpha \leq n^{7\delta r^4}$ and $\alpha/n^{11\delta r^4} \geq n^{-12\delta r^4}$. Also, $|P_s| \leq 160(n^{6\delta r^4}/\alpha) \times (n^{11\delta r^4}/\alpha) \leq n^{18\delta r^4}$. Note that `LocalSearch`$(s)$ performs walks of all lengths up to $n^{7\delta r^4}$, and performs $n^{30\delta r^4}$ walks of each length. For any $v \in P_s$, the probability that `LocalSearch`(s) does not add v to B (the set of "discovered" vertices in `LocalSearch`(s)) is at most $(1 - n^{-12\delta r^4})^{n^{30\delta r^4}} \leq 1/n^2$. Taking a union bound over P_s, the probability that P_s is not contained in B is at most $1/n$. Consequently, for bad s, `LocalSearch`(s) outputs an H-minor with probability $> 1 - 1/n$.

Suppose there are more than $n^{1-30\delta r^4}$ bad vertices. The probability that a u.a.r. $s \in V$ is bad is at least $n^{-30\delta r^4}$. Since `FindMinor`(G, ε, H) invokes `LocalSearch` $n^{35\delta r^4}$ times, the probability that `LocalSearch`(s) is invoked for a bad vertex is at least $1 - 1/n$. Thus, `FindMinor`(G, ε, H) outputs an H-minor with probability $> 1 - 2/n$, contradicting the claim assumption.

We conclude that there are at most $n^{1-30\delta r^4}$ bad vertices. Each P_s has at most $n^{18\delta r^4}$ vertices, and $|\bigcup_{s \text{ bad}} P_s| \leq n^{1-12\delta r^4} \leq \varepsilon n/10$.

We can make G H-minor-free by deleting all edges incident to X, all edges incident to S, all edges incident to vertices in any bad P_s sets, and all edges between P_s sets. By Lemma 2.6.1 and the bound given above, the total number of edges deleted is at most $4\varepsilon dn/10 < \varepsilon dn$. ∎

Finally, we bound the running time.

Claim 2.6.3 *The running time of* `FindMinor`(G, ε, H) *is*

$$dn^{1/2+O(\delta r^2)} + d\varepsilon^{-2\exp(2/\delta)/\delta}$$

Proof If $\varepsilon < \varepsilon_{\mathsf{CUTOFF}}$, then the running time is simply $O(n^2)$. Since $\varepsilon < n^{-\delta/\exp(2/\delta)}$, this can be expressed as $\varepsilon^{-2\exp(2/\delta)/\delta}$.

Assume $\varepsilon \geq \varepsilon_{\mathsf{CUTOFF}}$. The total number of vertices encountered by all the `LocalSearch` calls is $n^{O(\delta r^4)}$. There is an extra d factor to determine all incident edges through vertex queries. Thus, the total running time is $dn^{O(\delta r^4)}$, because of the quadratic overhead of `KKR`. Consider a single iteration for the main loop of `FindBiclique`. First, `FindBiclique` performs $2r^2$ random walks of length $2^{i+1}n^{5\delta}$, and then for each of these, `FindPath` performs $n^{\delta i/2+9\delta}$ walks of length $2^i n^{5\delta}$. Hence, the total steps (and thus queries) in all walks done in a single call to `FindBiclique` is

$$\sum_{i=5r^2}^{1/\delta+3} \left(2r^2 2^{i+1} n^{5\delta} + 2r^2 n^{\delta i/2+9\delta} 2^i n^{5\delta}\right) = r^2 n^{1/2+O(\delta)}. \tag{2.33}$$

While this is the total number of vertices encountered, we note that the calls made to `KKR`(F, H) are for much smaller graphs. The output of find path has size $O(2^{1/\delta} n^{5\delta})$, and the subgraph F constructed has at most $O(2^{1/\delta} n^{5\delta})$ vertices. We incur an extra d factor to determine the induced subgraph through vertex queries. Thus, the time for each call to `KKR`(F, H) is $n^{O(\delta)}$. There are $n^{O(\delta r^4)}$ calls to `FindBiclique`, and we can bound the total running time by $dn^{1/2+O(\delta r^4)}$. ■

Acknowledgements

We would like to acknowledge Madhur Tulsiani for his improvements to Lemma 2.3.1.

3 A TWO-SIDED-TESTER FOR MINOR FREENESS IN BOUNDED DEGREE GRAPHS

The classic result of Hopcroft-Tarjan gives a linear time algorithm for deciding planarity [28]. As the old theorems of Kuratowski and Wagner show, planarity is characterized by the non-existence of K_5 and $K_{3,3}$ minors [29,30]. The monumental graph minor theorem of Robertson-Seymour proves that any property of graphs closed under minors can be expressed by the non-existence of a finite list of minors [15,16,31]. Moreover, given a fixed graph, H, the property of being H-minor-free can be decided in quadratic time [24]. Thus, any minor-closed property of graphs can be decided in quadratic time.

What if an algorithm is not allowed to read the whole graph? This question was first addressed in the seminal result of Benjamini-Schramm-Shapira (BSS) in the language of property testing [32]. Consider the model of random access to a graph adjacency list, as introduced by Goldreich-Ron [33]. Let $G = (V, E)$ be a graph where $V = [n]$ and the maximum degree is d. We have random access to the list through *neighbor queries*. There is an oracle that, given $v \in V$ and $i \in [d]$, returns the ith neighbor of v (if no neighbor exists, it returns $\perp$).

For a property $\mathcal{P}$ of graphs with degree bound d, the distance of G to $\mathcal{P}$ is the minimum number of edge additions/removals required to make G have $\mathcal{P}$, divided by dn. We say that G is ε-far from $\mathcal{P}$ if the distance to $\mathcal{P}$ is more than ε. A property tester for $\mathcal{P}$ is a randomized procedure that takes as input (query access to) G and a proximity parameter, $\varepsilon > 0$. If $G \in \mathcal{P}$, the tester must accept with probability at least $2/3$. If G is ε-far from $\mathcal{P}$, the tester must reject with probability at least $2/3$. A tester is one-sided if it accepts $G \in \mathcal{P}$ with probability 1.

Let $\mathcal{P}$ be some minor-closed property such as planarity. BSS proved the remarkable result that any such $\mathcal{P}$ is testable in time independent of n. Their query complexity was triply exponential in (d/ε). Hassidim-Kelner-Nguyen-Onak improved this complexity to singly exponential, introducing the novel concept of partition oracles [41]. Levi-Ron gave a more efficient analysis, proving the existence of testers with query complexity quasi-polynomial in (d/ε) [42]. For the special cases of outerplanarity and bounded treewidth, $\mathrm{poly}(d/\varepsilon)$ query testers are known [43, 44].

It has been a significant open problem to get $\mathrm{poly}(d/\varepsilon)$ query testers for all minor-closed properties. In Open Problem 9.26 of Goldreich's recent book on property testing, he states the "begging question of whether [the query complexity bound of testing minor-closed properties] can be improved to a polynomial [in $1/\varepsilon$]" [38]. Even for classic case of planarity, this was unknown.

In this paper, we resolve this open problem.

Theorem 3.0.1 *Let $\mathcal{P}$ be any minor-closed property of graphs with degree bound d. There exists a (two-sided) tester for $\mathcal{P}$ that runs in $d^2 \cdot \mathrm{poly}(\varepsilon^{-1})$ time.*

Thus, properties such as planarity, series-parallel graphs, embeddability in bounded genus surfaces, linkless embeddable, and bounded treewidth are all testable in time $d^2 \cdot \mathrm{poly}(\varepsilon^{-1})$.

By the graph minor theorem of Robertson-Seymour [31], Theorem 3.0.1 is a corollary of our main result for testing H-minor-freeness. As alluded to earlier, for any minor-closed property $\mathcal{P}$, there exists a finite list of graphs $\{H_1, H_2, \ldots, H_b\}$ satisfying the following condition. A graph G is in $\mathcal{P}$ iff for all $i \leq b$, G does not contain an H_i-minor. Let $\mathcal{P}_{H_i}$ be the property of being H_i-minor-free. The characterization implies that if G is ε-far from $\mathcal{P}$, there exists $i \leq b$ such that G is $\Omega(\varepsilon)$-far from $\mathcal{P}_{H_i}$. Thus, property testers for H_i-minor freeness imply property testers for $\mathcal{P}$ (with constant blowup in the proximity parameter).

Our main quantitative theorem follows.

Theorem 3.0.2 *There is an absolute constant c such that the following holds. Fix a graph H with r vertices. The property of being H-minor-free is testable in $d(r/\varepsilon)^c$ queries and $d^2(r/\varepsilon)^{2c}$ time.*

We stress that c is independent on r. Currently, our value of c is likely more than 100, and we have not tried to optimize the exponent of ε. We believe that significant improvement is possible, even by just tightening the current analysis. It would be of significant interest to get a better bound, even for the case of planarity.

3.0.1 Related work

Property testing on bounded-degree graphs is a large topic, and we point the reader to Chapter 9 of Goldreich's book [38]. Graph minor theory is immensely deep, and Chapter 12 of Diestel's book is an excellent reference [36]. We will focus on the work regarding property testing of H-minor-freeness.

As noted before, this line of work started with Benjamini-Schramm-Shapira [32]. Their tester basically approximates the frequency of all subgraphs with radius $2^{1/\varepsilon}$, which leads to the large dependence in d/ε. Central to their result (and subsequent) work is the notion of *hyperfiniteness*. A hyperfinite class of graphs has the property that the removal of a small constant fraction of edges leaves connected components of constant size. Hassidim-Kelner-Nguyen-Onak design *partition oracles* for hyperfinite graphs to get improved testers [41,42]. These oracles are local procedures that output the connected component that a vertex lies in, without explicit knowledge of any global partition. This is extremely challenging as one has to maintain consistency among different queries. The final construction is an intricate recursive procedure that makes $\exp(d/\varepsilon)$ queries. Levi-Ron gave a significantly simpler and more efficient analysis leading to their query complexity of $(d\varepsilon^{-1})^{\log \varepsilon^{-1}}$. Newman-Sohler show how partition oracles lead to testers for any property of hyperfinite graphs [60].

Given the challenge of $\text{poly}(d\varepsilon^{-1})$ testers for planarity, there has been focus on other minor-closed properties. Yoshida-Ito give such a tester for outerplanarity [43], which was subsumed by a $\text{poly}(d\varepsilon^{-1})$ tester by Edelman et al for bounded treewidth graphs [44]. Nonetheless, $\text{poly}(d\varepsilon^{-1})$ testers for planarity remained open.

Unlike general (two-sided) testers, one-sided testers for H-minor-freeness must have a dependence on n. BSS conjectured that the complexity of testing H-minor-freeness (and specifically planarity) is $\Theta(\sqrt{n})$. Czumaj et al [34] showed such a lower bound for any H containing a cycle, and gave an $\widetilde{O}(\sqrt{n})$ tester when H is a cycle. Fichtenberger-Levi-Vasudev-Wötzel give an $\widetilde{O}(n^{2/3})$ tester for H-minor-freeness when H is $K_{2,k}$, the $(k \times 2)$-grid or the k-circus graph [35]. Recently, Kumar-Seshadhri-Stolman (henceforth KSS) nearly resolved the BSS conjecture with an $n^{1/2+o(1)}$-query one-sided tester for H-minor-freeness [6]. The underlying approach uses the proof strategy of the bipartiteness tester of Goldreich-Ron [45].

The body of work on two-sided (independent of n) testers is primarily combinatorial. The proof of Theorem 3.0.2 is a significant deviation from this line of work, and is inspired by the spectral graph theoretic methods in KSS. As we explain in the next section, we do not require the full machinery of KSS, but we do follow the connections between random walk behavior and graph minors. The tester of Theorem 3.0.2 is simpler than those of Hassidim et al and Levi-Ron, who use recursive algorithms to construct partition oracles [41, 42].

3.0.2 Main ideas

Let us revisit the argument of KSS, that gives an $n^{1/2+o(1)}$-query one-sided tester for H-minor-freeness. We will take great liberties with parameters, to explain the essence. The proof of Theorem 3.0.2 is inspired by the approach in KSS, but the proof details deviate significantly. We discover that the full machinery is not required. But the main idea is to exploit connections between random walk behavior and graph minor-freeness.

First, we fix a random walk length $\ell = n^\delta \gg 1/\varepsilon$, for small constant $\delta > 0$. One of the building blocks is a random walk procedure that finds H-minors by performing $\sqrt{n} \cdot \mathrm{poly}(\ell)$ random walks of length ℓ. For our purposes, it is not relevant what the algorithm is, and we simply refer to this as the "random walk procedure". One of the significant concepts in KSS is the notion of a *returning random walk*. For any subset of vertices $S \subset V$, an S-returning random walk of length ℓ is a random walk that starts from S and ends at S. For any vertex $s \in S$, we use $\mathbf{q}_{[S],s,\ell}$ to denote the $|S|$-dimensional vector of probabilities of an S-returning walk of length ℓ starting from s.

KSS proves the following two key lemmas. We use c to denote some constant that depends only on H.

1. Suppose there is a subset $S \subseteq V$, $|S| \geq n/\ell$, with the following property. For at least half the vertices $s \in S$, $\|\mathbf{q}_{[S],s,\ell}\| \leq \ell^{-c}$. Then, whp, the $\sqrt{n} \cdot \text{poly}(\ell)$-time random walk procedure finds an H-minor.

2. Suppose there is a subset $S \subseteq V$, $|S| \geq n/\ell$, with the following property. For at least half the vertices $s \in S$, $\|\mathbf{q}_{[S],s,\ell}\| > \ell^{-c}$. Then, for every such vertex s, there is a cut of conductance at most $1/\ell$ contained in S, where all vertices (in the cut) are reached with probability at least $1/\text{poly}(\ell)$ by ℓ-length S-returning walks from s.

To get a one-sided tester, we run the $\sqrt{n} \cdot \text{poly}(\ell)$ random walk procedure. If it does not find an H-minor, then the antecedent of the second part above is true for all S such that $|S| \geq n/\ell$. The consequent basically talks of local partitioning *within* S, even though random walks are performed in the whole graph G. The statement is proven using arguments from local partitioning theorems of Spielman-Teng [54]. By iterating the argument, we can prove the existence of a set of εdn edges, whose removal breaks G into connected components of size at most $\text{poly}(\ell)$. Moreover, a superset of any piece can be "discovered" by performing $\text{poly}(\ell)$ random walks (of length ℓ) from some starting vertex. Roughly speaking, each piece has a distinct starting vertex. Thus, if G was ε-far from being H-minor-free, an ε-fraction (by size) of the pieces will contain H-minors. A procedure that picks $\text{poly}(\ell)$ random vertices (to hit the starting vertex of these pieces) and runs $\text{poly}(\ell)$ random walks of length ℓ will, whp, cover a subgraph that contains an H-minor. We refer to this as the "local search procedure", which runs in $\text{poly}(\ell)$ time.

This sums up the KSS approach. Observe that in the first case above, by the probabilistic method, we are guaranteed the existence of a minor. Let us abstract out the argument as follows. Let $\boldsymbol{Q}$ be the statement/condition: there exists a subset $S \subseteq V$, $|S| \geq n/\ell$ such that for at least half the vertices $s \in S$, $\|\mathbf{q}_{[S],s,\ell}\| \leq \ell^{-c}$. KSS basically proves the following lemmas, which we refer to subsequently as Lemma 1 and Lemma 2.

1. $\boldsymbol{Q} \Rightarrow G$ contains an H-minor.
2. $\neg\boldsymbol{Q} \Rightarrow$ If G is ε-far from being H-minor-free, the local search procedure finds an H-minor whp.

We now have an approach to get a poly(ε^{-1}) tester. Suppose we could set the random walk length ℓ to be poly(ε^{-1}). And suppose we could test the condition $\boldsymbol{Q}$ in time poly(ε^{-1}). We could then run local search on top of this, and get a bonafide tester.

A simple adaptation of proofs of both Lemma 1 and Lemma 2 run into some fundamental difficulties. The proof of Lemma 1 crucially requires ℓ to be n^δ (or at least $\Omega(\log n)$). The existence of the minor is shown through the success of the $\sqrt{n} \cdot \text{poly}(\ell)$ random walk procedure. Constant length random walks cannot find an H-minor, even if G was $\Omega(1)$-far from being H-minor-free (G could be a 3-regular expander).

From hyperfiniteness to $\ell = \text{poly}(\varepsilon^{-1})$. We employ a different (and simpler) approach to reduce the walk length. A classic result of Alon-Seymour-Thomas asserts that any H-minor-free bounded-degree graph G satisfies the following "hyperfinite" decomposition: for any $\alpha \in (0,1)$, we can remove an α-fraction of the edges to get connected components of size $O(\alpha^{-2})$. Let us set $\alpha = \text{poly}(\varepsilon)$ and the walk length $\ell \ll 1/\alpha$. We can show that ℓ-length random walks in G encounter the removed edges with very low probability. By and large, the walks behave as if they were performed on the decomposition. Thus, walks in G are "trapped" in the small components of size $O(\alpha^{-2})$. Quantitatively, we can show that most vertices s, $|\mathbf{p}_s^\ell\|_2 \geq$

poly(ε). (We use $\mathbf{p}_s^\ell$ to denote the random walk distribution with starting vertex s.) By the contrapositive: if there are at least poly(ε)-fraction of vertices s such that $|\mathbf{p}_s^\ell\|_2 \leq \text{poly}(\varepsilon)$, then G contains an H-minor. This is easily testable. We get a more convenient, poly(ε^{-1})-query testable version of Lemma 1.

Clipped norms for local partitioning. We can now express our new condition $\neg\boldsymbol{Q}$ as: for more than a $(1 - \text{poly}(\varepsilon))$-fraction of vertices s, $\|\mathbf{p}_s^\ell\|_2 \geq \text{poly}(\varepsilon)$. This is a *weakening* of the antecedent. Previously, the condition referred to returning walks, which have smaller norm. Furthermore, the returning walks specifically reference S, the set in which we are performing local partitioning. Thus, we have some conditions on the behavior of random walks within S itself, which is necessary to perform the local partitioning. Our new condition only refers to the l_2-norms of random walks in G.

The new condition appears to be too fragile to get local partitioning within S. It is possible that the l_2-norm of $\mathbf{p}_s^\ell$ is dominated by a few vertices outside of S, whose l_1-norm is tiny. In other words, an event of small probability dominates the l_2-norm. The existing proof of Lemma 2 (from KSS) is not sensitive enough to handle such situations.

We overcome this problem by using a more robust version of norm, called the *clipped norm*. We define $\text{cl}(\boldsymbol{x}, \xi)$ for distribution vector $\boldsymbol{x}$ and $\xi \in (0, 1)$ to be the smallest l_2-norm obtained by removing ξ probability mass (l_1-norm) from $\boldsymbol{x}$. In other words, we can measuring the l_2-norm after "clipping" away ξ probability worth of outliers. We can prove a version of Lemma 2 with a lower bound of the clipped norm. We need to now rework Lemma 1 in terms of clipped norms. This turns out to be relatively straightforward.

Putting it all together. Our final tester is as follows. The length ℓ is set to poly(ε^{-1}). It picks some random vertices, and estimates the l_2-norm of *clipped* probability vectors of ℓ-length random walks from these vertices. If sufficiently many of them have "small" (poly(ε^{-1})) norms, then the tester rejects. Otherwise, it runs poly(ε^{-1}) walks to find a superset of a low conductance cut. The tester employs some exact H-minor finding algorithm on the observed subgraph.

3.1 The algorithm

In the algorithm and analysis, we will use the following notation.

- **Random walks** - Unless stated otherwise, we consider lazy random walks on graphs. If the walk is at a vertex, v, it transitions to each neighbor of v with probability $1/2d$ and remains at v with probability $1 - \frac{d_v}{2d}$ where d_v is the degree of the vertex v. Note that the stationary distribution is uniform. We use M to denote the transition matrix of this random walk.

- $\mathbf{p}_v^t$ - the n-dimensional probability vector, where the uth entry is the probability that a length t random walk started from v ends at u. We denote each entry as $\mathbf{p}_v^t(u)$.

- $\| \cdot \|_p$ - the usual l_p norm on vectors.

The two parameters to the algorithm are $\varepsilon \in [0, 1/2]$, and a graph H on $r \geq 3$ vertices. We set the walk length $\ell = \alpha r^3 + \lceil \varepsilon^{-20} \rceil$, where α is some absolute constant.

Our algorithm runs as a subroutine the exact quadratic time minor-finding algorithm of Kawarabayashi-Kobayashi-Reed [24]. We denote this procedure by `KKR`.

`FindMinor`(G, ε, H)

1. Pick multiset S of ℓ^{21} uniform random vertices.
2. For every $s \in S$, run `EstClip`(s) and `LocalSearch`(S).
3. If any call to `LocalSearch` returns FOUND, REJECT.
4. If more than $2\ell^{20}$ calls to `EstClip` return LOW, REJECT.
5. ACCEPT

`LocalSearch`(s)

1. Perform ℓ^{21} independent random walks of length ℓ^{11} from s. Add all the vertices encountered to set B_s.
2. Determine $G[B_s]$, the subgraph induced by B_s.
3. If $\texttt{KKR}(G[B_s], H)$ finds an H-minor, return FOUND.

`EstClip`(s)

1. Perform $w = \ell^{14}$ walks of length ℓ from s.
2. For every vertex v, let $w_v =$ number of walks that end at v.
3. Let $T = \{v \mid w_v \geq \ell^7/2\}$.
4. If $\sum_{v \in T} w_v \geq w/3$, output HIGH, else output LOW.

Theorem 3.0.2 follows directly from the following theorems.

Theorem 3.1.1 *If G is H-minor-free,* `FindMinor` *outputs ACCEPT with probability at least* $2/3$.

Theorem 3.1.2 *If G is ε-far from H-minor-freeness, then* `FindMinor` *outputs REJECT with probability at least* $2/3$.

Claim 3.1.3 *There exists an absolute constant, c such that the query complexity of* `FindMinor` *is $O(d(r/\varepsilon)^c)$ and time complexity is $O(d^2(r/\varepsilon)^2)$.*

Proof The entire algorithm is based on performing $\mathrm{poly}(\ell)$ random walks of length $\mathrm{poly}(\ell)$. Note that $\ell = \mathrm{poly}(r/\varepsilon)$. The dependence on d appears because the subgraph $G[B_s]$ is constructed by query the neighborhood of all vertices in B_s. The quadratic overhead in running time is because of KKR. ■

3.2 Random walks do not spread in minor-free graphs

We first define the clipped norm.

Definition 3.2.1 *Given $\boldsymbol{x} \in (\mathbb{R}^+)^{|V|}$ and parameter $\xi \in [0, 1)$, the ξ-clipped vector $\mathrm{cl}(\boldsymbol{x}, \xi)$ is the lexicographically least vector $\boldsymbol{y}$ optimizing the program:* $\min \|\boldsymbol{y}\|_2$, *subject to $\|\boldsymbol{x} - \boldsymbol{y}\|_1 \leq \xi$ and $\forall v \in V, \boldsymbol{y}(v) \leq \boldsymbol{x}(v)$.*

The clipping operation removes “outliers” from a vector, with the intention of minimizing the l_2-norm. For a probability distribution $\mathbf{p}_s^\ell$, a small value of $\|\mathbf{p}_s^\ell\|_2^2$ is a measure of the spread of the walk. But this is a crude lens. There may be one large coordinate in $\mathbf{p}_s^\ell$ that determines the norm, while all other coordinates are (say) uniform. The clipped norm better captures (for our purposes) the notion of a random walk spreading.

We state the main result of this section. The constant 3/8 below is just for convenience, and can be replaced by any non-zero constant (with a constant drop in the lower bound).

Lemma 3.2.1 *There is an absolute constant α such that the following holds. Let H be a graph on r vertices. Suppose G is a H-minor-free graph. Then for any $\ell \geq \alpha r^3$, there exists at least $(1-1/\ell)n$ vertices such that $\|\mathrm{cl}(\mathbf{p}_v^\ell, 3/8)\|_2^2 \geq \ell^{-7}$.*

In order to show this lemma, we will use the classic decomposition theorem for minor-free graphs by Alon-Seymour-Thomas [21]. It originally appears phrased in terms of a weight function $w : V \to \mathbb{R}^+$. We use the uniform weight function $\forall v \in V$ $w(v) = 1/n$ to obtain the restatement below.

Lemma 3.2.2 (Proposition 4.1 of [21]) *There is an absolute constant α such that the following holds. Let H be a graph on r vertices. Suppose G is an H-minor-free graph with maximum degree d. Then, for all $k \in \mathbb{N}$, there exists a set of at most $\alpha n r^{3/2}/k^{1/2}$ vertices whose removal leaves G will all connected components of size at most k.*

It is convenient to think of the Markov chain on G in terms of a multigraph on G, with $2d$ edges from each vertex. Each edge has probability exactly $1/2d$, and self-loops consist of many such edges. Note that every edge of the original graph is a single edge in this multigraph. For any subset of vertices $C \subseteq V$, let us define the random walk restricted to C. We remove every cut edge (u, v) (where $u \in C$ and $v \notin C$) and add a self-loop of the same probability at u. This produces a Markov chain on C that is symmetric. Given a subset C and $v \in C$, we use $\mathbf{p}'_{v,t}$ to denote the distribution of endpoints of t-length random walk starting from v and restricted to C. (In our use, C will apparent from context, so we will not carry the dependence on C in the notation.)

The following claim relates the clipped norms of the $\mathbf{p}_v^t$ and $\mathbf{p}'_{v,t}$ vectors.

Claim 3.2.3 *Let $C \subset V$ and $v \in C$. Let η be the probability that a t-length random walk from v (in G) leaves C. For any $\sigma > \eta$, $\|\mathrm{cl}(\mathbf{p}_v^t, \sigma - \eta)\|_2^2 \geq \|\mathrm{cl}(\mathbf{p}'_{v,t}, \sigma)\|_2^2$.*

Proof The random walk restricted to C is obtained by adding some self-loops that are not in the original Markov chain. Color all these self-loops red. Let $\mathbf{r}_{v,t}(u)$ be the probability of a t-length walk from v to u that contains a red edge. Any path without a red edge is a path in G (with the same probability), so $\mathbf{p}'_{v,t}(u) \leq \mathbf{p}_v^t(u) + \mathbf{r}_{v,t}(u)$.

Note that $\sum_{u \in C} \mathbf{r}_{v,t}(u)$ is the total probability of a random walk from u restricted to C encountering a red self-loop. Red self-loops correspond to cut edges in the original graph, and thus, this is the probability of encountering a cut edge. Hence, $\sum_{u \in C} \mathbf{r}_{v,t}(u) \leq \eta$.

Intuitively, we can obtain a σ-clipping of $\mathbf{p}'_{v,t}$ by first clipping at most η probability mass to get $\mathbf{p}_v^t$, and then performing a $(\sigma - \eta)$-clipping of $\mathbf{p}_v^t$. We formalize this below.

Let $\boldsymbol{q} = \mathrm{cl}(\mathbf{p}_v^t, \sigma - \eta)$, and let us define the $|C|$-dimensional vector $\boldsymbol{w}$ by $\boldsymbol{w}(u) = \min\big(\boldsymbol{q}(u), \mathbf{p}'_{v,t}(u)\big)$. Since $\boldsymbol{w}$ is non-negative and $\boldsymbol{w}(u) \leq \boldsymbol{q}(u)$ for all $u \in C$, it follows that $\|\boldsymbol{w}\|_2^2 \leq \|\boldsymbol{q}\|_2^2 = \|\mathrm{cl}(\mathbf{p}_v^t, \sigma - \eta)\|_2^2$. By construction, for all $u \in C$, $\boldsymbol{w}(u) \leq \mathbf{p}'_{v,t}(u)$. We will prove that $\|\boldsymbol{w} - \mathbf{p}'_{v,t}\|_1 \leq \sigma$, implying that $\|\mathrm{cl}(\mathbf{p}'_{v,t}, \sigma)\|_2^2 \leq \|\boldsymbol{w}\|_2^2$. This will complete the argument.

Let $D \subseteq C$ be the set of coordinates such that $\boldsymbol{q}(u) < \mathbf{p}'_{v,t}(u)$. Since $\boldsymbol{w}(u) = \min(\boldsymbol{q}(u), \mathbf{p}'_{v,t}(u))$, $\|\mathbf{p}'_{v,t} - \boldsymbol{w}\|_1 = \sum_{u \in D}[\mathbf{p}'_{v,t}(u) - \boldsymbol{q}(u)]$. Combining with the previous observations and noting that $\boldsymbol{q} = \mathrm{cl}(\mathbf{p}_v^t, \sigma - \eta)$,

$$\|\mathbf{p}'_{v,t} - \boldsymbol{w}\|_1 \leq \sum_{u \in D} [\mathbf{p}_v^t(u) + \mathbf{r}_{v,t}(u) - \boldsymbol{q}(u)] \tag{3.1}$$

$$\leq \|\mathbf{p}_v^t(u) - \boldsymbol{q}\|_1 + \sum_{u \in C} \mathbf{r}_{v,t}(u) \leq (\sigma - \eta) + \eta = \sigma \tag{3.2}$$

■

We now prove the main lemma of this section.

Proof [Proof of Lemma 3.2.1] Fix some $\ell \in \mathbb{N}$, $\ell > \alpha r^3$ and use Lemma 3.2.2 with $k = r^3\ell^6$. There exists a subset R of at most $\alpha dn/\ell^3$ edges whose removal breaks up G into connected components of size at most $r^3\ell^6$. Refer to these as AST components. Now, consider an ℓ-length walk in G starting from the stationary distribution (which is uniform). The probability that this walk encounters an edge in R at any step is exactly $|R|/2dn$. Let the random variable X_v be the number of edges of R encountered in an ℓ-length walk from v. Note that when $X_v = 0$, then the walk remains in the AST component containing v. Thus, letting E_v denote the event that walk from v leaves its AST component, we get

$$(1/n)\sum_v \Pr[E_v] \leq \mathbf{E}_{v\sim \text{u.a.r.}}[X_v] = \ell|R|/2dn \leq \alpha/(2\ell^2)$$

Since $\ell > \alpha r^3 > 4\alpha$, we can upper bound by $1/8\ell$. By the Markov bound, for at least $(1-1/\ell)n$ vertices, the probability that an ℓ-length walk starting at v encounters an edge of R and thus leaves the AST piece containing v is at most $1/8$. Denote the set of these vertices by S.

Consider any $s \in S$. Suppose it is contained in the AST component C. Note that $\|\text{cl}(\mathbf{p}'_{s,\ell}, 1/2)\|_1 \geq 1/2$. Furthermore, it has support at most $|C| \leq r^3\ell^6$. By Jensen's inequality, $\|\text{cl}(\mathbf{p}'_{s,\ell}, 1/2)\|_2^2 \geq 1(4r^3\ell^6)$. As argued earlier, the probability that a random walk (in G) from s leaves C is at most $1/8$. Applying Claim 3.2.3 for $\sigma = 1/2$ and $\eta = 1/8$, we conclude that $\|\text{cl}(\mathbf{p}_s^\ell, 1/2 - 1/8)\|_2^2 \geq 1/(4r^3\ell^6) \geq 1/\ell^7$. (For convenience, we assume that $\alpha > 4$.) ■

3.3 The existence of a discoverable decomposition

If many vertices have large clipped norms, we prove that G can be partitioned into small low conductance cuts. Furthermore, each cut can be discovered by poly(ℓ) ℓ-length random walks. The analysis follows the structure given in [6].

Lemma 3.3.1 *Let $c > 1$ be a parameter. Suppose there exists $S \subseteq V$ such that $|S| > n/\ell^{1/5}$ and $\forall s \in S, \|\mathrm{cl}(\mathbf{p}_s^\ell, 1/4)\|_2^2 > \ell^{-c}$. Then, there exists $\widetilde{S} \subseteq S$ with $|\widetilde{S}| \geq |S|/4$ such that for each $s \in \widetilde{S}$, there exists a subset $P_s \subseteq S$ where*

- $\forall v \in P_s$, $\sum_{t<16\ell^{c+1}} p_{s,t}(v) \geq 1/8\ell^{c+1}$.
- $|E(P_s, S \setminus P_s)| \leq 4d|P_s|\sqrt{c\ell^{-1/5}\log \ell}$.

A straightforward application of this lemma leads to the main decomposition theorem.

Theorem 3.3.2 *Suppose there are $\geq (1-1/\ell^{1/5})n$ vertices s such that $\|\mathrm{cl}(\mathbf{p}_s^\ell, 1/4)\|_2^2 > \ell^{-c}$. Then, there is a partition $\{P_1, P_2, \ldots, P_b\}$ of the vertices such that:*

- *For each P_i, there exists $s \in V$ such that: $\forall v \in P_i$, $\sum_{t<10\ell^{c+1}} p_{s,t}(v) \geq 1/8\ell^{c+1}$.*
- *The total number of edges crossing the partition is at most $8dn\sqrt{c\ell^{-1/5}\log \ell}$.*

Proof We simply iterate over Lemma 3.3.1. Let $T = \{s \mid \|\mathrm{cl}(\mathbf{p}_s^\ell, 1/4)\|_2^2 \leq \ell^{-c}\}$. By assumption, $|T| \leq n/\ell^{1/5}$. We keep a partition of the vertices $\{T, Q_1, Q_2, \ldots, Q_a, S\}$ with the following properties. (1) Each Q_i satisfies the first condition of the theorem. (2) In addition, we also have that the total number of edges crossing the partition is no bigger than $4d\sqrt{c\ell^{-1/5}\log \ell}\sum_{i \leq a}|Q_i| + d|T|$. We initialize with the trivial partition $\{T, S = V \setminus T\}$.

As long as $|S| > n/\ell^{1/5}$, we invoke Lemma 3.3.1. We get a new set $Q \subseteq S$ satisfying the first condition of the theorem, and the number of edges from Q to $S \setminus Q$ is at most $4d\sqrt{c\ell^{1/5}\log \ell}|Q|$. We add Q to our partition, reset $S = S \setminus Q$, and iterate.

When this process terminates, $|S| \leq n/\ell^{1/5}$. We get the final partition by removing all edges incident to $S \cup T$. Alternately, every single vertex in $S \cup T$ becomes a separate set. Note that a single vertex trivially satisfies the first condition of theorem, since for all s, $p_{s,s}(1) \geq 1/2$. The total number of edges crossing the partition is at most $4dn\sqrt{c\ell^{-1/5}\log \ell} + 2dn\ell^{-1/5} \leq 8dn\sqrt{c\ell^{-1/5}\log \ell}$. ■

3.3.1 Proving Lemma 3.3.1

An important tool used to argue about conductances within S is the projected Markov chain. These ideas come from the work of Kale-Peres-Seshadhri to analyze random walks in noisy expanders [50], and were used by the authors in their previous paper on one-sided testers for minor-freeness [6]. We closely follow the structure and notation of that paper, and explicitly mention the differences.

We define the "projection" of the random walk onto the set S. We define a Markov chain M_S, over the set S. We retain all transitions from the original random walk on G that are within S, and we denote these by $e^{(1)}_{u,v}$ for every u to v transition in the random walk on G. Additionally, for every $u, v \in S$ and $t \geq 2$, we add a transition $e^{(t)}_{u,v}$. The probability of this transition is equal to the total probability of t-length walks in G from u to v, where all internal vertices in the walk lie outside S.

Note that $e^{(t)}_{u,v} = e^{(t)}_{v,u}$. Since G is irreducible and the stationary mass on S is nonzero, all walks eventually reach S. Thus, for any u, $\sum_t \sum_v e^{(t)}_{u,v} = 1$, so M_S is a symmetric Markov chain. The stationary distribution of M_S is uniform on S.

For a transition $e^{(t)}_{u,v}$ in M_S, define the "length" of this transition to be t. For clarity, we use "hops" to denote the number of steps of a walk in M_S, and retain "length" for walks in G. The length of an h hop random walk in M_S is defined to be the sum of the lengths of the transitions it takes.

We use $\boldsymbol{\tau}_{s,h}$ to denote the distribution of the h-hop walk from s, and $\tau_{s,h}(v)$ to denote the corresponding probability of reaching v. We use $\mathcal{W}_h$ to denote the distribution of h-hop walks starting from the uniform distribution.

The following lemma is crucial for relating walks in G with M_S.

Lemma 3.3.3 (Lemma 6.4 of [61]) $\mathbf{E}_{W \sim \mathcal{W}_h}[\textit{length of } W] = hn/|S|$

We come to an important lemma. The conditions in Lemma 3.3.1 are on the clipped norms of random walks in G, but the conclusion (regarding the cut) refers to conductances within the projected Markov chain M_S. The following lemma shows that random walks in M_S must also be sufficiently trapped. This is an analogue of

Lemma 6.5 of [61], but the proof deviates significantly because of the use of clipped norms.

Lemma 3.3.4 *$\exists S' \subseteq S$, $|S'| \geq |S|/2$, such that $\forall s \in S'$, $\|\boldsymbol{\tau}_{s,\ell^{1/5}}\|_\infty \geq 1/2\ell^{c+1}$.*

Proof Consider ℓ-length random walks in G starting from $s \in S$. For any such walk, we can define the number of hops it makes as the number of vertices in S encountered minus one.

For $h \in \mathbb{N}$ and $s \in S$, define the event $\mathcal{E}_{s,h}$ that an ℓ-length walk from s makes h hops. We will further split this event into $\mathcal{F}_{s,h}$, when the walk ends at S, and $\mathcal{G}_{s,h}$, when the walk does not end at S. A walk that ends in S directly corresponds to an h-hop walk in M_S. By Lemma 3.3.3, $|S|^{-1}\sum_{s\in S}\Pr[\mathcal{F}_{s,h}]\ell \leq hn/|S|$. Consider any walk in the event $\mathcal{G}_{s,h}$. If one continued until it ends in S, this gives a walk in M_S with a single additional hop (and a longer length). Thus, the total probability mass $\Pr[\mathcal{G}_{s,h}]$ corresponds to walks in M_S that make $(h+1)$ hops and have length at least ℓ. By Lemma 3.3.3 again, $|S|^{-1}\sum_{s\in S}\Pr[\mathcal{G}_{s,h}]\ell \leq (h+1)n/|S|$.

Summing these bounds and applying the size bound on S,

$$\frac{\sum_{s\in S}\Pr[\mathcal{E}_{s,h}]}{|S|}\ell \leq (2h+1)n/|S| \leq \ell^{1/5}(2h+1) \Longrightarrow \frac{\sum_{s\in S}\Pr[\mathcal{E}_{s,h}]}{|S|} \leq \ell^{-4/5}(2h+1)$$

Now, we sum over h and use the fact that ℓ is a sufficiently large constant.

$$|S|^{-1}\sum_{h\leq \ell^{1/5}}\sum_{s\in S}\Pr[\mathcal{E}_{s,h}] \leq \ell^{-4/5}\sum_{h\leq \ell^{1/5}}(2h+1) \leq 4\ell^{-2/5} < 1/10$$

By the Markov bound, there is a set S', $|S'| \geq |S|/2$ such that $\forall s \in S'$, it holds that $\sum_{h\leq \ell^{1/5}}\Pr[\mathcal{E}_{s,h}] < 1/5$.

For $v \in V$, let $y_s(v)$ be the probability that an ℓ-length walk from s to v makes at most $\ell^{1/5}$ hops. Note that $\sum_{v\in S} y_s(v) \leq \sum_{h\leq\ell^{1/5}}\Pr[\mathcal{E}_{s,h}] < 1/5$. We now use the clipped norm definition. Since $\|\mathrm{cl}(\mathbf{p}^\ell_s, 1/4)\|_2^2 \geq \ell^{-c}$, $\sum_{v\in V}(p_{s,\ell}(v) - y_s(v))^2 \geq \ell^{-c}$. This is important, since we can "remove" the low hop walks and still have a large norm.

Consider the probability α that a 2ℓ-length walk from s back to s makes at least $\ell^{1/5}$ hops. (Note that this corresponds to walks in M_S.) Clearly, any walk going from s to v in an ℓ-length walk making at least $\ell^{1/5}$ hops and then returning to s in an ℓ-length walk contributes to this probability. Thus, we can lower bound α by $\sum_{v \in V}(p_{s,\ell}(v) - y_s(v))^2 \geq \ell^{-c}$. Note that all walks considered make at most 2ℓ hops.

Thus, $\sum_{h \geq \ell^{1/5}}^{2\ell} \|\boldsymbol{\tau}_{s,\ell^{1/5}}\|_\infty \geq \ell^{-c}$. Since the infinity norm is non-increasing in hops, by averaging, $\|\boldsymbol{\tau}_{s,\ell^{1/5}}\|_\infty \geq 1/2\ell^{c+1}$. ■

The remaining proof of Lemma 3.3.1 is almost identical to analogous calculations in Section 6 of [61]. Therefore, we move it to the end of this chapter in §3.5

3.4 Proof of main result

Before we show Theorem 3.1.1 and Theorem 3.1.2, we argue about the guarantees of `EstClip`. The proofs of the next two claims are relatively routine concentration arguments. Recall that T is the vertex set constructed in a call to `EstClip`(s).

Claim 3.4.1 *Consider any vertex s. With probability at least $1 - 2^{-1/\varepsilon^2}$ over the randomness in* `EstClip`(s)*: all v such that $\mathbf{p}_s^\ell(v) \geq 1/\ell^7$ are in T, and no v such that $\mathbf{p}_s^\ell(v) \leq 1/\ell^8$ is in T.*

Proof Consider v such that $\mathbf{p}_s^\ell(v) \geq 1/\ell^7$. Recall that the total number of walks is $w = \ell^{14}$. The expected value of w_v is at least $\ell^{14}/\ell^7 = \ell^7$. Note that w_v is a sum of Bernoulli random variables. By a multiplicative Chernoff bound (Theorem 1.1 of [62]), $\Pr[w_v \leq \ell^7/2] \leq \exp(-\ell^7/8)$. There are at most ℓ^7 such vertices v. By a union bound over all of them, the probability that some such v is not in T is at most $\ell^7 \cdot \exp(-\ell^7/8) \leq \exp(-\ell^6) \leq 2^{-2/\varepsilon^2}$. (Note that $\ell > \varepsilon^{-20}$.) This proves the first part.

For the second part, consider v such that $\mathbf{p}_s^\ell(v) \leq 1/\ell^8$. We split into two cases.

Case 1, $\mathbf{p}_s^\ell(v) \geq \exp(-\ell/2)$. The expectation of w_v is at most $\ell^{14}/\ell^8 = \ell^6$. Since $\ell^7/2 \geq 2e\ell^6$, by a Chernoff bound (third part, Theorem 1.1 of [62]), $\Pr[w_v \geq \ell^7/2] \leq 2^{-\ell^7/2}$. There are at most $\exp(\ell/2)$ such vertices v. Taking a union bound over all of them, the probability that any such vertex appears in T is at most $\exp(\ell/2)2^{-\ell^7/2} \leq 2^{-\ell^5} \leq 2^{-2/\varepsilon^2}$.

Case 2, $\mathbf{p}_s^\ell(v) < \exp(-\ell/2)$. For convenience, set $p = \mathbf{p}_s^\ell(v)$. The probability that $w_v \leq 1$ is:

$$(1-p)^w + wp(1-p)^{w-1} \geq (1-wp) + wp(1-p(w-1)) = 1-p^2w(w-1) \geq 1-p^2w^2 \quad (3.3)$$

(We use the inequality $(1-x)^r \geq 1-xr$, for $|x| \leq 1, r \in \mathbb{N}$.) Thus, the probability that $w_v > 1$ is at most p^2w^2. Note that $\ell^7/2$ (the threshold to be placed in T) is at least 2.

Let us take a union bound over all such vertices. We note that $w = \ell^{14}$ and $\ell > \varepsilon^{-20}$. The probability that any such v is placed in T is at most

$$\sum_{v:\mathbf{p}_s^\ell(v)<\exp(-\ell/2)} \mathbf{p}_s^\ell(v)^2 w^2 \leq \ell^{28}\exp(-\ell/2)\sum_v \mathbf{p}_s^\ell(v) \leq \exp(-1/\varepsilon^2) \quad (3.4)$$

We union bound over all errors to complete the proof.

■

We can now argue about the main guarantee of `EstClip`.

Claim 3.4.2 *For all vertices s, with probability at least $1-2^{-1/\varepsilon}$ over the randomness of* `EstClip`(s)*:*

- *If $\|\mathrm{cl}(\mathbf{p}_s^\ell, 1/4)\|_2^2 < \ell^{-8}/400$, then* `EstClip`$(s)$ *outputs LOW.*
- *If $\|\mathrm{cl}(\mathbf{p}_s^\ell, 3/8)\|_2^2 > \ell^{-7}$, then* `EstClip`$(s)$ *outputs HIGH.*

Proof Consider the first case. Let $H = \{v \mid \mathbf{p}_s^\ell(v) \geq \ell^{-8}\}$. We first argue that $\sum_{v\in H}\mathbf{p}_s^\ell(v) \leq 1/4+1/20$. Suppose not. Then, any clipping of 1/4 of the probability mass of $\mathbf{p}_s^\ell$ leaves at least 1/20 probability mass on H. The size of H is at most ℓ^8. By Jensen's inequality, $\|\mathrm{cl}(\mathbf{p}_s^\ell, 1/4)\|_2^2 \geq 1/400\ell^8$, contradicting the case condition.

Thus, $\sum_{v\in H}\mathbf{p}_s^\ell(v) \leq 1/4+1/20$. The expected value of $\sum_{v\in H} w_v \leq w(1/4+1/20)$. By an additive Chernoff bound (first part, Theorem 1.1 of [62]), we get $\Pr[\sum_{v\in H} w_v \geq w/3] \leq \exp(-2(1/3-1/4-1/20)^2 w) \leq \exp(-\ell^{13})$. By Claim 3.4.1, with probability at least $1-2^{-1/\varepsilon^2}$, $T \subseteq H$. By a union bound, with probability at least $1-2^{-1/\varepsilon}$, $\sum_{v\in T} w_v \leq \sum_{v\in H} w_v < w/3$, and the output is LOW.

Now for the second case. Let $H' = \{v \,|\mathbf{p}_s^\ell(v) \geq \ell^{-7}\}$. We will show that $\sum_{v \in H} \mathbf{p}_s^\ell(v) \geq 3/8$. Suppose not. We can clip away all the probability mass of $\mathbf{p}_s^\ell$ that is on H, which is at most 3/8. All remaining probability/entries of the clipped vector are at most ℓ^{-7}. Thus, the squared l_2-norm is at most ℓ^{-7}, implying $\|\mathrm{cl}(\mathbf{p}_s^\ell, 3/8)\|_2^2 \leq \ell^{-7}$ (contradiction).

Thus, $\sum_{v \in H'} \mathbf{p}_s^\ell(v) \geq 3/8$. By an additive Chernoff bound (first part, Theorem 1.1 of [62]), we get $\Pr[\sum_{v \in H} w_v < w/3] \leq \exp(-2(3/8 - 1/3)^2 w) \leq \exp(-\ell^{13})$. By Claim 3.4.1, with probability at least $1 - 2^{-1/\varepsilon^2}$, $H' \subseteq T$. By a union bound, with probability at least $1 - 2^{-1/\varepsilon}$, $\sum_{v \in T} w_v \geq \sum_{v \in H'} w_v \geq w/3$, and the output is HIGH. ■

We now prove completeness, Theorem 3.1.1. We will prove that if G is H-minor-free, then the tester `FindMinor` accepts with probability $> 2/3$. This follows almost directly from Lemma 3.2.1.

Proof [Proof of Theorem 3.1.1] Suppose G is H-minor-free. Note that calls to `LocalSearch` can never return FOUND, so rejection can only happen because of the output of calls to `EstClip`.

By Lemma 3.2.1, there are at least $(1 - 1/\ell)n$ vertices such that $\|\mathrm{cl}(\mathbf{p}_s^\ell, 3/8)\|_2^2 \geq \ell^{-7}$. Call these vertices *heavy*. The expected number of light vertices in the multiset S chosen in Step 1 of `FindMinor` is at most $1/\ell \times \ell^{21} = \ell^{20}$. By a multiplicative Chernoff bound (Theorem 1 of [62]), the number of light vertices in S is strictly less than $2\ell^{20}$ with probability at least $1 - \exp(-\ell^{19}) > 9/10$. Let us condition on this event. The probability that any call to `EstClip`(s) returns HIGH for a heavy $s \in S$ is at least $1 - 2^{-1/\varepsilon}$, by Claim 3.4.2. By a union bound over the at most ℓ^{21} heavy vertices in S, all calls to `EstClip`(s) for heavy $s \in S$ return HIGH with probability at least $1 - \ell^{21} 2^{-1/\varepsilon} > 9/10$.

We now remove the conditioning. With probability $> (9/10)^2 > 2/3$, there are strictly less than $2\ell^{18}$ calls (for the light vertices) that return LOW. When this happens, `FindMinor` accepts. ■

Now we prove soundness, Theorem 3.1.2. We prove that if G is ε-far from H-minor-freeness, the tester rejects with probability $> 2/3$. The main ingredient is the decomposition of Theorem 3.3.2.

Proof [Proof of Theorem 3.1.2] Assume G is ε-far from being H-minor free. We split into two cases.

Case 1: There are less than $(1 - 1/\ell^{1/5})n$ vertices such that $\|\mathrm{cl}(\mathbf{p}_s^\ell, 1/4)\|_2^2 > \ell^{-9}$.

Then, there are at least $n/\ell^{1/5}$ vertices such that $\|\mathrm{cl}(\mathbf{p}_s^\ell, 1/4)\|_2^2 \leq \ell^{-9}$. The expected number of such vertices (with repetition) in the multiset S (of Step 1) is at least $\ell^{21}/\ell^{1/5}$. By a multiplicative Chernoff bound, there are at least $\ell^{21}/2\ell^{1/5} > 2\ell^{20}$ such vertices in S, with probability at least $1 - \exp(-\ell^{20}/4)$. For each such vertex s, the probability that $\texttt{EstClip}(s)$ outputs LOW is at least $1 - 2^{-1/\varepsilon}$ (Claim 3.4.2). By a union bound over all vertices in S, with probability $> (1 - \exp(-\ell^{20}))(1 - \ell^{21}2^{-1/\varepsilon}) > 5/6$, there are at least $2\ell^{20}$ calls to $\texttt{EstClip}(s)$ that return LOW. So the tester rejects.

Case 2: There are at least $(1 - 1/\ell^{1/5})n$ vertices such that $\|\mathrm{cl}(\mathbf{p}_s^\ell, 1/4)\|_2^2 > \ell^{-9}$.

We apply the decomposition of Theorem 3.3.2 (with $c = 9$). There is a partition $\{P_1, P_2, \ldots, P_b\}$ of the vertices such that:

- For each P_i, there exists $s \in V$ such that: $\forall v \in P_i$, $\sum_{t<10\ell^{10}} p_{s,t}(v) \geq 1/8\ell^{10}$. Call s the *anchor* for P_i, noting that multiple sets may have the same anchor.
- The total number of edges crossing the partition is at most $24dn\sqrt{\ell^{-1/5}\log \ell}$.

Among the sets in the partition, let $\{Q_1, Q_2, \ldots, Q_a\}$ be the sets of vertices that contain an H-minor (or technically, the subgraphs induced by these sets contain an H-minor). Note that one can remove $d\sum_{i\leq a}|Q_i| + 24dn\sqrt{\ell^{-1/5}\log \ell}$ edges to make G H-minor-free. Since $\ell > \varepsilon^{-20}$, $24dn\sqrt{\ell^{-1/5}\log \ell} \leq \varepsilon nd/2$. Since G is ε-far from being H-minor free, we deduce from the above that $\sum_{i\leq a}|Q_i| \geq \varepsilon n/2$.

Let $Z = \{s \mid s \text{ is anchor for some } Q_i\}$. Let us lower bound $|Z|$. For every Q_i, there is some $s \in Z$ such that $\forall v \in Q_i$, $\sum_{t<10\ell^{10}} p_{s,t}(v) \geq 1/8\ell^{10}$. Thus, for every Q_i, there is some $s \in Z$ such that $\sum_{v \in Q_i} \sum_{t < 10\ell^{10}} p_{s,t}(v) \geq |Q_i|/8\ell^{10}$. Let us sum over all $s \in Z$ (and note that $\sum_{v \in V} p_{s,t}(v) = 1$).

$$\sum_{i \leq a} |Q_i|/8\ell^{10} \leq \sum_{s \in Z} \sum_{v \in V} \sum_{t < 10\ell^{10}} p_{s,t}(v) \leq \sum_{t < 10\ell^{10}} \sum_{s \in Z} \sum_{v \in V} p_{s,t}(v) \leq 10\ell^{10}|Z| \tag{3.5}$$

Since $\sum_{i \leq a} |Q_i| \geq \varepsilon n/2$, $|Z| \geq \varepsilon n/160\ell^{20} \geq 5n/\ell^{21}$.

Focus on the multiset S in Step 1 of `FindMinor`. Note that S contains an element of Z with probability $\geq 1 - (1 - 5/\ell^{21})^{\ell^{21}} \geq 9/10$. Let us condition of this event, and let $s \in S \cap Z$. There exists some Q_i such that $\forall v \in Q_i$, $\sum_{t < 10\ell^{10}} p_{s,t}(v) \geq 1/8\ell^{10}$. By averaging over walk length, $\forall v \in Q_i$, $\exists t < 10\ell^{10}$ such that $p_{s,t}(v) \geq 1/80\ell^{20}$.

Now, consider the call to `LocalSearch`(s). The set B_s in Step 1 of `LocalSearch` is constructed by performing ℓ^{21} random walks of length ℓ^{11}. For any $v \in Q_i$, the probability that v is in B_s is at least $1 - (1 - 1/80\ell^{20})^{\ell^{21}} \geq 1 - \exp(-\ell/80)$. Taking a union bound over all $v \in Q_i$, the probability that $Q_i \subseteq B_s$ is at least $1 - \ell^{21}\exp(-\ell/80) \geq 9/10$. When $Q_i \subseteq B_s$, then $G[B_s]$ contains an H-minor and the tester rejects. The probability of this happening is at least $(9/10)^2 > 2/3$. ■

3.5 Local Partitioning Proofs

3.6 Local partitioning, and completing the proof of Lemma 3.3.1

We perform local partitioning on M_S, starting with an arbitrary $s \in S'$. We apply the Lovász-Simonovits curve technique. (The definitions are originally from [3]. Refer to Lecture 7 of Spielman's notes [59] as well as Section 2 in Spielman-Teng [54]. This is also a restatement of material in Section 6.1 of [61], which is needed to state the main lemma.)

- Conductance: for some $T \subseteq S$ we define the conductance of T in M_S to be

$$\Phi(T) = \frac{\sum_{\substack{u \in T \\ v \in S \setminus T}} \tau_{u,1}(v)}{\min\{|S \setminus T|, |T|\}}$$

- Ordering of states at time t: At time t, let us order the vertices in M_S as $v_1^{(t)}, v_2^{(t)}, \ldots$ such that $\tau_{s,t}(v_1^{(t)}) \geq \tau_{s,t}(v_2^{(t)}) \ldots$, breaking ties by vertex id. At $t = 0$, we set $\tau_{s,0}(s) = 1$, and all other values to 0.

- The LS curve h_t: We define a function $h_t : [0, |S|] \to [0, 1]$ as follows. For every $k \in [|S|]$, set $h_t(k) = \sum_{j \leq k} \tau_{s,t}(v_j^{(t)})$. (Set $h_t(0) = 0$.) For every $x \in (k, k+1)$, we linearly interpolate to construct $h(x)$. Alternately, $h_t(x) = \max_{\vec{w} \in [0,1]^{|S|}, \|\vec{w}\|_1 = x} \sum_{v \in S} [\tau_{s,t}(v) - 1/n] w_i$.

- Level sets: For $k \in [0, |S|]$, we define the (k, t)-level set,

$$L_{k,t} = \{v_1^{(t)}, v_2^{(t)}, \ldots, v_k^{(t)}\}.$$

  The *minimum probability* of $L_{k,t}$ denotes $\tau_{s,t}(v_k^{(t)})$.

The main lemma of Lovász-Simonovits is the following (Lemma 1.4 of [3], also refer to Theorem 7.3.3 of Lecture 7 in [59]).

Lemma 3.6.1 *For all k and all t,*

$$h_t(k) \leq \frac{1}{2}[h_{t-1}(k - 2\min(k, n-k)\Phi(L_{k,t})) + h_{t-1}(k + 2\min(k, n-k)\Phi(L_{k,t}))]$$

We employ this lemma to prove a condition of the level set conductances. An analogous lemma was proven in [61] for specific parameters. We redo the calculation here.

Lemma 3.6.2 *Suppose there exists $\phi \in [0,1]$ and $p > 2/n$ such that for all $t' \leq t$ it is true that for all $k \in [n]$ that if $L_{k,t'}$ has a minimum probability of at least p, then $\Phi(L_{k,t}) \geq \phi$. Then for all $k \in [0, n]$, $h_t(k) \leq \sqrt{k}(1 - \phi^2/2)^t + pk$.*

Proof We will prove by induction over t. For the base case, consider $t = 0$. The RHS is at least 1, proving the bound.

Now for the induction. Note that h_t is a concave, and the RHS is also concave. Thus, it suffices to prove the bound for the integer points ($h_t(k)$ for integer k). If

$k \geq 1/p$, then the RHS is at least 1. Thus the bound is trivially true. Let us assume that $k < 1/p < n/2$. We now split the proof into two cases based on the conductance of $L_{k,t}$.

First let us consider the case where $\Phi(L_{k,t}) \geq \phi$. By Lemma 3.6.1 and concavity of h,

$$
\begin{aligned}
h_t(k) &\leq \frac{1}{2}\Big(h_{t-1}\big(k(1-2\phi)\big) + h_{t-1}\big(k(1+2\phi)\big)\Big) && (3.6)\\
&\leq \frac{1}{2}\Big(\sqrt{k(1-2\phi)}(1-\phi^2/2)^{t-1} + \sqrt{k(1+2\phi)}(1-\phi^2/2)^{t-1} + 2kp\Big) && (3.7)\\
&\leq \frac{1}{2}\Big(\sqrt{k}\left(1-\phi^2/2\right)^{t-1}\left(\sqrt{1-2\phi}+\sqrt{1+2\phi}\right) + 2kp\Big) && (3.8)\\
&\leq \sqrt{k}\left(1-\phi^2/2\right)^{t} + kp && (3.9)
\end{aligned}
$$

For the last inequality we use the bound $\left(\sqrt{1+z}+\sqrt{1-z}\right)/2 \leq 1 - z^2/8$.

Now we deal with the case when $\Phi(L_{k,t}) < \phi$. By assumption, $L_{k,t}$ has minimum probability less than p. Let $k' < k$ be the largest index such that $L_{k',t}$ has minimum probability at least p. Note that $\Phi(L_{k',t}) \geq \phi$. Therefore, as proven in the first case, $h_t(k') \leq \sqrt{k'}\,(1-\phi^2/2)^t + k'p$. Every vertex we add to $L_{k',t}$ adds less than p probability mass to $L_{k',t}$, and therefore, by the concavity of $h_t(x)$,

$$
\begin{aligned}
h_t(k) &\leq h_t(k') + (k-k')p && (3.10)\\
&\leq \sqrt{k'}\left(1-\phi^2/2\right)^t + k'p + (k-k')p && (3.11)\\
&\leq \sqrt{k'}\left(1-\phi^2/2\right)^t + kp \leq \sqrt{k}\left(1-\phi^2/2\right)^t + kp && (3.12)
\end{aligned}
$$

■

For convenience, we restate Lemma 3.3.1.

Lemma 3.6.3 *Let $c > 1$ be a parameter. Suppose there exists $S \subseteq V$ such that $|S| > n/\ell^{1/5}$ and $\forall s \in S, \|\mathrm{cl}(\mathbf{p}_s^\ell, 1/4)\|_2^2 > \ell^{-c}$. Then, there exists $\widetilde{S} \subseteq S$ with $|\widetilde{S}| \geq |S|/4$ such that for each $s \in \widetilde{S}$, there exists a subset $P_s \subseteq S$ where*

- $\forall v \in P_s$, $\sum_{t<16\ell^{c+1}} p_{s,t}(v) \geq 1/8\ell^{c+1}$.

- $|E(P_s, S \setminus P_s)| \leq 4d|P_s|\sqrt{c\ell^{-1/5}\log \ell}$.

Proof By Lemma 3.3.4, there is a set $S' \subseteq S$, $|S'| \geq |S|/2$ such that for all $s \in S'$, $\|\boldsymbol{\tau}_{s,\ell^{1/5}}\|_\infty \geq 1/2\ell^{c+1}$. Consider any $s \in S'$.

Suppose for all $t' \leq \ell^{1/5}$, all level sets $L_{k,t'}$ with minimum probability $1/2\ell^{c+1}$ have conductance at least $\sqrt{4c\ell^{-1/5}\log \ell}$. Lemma 3.6.2 implies that $\|\boldsymbol{\tau}_{s,\ell^{1/5}}\|_\infty = h_{\ell^{1/5}}(1) \leq (1-2c\ell^{-1/5}\log \ell)^{\ell^{1/5}} + 1/4\ell^{c+1} < 1/4\ell^{c+1} + 1/4\ell^{c+1} = 1/2\ell^{c+1}$. This is a contradiction.

Thus, for every $s \in S'$, there exists a level set denoted P_s with minimum probability $1/2\ell^{c+1}$ and conductance at most $\sqrt{4c\ell^{-1/5}\log \ell}$. Note that $|P_s| \leq 2\ell^{c+1} < |S|/2$.

$$\sqrt{4c\ell^{-1/5}\log \ell} \geq \Phi(P_s) = \frac{\sum_{\substack{x\in P_s \\ y \in S\setminus P_s}} \tau_{x,1}(y)}{\min(|P_s|, |S \setminus P_s|} \geq \frac{E(P_s, S\setminus P_s)}{2d|P_s|} \tag{3.13}$$

The inequality is obtained by only considering transitions from S to $S \setminus P_s$ that come from a single edge in G. Each such edge has a traversal probability of $1/2d$. Therefore, $E(P_s, S \setminus P_s) \leq 4d|P_s|\sqrt{c\ell^{-1/5}\log \ell}$.

Set $L = 8\ell^{c+2}$. Define $\widetilde{S} \subseteq S'$ to be the vertices $s \in S'$ with the property that $\forall v \in P_s$, $\sum_{l<L} p_{s,v}(l) \geq 1/8\ell^{c+1}$. Together with the cut bound above, this clearly satisfies the conditions on the lemma. It remains the prove a suitable upper bound of $|S' \setminus \widetilde{S}|$, to show that $\widetilde{S}$ is sufficiently large.

For every $s \in S' \setminus \widetilde{S}$, there exists $v_s \in P_s$ such that $\sum_{l<L} p_{s,l}(v) < 1/8\ell^{c+1}$. Let $\hat{p}_{s,l}(v)$ denote that probability that an $\ell^{1/5}$-hop walk in M_S from s reaches v with length l. Consider $s \in S' \setminus \widetilde{S}$.

$$\tau_{s,\ell^{1/5}}(v_s) = \sum_{l \geq \ell^{1/5}} \hat{p}_{s,l}(v_s) = \sum_{l\geq \ell^{1/5}}^{L-1} \hat{p}_{s,l}(v_s) + \sum_{l \geq L} \hat{p}_{s,l}(v_s) \leq \sum_{l\geq \ell^{1/5}}^{L-1} p_{s,l}(v) + \sum_{l\geq L} \hat{p}_{s,l}(v) \tag{3.14}$$

Since the minimum probability of P_s is at least $1/4\ell^{c+1}$, $\tau_{s,\ell^{1/5}}(v_s) \geq 1/4\ell^{c+1}$. We argued above that $\sum_{l\geq \ell^{1/5}}^{L-1} p_{s,l}(v) \leq \sum_{l<L} p_{s,l}(v) < 1/8 \leq^{c+1}$. We conclude that

$\sum_{l>L} \hat{p}_{s,l}(v) \geq 1/8\ell^{c+1}$. Note that all of this probability mass corresponds to $\ell^{1/5}$-hop walks that have a large length. We now lower bound $\mathbf{E}_{W\sim\mathcal{W}_{\ell^{1/5}}}$[length of W].

$$\mathbf{E}_{W\sim\mathcal{W}_{\ell^{1/5}}}[\text{length of } W] \geq \frac{1}{|S|}\sum_{s\in S'\setminus\widetilde{S}}\Big(\sum_{l>L}\hat{p}_{s,l}(v_s)\Big)L \geq \frac{|S'\setminus\widetilde{S}|}{|S|}\cdot\frac{L}{8\ell^{c+1}} \geq \frac{\ell|S'\setminus\widetilde{S}|}{|S|} \tag{3.15}$$

By Lemma 3.3.3, $\mathbf{E}_{W\sim\mathcal{W}_{\ell^{1/5}}}$[length of W] $= \ell^{1/5}n/|S|$. Combining, $|S'\setminus\widetilde{S}| \leq n/\ell^{4/5} \leq n/4\ell^{1/5} \leq |S|/4$. By Lemma 3.3.4, $|S'| \geq |S|/2$. By the setting of Lemma 3.3.1, $|S| > n/\ell^{1/5}$. Thus, $|S'\setminus\widetilde{S}| \leq n/4\ell^{1/5}$, and $|\widetilde{S}| \geq |S|/4$. ■

4 AN EFFICIENT PARTITION ORACLE FOR BOUNDED DEGREE GRAPHS

The algorithmic study of planar graphs is a fundamental direction in theoretical computer science and graph theory. Classic results like the Kuratowski-Wagner characterization [29, 30], linear time planarity algorithms [28], and the Lipton-Tarjan separator theorem underscore the significance of planar graphs [63]. The celebrated theory of Robertson-Seymour give a grand generalization of planar graphs through minor-closed families [15, 16, 31]. This has led to many deep results in graph algorithms, and an important toolkit is provided by separator theorems and associated decompositions [64].

Over the past decade, there have been many advances in *sublinear* algorithms for planar graphs and minor-closed families. We focus on the model of random access to bounded degree adjacency lists, introduced by Goldreich-Ron [33]. Let $G = (V, E)$ be a graph with vertex set $V = [n]$ and degree bound d. The graph is accessed through *neighbor queries*: there is an oracle that on input $v \in V$ and $i \in [d]$, returns the ith neighbor of v. (If none exist, it returns $\perp$.)

One of the key properties of bounded-degree graphs in minor-closed families is that they exhibit hyperfinite decompositions. A graph G is hyperfinite if $\forall\ 0 < \varepsilon < 1$, one can remove εdn edges from G and obtain connected components of size independent of n (we refer to these as pieces). For minor-closed families, one can remove εdn edges and get pieces of size $O(\varepsilon^{-2})$.

The seminal result of Hassidim-Kelner-Nguyen-Onak (HKNO) [41] introduced the notion of *partition oracles*. This is a local procedure that provides "constant-time" access to a hyperfinite decomposition. The oracle takes a query vertex v and outputs the piece containing v. Each piece is of size independent of n, and at most εdn edges go between pieces. Furthermore, all the answers are consistent with a single hyperfinite decomposition, despite there being no preprocessing or explicit coordination.

(All queries uses the same random seed, to ensure consistency.) Partition oracles are extremely powerful as they allow a constant time procedure to directly access a hyperfinite decomposition. As observed in previous work, partition oracles lead to a plethora of property testing results and sublinear time approximation algorithms for minor-closed graph families [41,60]. In some sense, one can think of partition oracles as a moral analogue of Szémeredi's regularity lemma for dense graph property testing: it is a decomposition tool that immediately yields a litany of constant time (or constant query) algorithms. We also note that the breakthrough result of Benjamini-Schramm-Shapira that gave the first property testers for planarity implicitly yields partition oracles [32].

We give a formal definition of partition oracles.

Definition 4.0.1 *Let $\mathcal{P}$ be a family of graphs. A procedure $\boldsymbol{A}$ is an $(\varepsilon, t(\varepsilon))$-*partition oracle *for $\mathcal{P}$ if it satisfies the following properties. The deterministic procedure takes as input random access to $G = (V, E)$ in $\mathcal{P}$, random access to a random seed r (of length polynomial in graph size), a proximity parameter $\varepsilon > 0$, and a vertex v of G. (We will think of fixing G, r, ε, so denote the procedure $\boldsymbol{A}_{G,r,\varepsilon}$. All probabilities are with respect to r.) The procedure $\boldsymbol{A}_{G,r,\varepsilon}(v)$ outputs a set of vertices.*

1. *(Consistency) The sets $\{\boldsymbol{A}_{G,r,\varepsilon}(v)\}$, over all v, form a partition of V.*
2. *(Size bound) For every v, $|\boldsymbol{A}_{G,r,\varepsilon}(v)| \leq t(\varepsilon)$.*
3. *(Cut bound) With probability (over r) at least $2/3$, the number of edges between the sets $\boldsymbol{A}_{G,r,\varepsilon}(v)$ is at most εdn.*

We stress that there is no explicit "coordination" or sharing of state between calls to $\boldsymbol{A}_{G,r,\varepsilon}(v)$ and $\boldsymbol{A}_{G,r,\varepsilon}(v')$ (for $v \neq v'$). There is no global preprocessing step once the random seed is fixed. The consistency guarantee holds with probability 1.

The challenge in partition oracles is to bound $t(\varepsilon)$. HKNO gave a partition oracle with $t(\varepsilon) = (d\varepsilon^{-1})^{\mathrm{poly}(d\varepsilon^{-1})}$. Levi-Ron [42] built on the ideas from HKNO and dramatically improved the bound to $t(\varepsilon) = (d\varepsilon^{-1})^{\log(d\varepsilon^{-1})}$. Yet, all minor-closed families are $(\varepsilon, \Theta(\varepsilon^{-2}))$-hyperfinite, which is quite far from the bounds of partition oracles. HKNO raise the natural open question as to whether $(\varepsilon, \text{poly}(d\varepsilon^{-1}))$-partition oracles exist.

In this paper, we resolve this open problem.

Theorem 4.0.1 *Let $\mathcal{P}$ be the set of d-bounded degree graphs in a minor-closed family. There is an $(\varepsilon, \text{poly}(d\varepsilon^{-1}))$-partition oracle for $\mathcal{P}$.*

We also note that in our definition of partition oracles, we enforce the size bound with probability 1, whereas in previous results, the size bound fails with non-zero probability.

4.0.1 Consequences

As observed by HKNO and Newman-Sohler [60], partition oracles have many consequences for property testing and sublinear algorithms.

Recall the definition of property testers. Let $\mathcal{Q}$ be a property of graphs with degree bound d. The distance of G to $\mathcal{Q}$ is the minimum number of edge additions/removals required to make G have $\mathcal{Q}$, divided by dn. A property tester for $\mathcal{P}$ is a randomized procedure that takes query access to an input graph G and a proximity parameter, $\varepsilon > 0$. If $G \in \mathcal{P}$, the tester accepts with probability at least $2/3$. If the distance of G to $\mathcal{Q}$ is at least ε, the tester rejects with probability at least $2/3$. We often measure the query complexity as well as time complexity of the tester.

A direct consequence of Theorem 4.0.1 is an "efficient" analogue of a theorem of Newman-Sohler, which states that all properties of hyperfinite graphs are testable.

Theorem 4.0.2 *Let $\mathcal{Q}$ be any property of bounded degree graphs of a minor-closed family. There exists a $\text{poly}(d\varepsilon^{-1})$-query tester for $\mathcal{Q}$.*

If membership in $\mathcal{Q}$ can be determined exactly in polynomial (in input size) time, then $\mathcal{Q}$ has $\text{poly}(d\varepsilon^{-1})$-time testers.

An appealing consequence of Theorem 4.0.2 is that the property of bipartite planar graphs can be tested in $\mathrm{poly}(d\varepsilon^{-1})$ time. For any fixed subgraph H, the property of H-free planar graphs can be tested in the same time. And all of these bounds hold for any minor-closed family.

As observed by Newman-Sohler, partition oracles give sublinear query algorithms for any graph parameter that is "robust" to edge changes. Again, Theorem 4.0.1 implies an efficient version for minor-closed families.

Theorem 4.0.3 *Let f be a real-valued function on graphs that changes by $O(1)$ on edge addition/removals, and has the property that $f(G_1 \cup G_2) = f(G_1) + f(G_2)$ for graphs G_1, G_2 that are not connected to each other.*

For any minor-closed family $\mathcal{P}$, there is a randomized algorithm that, given $\varepsilon > 0$ and $G \in \mathcal{P}$, outputs an additive εn-approximation to $f(G)$ and makes $\mathrm{poly}(d\varepsilon^{-1})$ *queries. If f can be computed exactly in polynomial time, then the above algorithm runs in* $\mathrm{poly}(d\varepsilon^{-1})$ *time.*

The functions captured by Theorem 4.0.3 are quite general. Functions such as maximum matching, minimum vertex cover, maximum independent set, minimum dominating set, maxcut, etc. all have the robustness property. As a compelling application of Theorem 4.0.3, we can get $(1+\varepsilon)$-approximations[1] for the maximum matching in planar (or any minor-closed family) graphs in $\mathrm{poly}(d\varepsilon^{-1})$ time.

These theorems are easy consequences of Theorem 4.0.1. Using the partition oracle, an algorithm can essentially assume that the input is a collection of connected components of size $\mathrm{poly}(d\varepsilon^{-1})$, and run an exact algorithm on a collection of randomly sampled components. We give formal proofs in §4.5.

[1]The maximum matching is $\Omega(n/d)$ for a connected bounded degree graph. One simply sets $\varepsilon \ll 1/d$ in Theorem 4.0.3.

4.0.2 Related work

The subject of property testing and sublinear algorithms in bounded degree graphs is a vast topic. We refer the reader to Chapters 9 and 10 of Goldreich's textbook [38]. We focus on the literature relevant to sublinear algorithms for minor-closed families.

The first step towards a characterization of testable properties in the bounded-degree model was given by Czumaj-Sohler-Shapira, who showed hereditary properties in non-expanding graphs are testable [40]. This was an indication that notions like hyperfiniteness are connected to property testing. Benjamini-Schramm-Shapira achieved a breakthrough by showing that all minor-closed properties are testable, in time triply-exponential in $d\varepsilon^{-1}$ [32]. Hassidim-Kelner-Nguyen-Onak introduced partition oracles, and designed one running in time $\exp(d\varepsilon^{-1})$. Levi-Ron improved this bound to quasipolynomial in $d\varepsilon^{-1}$, using a clever analysis inspired by algorithms for minimum spanning trees [42]. Newman-Sohler built on partition oracles for minor-close families to show that all properties of hyperfinite graphs are testable.

There are two dominant combinatorial ideas in this line of work. The first is using subgraph frequencies in neighborhood of radius $\mathrm{poly}(\varepsilon^{-1})$ to characterize properties. This naturally leads to exponential dependencies in $\mathrm{poly}(\varepsilon^{-1})$. The second idea is to use random edge contractions to reduce the graph size. Recursive applications lead to hyperfinite decompositions, and the partition oracles of HKNO and Levi-Ron simulate this recursive procedure. This is extremely non-trivial, and leads to a recursive local procedure with a depth dependent of ε. Levi-Ron do a careful simulation, ensuring that the recursion depth is at most $\log(d\varepsilon^{-1})$, but this simulation requires looking at neighborhoods of radius $\log(d\varepsilon^{-1})$. Following this approach, this is little hope of getting a recursion depth independent of ε, which is required for a $\mathrm{poly}(d\varepsilon^{-1})$-time procedure. As an aside, the size guarantee of the components is necessarily probabilistic, since the random contractions could lead to large sizes.

Much of the driving force behind this work was the quest for a $\mathrm{poly}(d\varepsilon^{-1})$-time tester for planarity. This question was resolved recently (by the authors) using a different approach from spectral graph theory, which was itself developed for sublinear time algorithms for finding minors [6,7]. A major inspiration is the random walk based one-sided bipartiteness tester of Goldreich-Ron [45]. This paper is a continuation of that line of work, and is a further demonstration of the power of spectral techniques for sublinear algorithms. The tools build on local graph partitioning techniques pioneered by Spielman-Teng [54], which is itself based on classic mixing time results of Lovász-Simonovits [3]. In this paper, we develop new local partitioning tools that form the core of partition oracles.

We also mention other key results in the context of sublinear algorithms for minor-closed families, notably the Czumaj et al [34] upper bound of $O(\sqrt{n})$ for testing cycle minor-freeness, the Fichtenberger et al [35] upper bound of $O(n^{2/3})$ for testing $K_{2,r}$-minor-freeness, and poly$(d\varepsilon^{-1})$ testers for outerplanarity and bounded treewidth graphs [43, 44].

4.1 Main Ideas

The starting point for this work are the spectral methods used in [6, 7]. These methods allow discovering cut properties within a neighborhood of radius poly$(d\varepsilon^{-1})$, without explicitly constructing the entire neighborhood.

One of the key tools used in these results in a local partitioning algorithm, based on techniques of Spielman-Teng [54]. The algorithm takes a seed vertex s, performs a diffusion from s (equivalently, performs many random walks) of length poly$(d\varepsilon^{-1})$, and tracks the diffusion vector to detect a low conductance cut around s in poly$(d\varepsilon^{-1})$ time. We will use the term *diffusions*, instead of random walks, because we prefer the deterministic picture of a unit of "ink" spreading through the graph. A key lemma in previous results states that, for graphs in minor-closed families, this procedure succeeds from more than $(1-\varepsilon)n$ seed vertices. This yields a global algorithm to construct a hyperfinite decomposition with components of poly$(d\varepsilon^{-1})$ size. Pick a vertex s at random, run the local partitioning procedure to get a low conductance cut, remove and recurse. Can there be a local implementation of this algorithm?

Let us introduce some setup. We will think of a global algorithm that processes seed vertices in some order. Given each seed vertex s, a local partitioning algorithm generates a low conductance set $C(s)$ containing s (this is called a cluster). The final output is the collection of these clusters. For any vertex v, let the *anchor* of v be the s such that $v \in C(s)$. A local implementation boils down to finding the anchor of query vertex v.

Observe that at any point of the global procedure, some vertices have been clustered, while the remaining are still *free*. The global procedure described above seems hopeless for a local implementation. The cluster $C(s)$ is generated by diffusion in some subgraph G' of G, which was the set of free vertices when seed s was processed. Consider a local procedure trying to discover the anchor of v. It would need to figure out the free set corresponding to every potential anchor s, so that it can faithfully simulate the diffusion used to cluster v. From an implementation standpoint, it seems that the natural local algorithm is to use diffusions from v in G to discover the anchor. But diffusion in a subgraph G' is markedly different from G. It appears that a local implementation needs to simulate the partitioning used to find clusters by using diffusion directly in G.

Finding low conductance cuts in subsets, by diffusion in supersets: Let us now modify the global algorithm with this constraint in mind. At some stage of the global algorithm, there is a set F of free vertices. We need to find a low conductance cut contained in F, while running random walks in G. Note that we must be able to deal with F as small as $O(\varepsilon n)$. Thus, random walks (even starting from F) will leave F quite often, leading to technical challenges in getting low conductance cuts contained in F.

One of our main insights is that these challenges can be dealt with, even for diffusions of $\text{poly}(d\varepsilon^{-1})$ length. We show that, for a uniform random vertex $s \in F$, a spectral partitioning algorithm that performs diffusion from s in G can detect low conductance cuts contained in F. Diffusion in the superset (all of V) provides information about the subset F. This is a rather technical and non-trivial result, and crucially uses the spectral properties of minor-closed families. Note that diffusions from F can spread very rapidly in short random walks, even in planar graphs. Consider a graph G, where F is a path on εn vertices, and there is a tree of size $1/\varepsilon$ rooted at every vertex of F. Diffusions from any vertex in F will initially be dominated by the trees, and one has to diffuse for at least $1/\varepsilon$ timesteps before structure within F can be detected. Thus, the proof of our theorem has to look at average behavior over a sufficiently large time horizon before low conductance cuts in F are "visible". Remarkably, it suffices to look at $\text{poly}(d\varepsilon^{-1})$ timesteps to find structure in F, because of the behavior of diffusions in minor-closed families.

The main technical tool used is the Lovász-Simonovits curve technique [3], whose use was pioneered by Spielman-Teng [54]. We also use the truncated probability vector technique from Spielman-Teng to give cleaner implementations and proofs. A benefit of using diffusion (instead of random walks) on truncated vectors is the partitioning process, as well as the size bounds, becomes deterministic.

The problem of ordering the seeds: With one technical hurdle out of the way, we end up at another gnarly problem. The above procedure only succeeds if the seed is in F. Quite naturally, one does not expect to get any cuts in F by diffusing from a random vertex in G. From the perspective of the global algorithm, this means that we need some careful ordering of the seeds, so that low conductance cuts are discovered. Unfortunately, we also need some local procedures to discover this ordering. The authors struggled with carrying out this approach, but to no avail.

To rid ourselves of the ordering problem, let us consider the following, almost naive global algorithm. First, order the vertices according to a uniform random permutation. At any stage, there is a free set F. We process the next seed vertex s by running some spectral partitioning procedure, to get a low conductance cut $C(s)$. Simply output $C(s) \cap F$ (instead of $C(s)$) as the new cluster, and update F to $F \setminus C(s)$. It is easy to locally implement this procedure. To find the anchor of v, perform a diffusion of poly(ε^{-1}) timesteps from v. For every vertex s with high enough value in the diffusion vector, determine if $C(s) \ni v$. The vertex s that is lowest according to the random ordering is the anchor of v. Unfortunately, there is little hope of bounding the number of edges cut by the clustering. When s is processed, it may be that $s \notin F$, and there is no guarantee of $C(s) \cap F$. Can we modify the procedure to bound the number of cut edges, but still maintain its ease of local implementability?

The amortization argument: Consider the scenario when $F = \Theta(\varepsilon n)$. Most of the subsequent seeds processed are not in F and there is no guarantee on the cluster conductance. But every $\Theta(1/\varepsilon)$ seeds (in expectation), we will get a "good" seed s contained in F, such that $C(s) \cap F$ is a low conductance set. (This is promised by the diffusion algorithm that we develop in this paper, as discussed earlier.) Our aim is to perform some amortization, to argue that $|C(s) \cap F|$ is so large, that we can "charge" away the edges cut by the previous $\Theta(1/\varepsilon)$ seeds.

This amortization is possible because our spectral tools give us much flexibility in the (low) conductances obtained. Put differently, we essentially prove that existence of many cuts of extremely low conductance, and show that it is "easy" for a diffusion-based algorithm to find such cuts. (This is connected to the spectral behavior of minor-closed families.) As a consequence, we can actually pre-specify the size of the low conductance cuts obtained. We show that for $F = \Omega(\varepsilon n)$, there exists a value $k = \text{poly}(\varepsilon^{-1})$ such that for at least poly(ε)n vertices $s \in F$, a spectral partitioning procedure seeded at s can find a cut of size $\Theta(k)$ and conductance at most ε^c. Moreover, this cut contains at least $\varepsilon^{c'} k$ vertices in F, despite the procedure being oblivious to F. The parameter c can be easily tuned, so we can increase c arbitrarily, at the cost of polynomial increases in running time. This tunability is crucial to our amortization argument. We also show that given query access to F, we can actually determine such a value k in poly($d\varepsilon^{-1}$) time.

So when the global algorithm processes a seed s, it runs the above spectral procedure to try to obtain a set of size $\Theta(k)$ with conductance at most ε^c. (If the procedure fails, the global algorithm simply set $C(s) = \{s\}$.) Thus, we cut $O(\varepsilon^c kd)$ edges for each seed processed. But after every $O(1/\varepsilon)$ seeds, we choose a "good" seed such that $|C(s) \cap F| > \varepsilon^{c'} k$. The total number of edges cut is $O(\varepsilon^c kd \times \varepsilon^{-1}) = O(\varepsilon^{c-1} kd)$. The total number of new vertices clustered is at least $\varepsilon^{c'} k$. Because we can tune parameters with much flexibility, we can set $c \gg c'$. So the total number of edges cut is $O(\varepsilon^{c-c'-1} d)$ times the number of vertices clustered, where $c - c' - 1 > 1$. Overall, we will cut only $O(\varepsilon nd)$ edges.

Making it work through phases: Unfortunately, as the process described above continues, F shrinks. Thus, the original choice of k might not work, and moreover, the guarantees on $|C(s) \cap F|$ for good seeds no longer hold. So we need to periodically recompute the value of k. In a careful analysis, we show that this recomputation is only required $\mathrm{poly}(\varepsilon^{-1})$ times. Formally, we implement the recomputation through *phases*. Each vertex is independently assigned to one of $\mathrm{poly}(\varepsilon^{-1})$ phases. Technically, we choose the phase of a vertex by sampling an independent geometric random variable. We heavily use the memoryless property in our analysis.

In each phase, the value of k is fixed (in our analysis, we call it r_h). At the end of each phase, we compute a fresh value of k, using random access to the free set F at that point. The local partition oracle simply computes all values of k (for all phases) before beginning any partition. The oracle (for v) runs a diffusion from v to get a collection of candidate anchors. For each candidate s, the oracle determines its phase, runs the spectral partitioning algorithm with correct phase parameters, and determines if the candidate's low conductance cut contains v. Call such an s successful. The anchor is simply the successful vertex of minimum phase, with ties broken by vertex id.

4.2 Algorithm overview

We first present the global algorithm that the partition oracle simulates. It takes as input a graph G from a minor-closed family, a parameter ε, and a random seed $\boldsymbol{R}$.

The following parameters are derived from $\boldsymbol{R}$ which is discussed in §4.2.4. For now we will introduce the following which are independent random variables:

- h_v: For each $s \in V$ associate a number which is $min\frac{2\log(1/\beta)}{\delta}, Geom(\delta)$. Thus, phases are geometrically distributed with rate δ (unless, the Geometric Variable takes on a very large value in which case we truncate it down to $\frac{2\log(1/\beta)}{\delta}$.
- $t_{s,h}$: For each $s \in V$ and phase h, we select a uar length in $t_{s,h}$ from $[1,\ell]$.

The following parameters are derived from ε:

- $\rho = \varepsilon^{1500}$: This is the minimum probability parameter.
- $\ell = \varepsilon^{-30}$: This is the random walk length.
- $\beta = 50\varepsilon$: This is the unclustered fraction cutoff.
- $\delta = \varepsilon^{1600}$: This is the probability with which a free vertex is added to P_h

`globalPartition`$(G, \varepsilon, \boldsymbol{R})$

1. Initialize the free set, $F_0 = V$.
2. Initialize the partition $\boldsymbol{P}$ as an empty collection.
3. For $h = 1$ to $\frac{2\log(1/\beta)}{\delta}$:
   (a) Set $r_h = \texttt{findr}(h, \boldsymbol{R})$.
   (b) Assign $P_h = \{w \in V \ : \ h_w = h\}$ to be the set of phase h vertices, and initialize the first free set in this phase $F_{(h,1)} = F_h$
   (c) For v_i in P_h, $i = 1$ to $|P_h|$ ordered according to label:
      i. Compute $C = \texttt{cluster}(r_h, t_{v,h}, v)$
      ii. Add $C \cap F_{(h,i)}$ to the partition $\boldsymbol{P}$.
      iii. Set $F_{(h,i+1)} = F_{(h,i)} \setminus C$
   (d) Set $F_{h+1} = F_{h,|P_h|+1}$
4. For every v such that $h_v > \frac{2\log(1/\beta)}{\delta}$ and $v \in F_{-2\log\beta/\delta+1}$, add $\{v\}$ to the partition $\boldsymbol{P}$.
5. Output $\boldsymbol{P}$.

Theorem 4.2.1 *Given random access to a graph G from a minor-closed family, random access to random seed $\boldsymbol{R}$, and $\varepsilon \in [0,1]$, the partition $\boldsymbol{P}$ output by the procedure* `globalPartition`$(G, \varepsilon, \boldsymbol{R})$ *has the following properties.*

- *In expectation over the randomness of $\boldsymbol{R}$, at $100\varepsilon dn$ edges cross between parts of $\boldsymbol{P}$.*
- *Every set of vertices in $\boldsymbol{P}$ has size at most $2\ell^{80}$.*

The proof of Theorem 4.2.1 is in §4.2.5. The proof relies on the procedure `findr` detailed in §4.2.3.

4.2.1 The partition oracle

We now describe the main partition oracle procedure, `findPartition`$(v, \boldsymbol{R})$. (Recall that $\boldsymbol{R}$ represents all the randomness.) The real work is done by an auxiliary procedure `findAnchor`$(v, \boldsymbol{R})$ that outputs the seed whose cluster contains v.

`findAnchor`$(v, \boldsymbol{R})$

1. Compute $B(v)$.
2. Initialize $D = \emptyset$.
3. For every $s \in B(v)$:
   (a) Call `findr`(h_s) to get r_{h_s}.
   (b) Using r_{h_s}, compute $C =$ `cluster`(r_{h_s}, t_{s,h_s}, s).
   (c) If C contains v, add s to D.
4. Output the smallest vertex according to label in D.

`findPartition`$(v, \varepsilon, \boldsymbol{R})$

1. Call `findAnchor`(v) to get the anchor s.
2. Call `findr`(h_s)
3. Compute $C =$ `cluster`(r_{h_s}, t_{s,h_s}, s).
4. Initialize $P = \emptyset$.
5. For every $u \in C$: call `findAnchor`(u). If the output is s, add u to P.
6. Output P.

Claim 4.2.2 *For some absolute constant c,* $\texttt{findPartition}(v, \varepsilon, \boldsymbol{R})$ *returns a set containing v that is consistent with the partition from* $\texttt{globalPartition}(G, \varepsilon, \boldsymbol{R})$*, in time $O((d/\varepsilon)^c)$.*

Proof First let us show that the set returned by $\texttt{findPartition}(v, \varepsilon, \boldsymbol{R})$ is the same as some part in the partition $\boldsymbol{P}$ produced by $\texttt{globalPartition}(G, \varepsilon, \boldsymbol{R})$. In the global algorithm, let $\mathit{anchor}(v)$ be the least vertex, s, according to label such that $v \in \texttt{cluster}(r_{h_s}, t_{s,h_s}, s)$. By construction, for every part, P, in the partition returned by the global algorithm, there is some vertex, s, such that $P = \{u \in V \ : \ \mathit{anchor}(u) = s\}$.

For any u, $\texttt{findAnchor}(u)$ must return $\mathit{anchor}(u)$. Suppose this does not hold. Then we would have that $\texttt{findAnchor}(u) = t$, but $\mathit{anchor}(u) = s$, $s \neq t$. It must be that $s \in B(u)$ since random walks on G are symmetric and since $u \in \texttt{cluster}(r_{h_s}, t_{s,h_s}, s)$, $p_{s,t_{s,h_s}}(u) \geq \rho$. So, if $\texttt{findAnchor}(u) = t$, then t precedes s according to label and $u \in \texttt{cluster}(r_{h_t}, t_{t,h_t}, t)$, which would mean $s = t$ which is a contradiction.

Fix some $v \in V$, and let $P \subseteq V$ be the part v belongs to in the global algorithm, and let Q be the part v belongs to according to $\texttt{findPartition}(v)$. If $u \in Q$, then $\texttt{findAnchor}(u) = \texttt{findAnchor}(v) = \mathit{anchor}(v)$ and so $u \in P$. If $u \in P$, then $\mathit{anchor}(u) = \mathit{anchor}(v) = \texttt{findAnchor}(v)$ and also $u \in \texttt{cluster}(r_{h_s}, t_{s,h_s}, s)$ and so $u \in Q$. Therefore, $P = Q$ and the two algorithms are consistent.

The runtime calculation is the sum of the time taken over all the steps. The time to compute $\mathrm{B}(v)$ is $O((d/\varepsilon)^{1530})$, by Claim 4.2.3. The procedure $\texttt{findAnchor}(v)$ then takes time $T_{\texttt{findAnchor}} = |B(v)|\,(T_{\texttt{findr}} + T_{\texttt{cluster}})$ where one observes $T_{\texttt{findr}} = O((d/\varepsilon)^{3200})$ is the time complexity of $\texttt{findr}$ (Claim 4.2.6), $T_{\texttt{cluster}} = O((d/\varepsilon)^{3001})$ is the time complexity of $\texttt{cluster}$ (Claim 4.2.4) and $|B(v)| \leq \ell/\rho = O(\varepsilon^{-2000})$ (Definition 4.2.2). Combined, this gives the call to $\texttt{findAnchor}$ a time complexity of $O((d/\varepsilon)^{5200})$. The call to $\texttt{findr}$ in the next line is subsumed by this running time. The size of each cluster is at most $1/\rho$. The loop in line 5 involves at most $\rho^{-1} = \varepsilon^{-1500}$ calls to $\texttt{findAnchor}$, which gives us an overall time complexity of $O\left((d/\varepsilon)^{6700}\right)$. ■

4.2.2 The `cluster()` routine

Now we define the objects essential to our implementation and analysis of the partition oracle. We will need a little setup first. We use M to denote the transition matrix of the lazy random walk.

Definition 4.2.1 *For any non-negative vector $\vec{x}$ and $\delta > 0$, let $\mathrm{tr}(\vec{x}, \delta)$ be the vector obtained by zeroing out all coordinates whose values is at most δ.*

Define operator $\widehat{M}\vec{x}$ as $\mathrm{tr}(M\vec{x}, \rho)$.

Abusing notation, for any vertex $v \in V$, we will use $\vec{v}$ to denote the unit vector corresponding to vertex v.

Definition 4.2.2

- *Define $\widehat{p}_{v,t}(w)$ to be the coordinate corresponding to vertex w in $\widehat{M}^t\vec{v}$. Also, define the coordinate corresponding to vertex w in $M^t\vec{v}$ be denoted as $p_{v,t}(w)$.*
- *Define $B(v) = \{w \mid \exists t \leq \ell, \widehat{p}_{w,t}(v) \neq 0\}$.*
- *We define the* level set *$L_{v,t,k}$ to be the set of vertices corresponding to the k largest coordinates in $\widehat{M}^t\vec{v}$. Ties are broken by vertex id.*

We first argue that B(v) can be computed efficiently.

Claim 4.2.3 *The set $B(v)$ can be computed in time $O((d/\varepsilon)^{3100})$.*

Proof Truncated walks are not symmetric, so this need a little care. Let us define $B_t(v) = \{w | \widehat{p}_{w,t}(v) \neq 0\}$, so we wish to compute $\bigcup_{t \leq \ell} B_t(v)$. Note that given any w, we can determine if $w \in B_t(v)$ by computing $\widehat{M}^t\vec{w}$ in time dt/ρ by simulating diffusion. Note that the support of $\widehat{M}^t\vec{w}$ has size at most $1/\rho$ for any t, and so we can update all d neighbors of every vertex in the support at time s to obtain $\widehat{M}^{t+1}\vec{w}$. We repeat for $t \leq \ell$ steps to get $\widehat{M}^t\vec{w}$ in time $O(d\ell/\rho)$.

Note that if $\widehat{p}_{w,t}(v) \neq 0$, then $p_{w,t}(v) \geq \rho$. Since the standard random walk is symmetric, this means that $p_{v,t}(w) \geq \rho$. Thus, $|B_t(v)| \leq 1/\rho$.

Observe that if $w \in B_t(v)$, then some neighbor of w is in $B_{t-1}(v)$. Thus, if we can compute $B_{t-1}(v)$, we can go over all neighbors of vertices in $B_{t-1}(v)$, and check if each belongs to $B_t(v)$. The set $B_0(v)$ is just $\{v\}$. By iterating over t, we can compute all sets $B_t(v)$ in time $O(\ell \times (d/\rho) \times d\ell/\rho) = O((d/\varepsilon)^{3100}$ (by the choice of parameters). ■

We now describe a key procedure, that finds low conductance cuts around a vertex v.

`cluster`(r, t, v)

1. Determine $\widehat{M}^t \vec{v}$
2. For all $k = 2^r, 2^r + 1, \ldots, 2^{r+1}$ calculate $E(L_{v,t,k}, \overline{L_{v,t,k}})$
3. Let k' be the largest k such that $E(L_{v,t,k}, \overline{L_{v,t,k}})2^{r_h}/\ell^{1/3}$ if it exists
4. Set $C = L_{v,t,k'}$ if k' exists else $C = \emptyset$
5. Return $C \cup \{v\}$

Claim 4.2.4 ***cluster***(r, t, v) *runs in time* $O((d/\varepsilon)^{3001})$.

Proof As in the proof of the previous claim, one can compute $\widehat{M}^t \vec{v}$ for all $t \leq \ell$ in time $O(d\ell/\rho)$ by simulating diffusion and truncating at every step. By the properties of `findr`(Theorem 4.2.5), $2^r = O(1/\rho)$. In line (3), there are 2^r sets of vertices to consider, each has size at most $1/\rho$ with at most d edges adjacent. Counting all these edges take time $O(2^r d/\rho) = O(d\ell^{100}/\rho) = O((d/\varepsilon)^{3001})$. ■

4.2.3 The routine `findr`

`IsFree`$(v, h, \boldsymbol{R})$

1. If $h = 0$, output TRUE.
2. Determine $B(v)$. Let U be the subset of $B(v)$ with phase at most $h-1$.
3. For every $u \in U$, compute `cluster`(r_{h_u}, t_{u,h_u}, u).
4. If any `cluster`(r_{h_u}, t_{u,h_u}, u) contains v, output FALSE. Else output TRUE.

`findr`$(h, \boldsymbol{R})$

1. Call `findr`$(h-1, \boldsymbol{R})$ to determine $r_1, \ldots, r_{h-1}$.
2. Use $\boldsymbol{R}$ to determine S_h by letting $S_h = \{w \in A_h : h_w \geq h\}$
3. For every $0 \leq r \leq 100 \log \ell$ and $s \in S_h$:
   (a) Compute $\widehat{M}^{t_{s,h}}(\vec{s})$.
   (b) For every $k \in [2^r, 2^{r+1}]$ and every $w \in L_{s,t_{s,h},k}$, call `IsFree`(w, h).
   (c) If there is a k such that $E(L_{s,t_{s,h},k}, \overline{L_{s,t_{s,h},k}}) \leq kd/\ell^{1/3}$ *and* $|L_{s,t_{s,h},k} \cap F_{h-1}| \geq \beta^3 k$, mark s as an r-SUCCESS.
4. If, for some r, there are at least $|S_h|/10 \lg \ell$ successes, output this value as r_h. Otherwise, output 0

The utility of this routine lies in the following theorem guaranteeing that, with high probability over $\boldsymbol{R}$, the value r_h returned by `findr`(h) has the important property that `cluster`$(r_h, t_{s,h}, s)$ contains many free vertices whenever the free set is sufficiently large.

Theorem 4.2.5 *The following holds with probability at least* $1 - \exp(-1/\varepsilon)$. *For all* h, *if* $|F_{h-1}| \geq \beta n$, *then* `findr`(h) *outputs a (non-zero)* $r_h \leq -\lg \rho + O(1)$ *that satisfies the following property. There are at least* $\beta^5 n$ *vertices* $s \in V_{h-1}$ *such that: there exists* $k \in [2^{r_h}, 2^{r_h+1}]$ *such that* $E(L_{s,t_{s,h},k}, \overline{L_{s,t_{s,h},k}}) \leq 2^{r_h}/\ell^{1/3}$ *and* $|L_{s,t_{s,h},k} \cap F_{h-1}| \geq \beta^3 k$.

The proof of Theorem 4.2.5 is the subject of §4.4.

Claim 4.2.6 `findr`(h) *runs in time* $O((d/\varepsilon)^{3200})$.

Proof Consider a call to `findr`(h). Let us consider what happens after the recursive call in step 1. Fix an integer $r \in [0, 100 \log \ell]$. Fix $s \in S_h$ (and recall $|S_h| \leq \log^4 \ell/\beta$). Let τ_b denote the time taken in step 1(b) of `findr`(h) (for the fixed r and s above). Let τ_{max} denote the maximum time spent in any call to `IsFree`(w, h) where w belongs to some level set of size k for $k \in [2^r, 2^{r+1}]$. We have

$$\tau_b \leq \tau_{max} \sum_{k=2^r}^{2^{r+1}} k \leq 4\tau_{max} \cdot 2^r.$$

Thus, total time spent in step 2(b) over all $r \in [0, 100 \log \ell]$ and all $s \in S_h$ is at most $4\tau_{max}|S_h| \sum 2^r$. Note that the time taken in `IsFree`(w, h') is at most

$$\tau_{max} \leq \sum_{u \in B(w)} \text{time taken to find } C(u)$$

By Claim 4.2.4 and Claim 4.2.3, this is at most $d\ell/\rho \cdot O(d\ell^{100}/\rho) = O(d^2\ell^{101}/\rho^2)$.

Thus total time taken in step 1.(b) across all iterations is at most $T_b = O(|S_h|\ell^{100} \cdot d^2\ell^{101}/\rho^2)$. Step 2.(c) and Step 3 take time at most $O(T_b)$ each and thus the overall running time after the recursive calls is $O(d^2\ell^{201}/(\beta^{10}\rho))$. Each recursive call takes at most the same amount of time and summing over all h of them gives $O\left(d^2h\ell^{201}/(\beta^{10}\rho)\right) = O((d/\varepsilon)^{3200})$. ∎

4.2.4 The random seed

In this section, we will describe how the various parameters are derived from $\boldsymbol{R}$. We think of $\boldsymbol{R}$ as a random tape with the following components:

- h_v for $v \in V$: For a vertex v, we pick $g_v \sim Geom(\delta)$ and let $h_v = \min(g_v, \frac{2\log(1/\beta)}{\delta})$
- $t_{s,h}$, for each $s \in V$: For each $s \in V$ and phase $h \in [H]$, we select a uar length in $t_{s,h}$ from $[1, \ell]$.
- A_h: For each $h \leq \frac{2\log(1/\beta)}{\delta}$, we sample a (multi)set of $\log^{10} \ell/\beta^{20}$ random numbers from 1 to n.

This set A_h is used to find a multiset S_h which is used in the `findr` routine. We simply set $S_h = \{w \in A_h : h_w \geq h\}$.

Let $h_0 = \frac{2\log(1/\beta)}{\delta}$. We have the following

Claim 4.2.7 *With probability at least* $1 - \frac{2\log(1/\beta)}{\delta}\exp(-1/12\beta^4) - \exp(-\beta^4 n/12)$, *we have that for all* $1 \leq h \leq \frac{2\log(1/\beta)}{\delta}$, $|S_h| \geq (\log \ell)^4/\beta^{10}$.

Proof We will find a lower bound on $|V_{h_0}|$ that holds with high probability. For a vertex v, let I_v denote the indicator that $v \in V_{h_0}$. We have

$$\mathbf{E}[I_v] = \Pr(I_v = 1) = (1-\delta)^{h_0} \geq \exp(-2\delta h_0) \geq \beta^4$$

Thus, $\mathbf{E}[|V_{h_0}|] \geq \beta^4 n$ and, by Chernoff bounds, the probability that $|V_{h_0}| \geq \beta^4 n/2$ is at least $p = 1 - \exp(-\beta^4 n/12)$. This means for $h < h_0$, we have $|V_h| \geq \beta^4 n$ holds with probability at least p.

Now, we will show that for any $h(< h_0)$ we have $|S_h| \geq (\log \ell)^4/\beta^{10}$. To do this, note that $\mathbf{E}[|S_h|] \geq (\log \ell)^4/\beta^{16}$ and thus, by a multiplicative Chernoff Bound, we have $|S_h| \geq (\log \ell)^4/2\beta^{16}$ with probability at least $(1 - \exp(-1/12\beta^{16}))$. This holds for a particular phase, phase h. Taking a union bound over all $h \leq h_0$ finishes the proof. ■

4.2.5 Proof of Theorem 4.2.1

Proof The proof crucially hinges on the choice of various parameters. We will design a charging scheme that deposits charge on vertices. We will ensure that, with exceedingly high probability, the total charge is at least the number of cut edges. Finally, we bound the expected charge on each vertex.

The charging is done phase by phase. Note that in each phase, the edges cut are only incident to F_h.

- If $|F_h| \geq \beta n$: invoking Theorem 4.2.5, there are at least $\beta^5 n$ vertices $s \in V_h$ such that: there exists $k \in [2^{r_h}, 2^{r_h+1}]$ such that $E(L_{s,t_{s,h},k}, \overline{L_{s,t_{s,h},k}}) \leq 2^{r_h}/\ell^{1/3}$ and $|L_{s,t_{s,h},k} \cap F_{h-1}| \geq \beta^3 k$. Some of these vertices s will be in P_h, and thereby be seeds for phase h. Call any such s good for phase h. For every good s, and for every vertex in $L_{s,t_{s,h},k} \cap F_h$, we add $d\beta^{-3}\ell^{-1/3}$ units of charge. Note that a vertex may get charged from various good seeds s.

Finally, we multiply every non-zero charge with $32\beta^{-5}$. We refer to this as the *scaling step*.

- If $|F_h| < \beta n$: all vertices in F_h get d units of charge, and the charging procees terminates.

Observe that every vertex v is charged in exactly one phase. When a vertex received charge in some phase, then it must have got clustered (or was in the last charging phase). Thus, it cannot be in any free set for a subsequent phase.

Claim 4.2.8 *With probability at least* $1 - \exp(-\delta^2\beta^5 n)$*, the total charge is an upper bound on the number of edges cut.*

Proof We will argue phase by phase. The easy case is when $|F_h| < \beta n$. In this case, the total amount of charge deposited is $|F_h|d$, which is at least the total number of edges incident to F_h. No more than $|F_h|d$ edges can be cut from F_h.

Consider phase h where $|F_h| \geq \beta n$. Let us first account for the edges cut by clusters from good seeds (call these good clusters). For any cluster generated in phase h, the number of edges cut is at most $kd/\ell^{1/3}$, where k is the cluster size. By the property of good seeds, for any good seed s, there are at least $\beta^3 k$ vertices in $L_{s,t_{s,h},k} \cap F_h$. Each of these vertices gets $d\beta^{-3}\ell^{-1/3}$ units of charge, leading to a total charge of $\beta^3 k \times d\beta^{-3}\ell^{-1/3} = kd\ell^{-1/3}$. This is at least the number of edges cut by the cluster.

By Theorem 4.2.5, there are at least $\beta^5 n$ good seeds in V_h. By the properties of the geometric random variables used to generate phases, conditioned on being in V_h, the probability that a seed is in P_h is δ (and these are independent for all seeds). The expected number of good seeds in P_h is at least $\delta\beta^5 n$. By a multiplicative Chernoff bound, the probability of there being at least $\delta\beta^5 n/2$ good seeds in P_h is at least $1 - \exp(-\delta\beta^5 n/12)$.

Conditioned on V_h, the expected number of seeds in P_h is at most δn. By a multiplicative Chernoff bound, the probability of there being at most $2e\delta n$ seeds in P_h is at least $1 - \exp(-2e\delta n)$.

By a union bound, these bounds both hold with probability $\geq (1 - 2\exp(-\delta\beta^5 n/12))$. Assuming these bounds, the total number of edges cut in phase h is at most

$$|P_h|(d2^{r_h+1}\ell^{-1/3} \leq 4e\delta 2^{r_h}\ell^{-1/3}n.$$

The total number of edges cut by good clusters is at least $(\delta\beta^5 n/2) \times d2^{r_h}\ell^{-1/3}$. Thus, the total number of edges cut is at most $8e\beta^{-5}$ times the number of edges cut by good clusters.

Our initial charge was an upper bound on the number of edges cut by good clusters. The scaling step multiplied all charges by $32\beta^{-5}(\geq 8e\beta^{-5})$, so the total charge now upper bounds the total number of edges cut in this phase. We union bound over all the c/δ^2 phases, and simply upper bound by $(c/\delta^2)(2\exp(-\delta\beta^5 n/12)$ which is at most $\exp(-\delta^2\beta^5 n)$ to complete the proof. ■

Let $\mathcal{E}$ be the event that the total charge upper bounds the total number of edges cut. By Claim 4.2.8, $\Pr[\overline{\mathcal{E}}] \leq \exp(-\delta^2\beta^5 n)$.

$$\begin{aligned}
\mathbf{E}[\#\text{ edges cut}] &\leq \Pr[\mathcal{E}]\mathbf{E}[\#\text{ edges cut}|\mathcal{E}] + \Pr[\overline{\mathcal{E}}]\mathbf{E}[\#\text{ edges cut}|\overline{\mathcal{E}}] \\
&\leq \Pr[\mathcal{E}]\mathbf{E}[\#\text{ total charge}|\mathcal{E}] + \exp(-\delta^2\beta^5 n)dn \\
&\leq \mathbf{E}[\#\text{ total charge}] + 1/n \qquad (4.1)
\end{aligned}$$

Thus, it suffices to bound the total charge. Let us define a collection of events. Fix a vertex v. For phase h, the event $\mathcal{E}_{v,h}$ is the event that $|F_h| \geq \beta n$ and that v is assigned charge in phase h. Let $\mathcal{F}_v$ be the event that v ends up in some F_h where $|F_h| < \beta n$. For a fixed v, note that all these events are disjoint and partition the event of v receiving non-zero charge.

Claim 4.2.9 *For any phase h, $\mathbf{E}[$charge on v from phase $h|\mathcal{E}_{v,h}] \leq 16d\beta^{-8}\ell^{-1/3}$.*

Proof Let $\mathcal{E}'_h$ be the event that $|F_h| \geq \beta n$ and $v \in F_h$. Note that if v is not captured in this phase, then it gets no charge in this phase.

$$\begin{aligned}
\mathbf{E}[\text{charge on } v \text{ from phase } h|\mathcal{E}'_h] = (&\Pr[v \text{ captured phase } h] \\
&\times\mathbf{E}[\text{charge on } v \text{ from phase } h|\mathcal{E}_{v,h}]) \qquad (4.2)
\end{aligned}$$

Conditioned on $\mathcal{E}'_h$, v will receive charge if a good seed s that captures v is selected in P_h. Suppose there are k_v such good seeds. Crucially, observe that $k_v \leq \ell\rho^{-1}$. Each cluster is constructed from a level set, which contains vertices v such that $\widehat{p}_{s,t_{s,h}}(v)$ is non-zero. This means that $p_{v,t_{s,h}}(s) \geq \rho$. For fixed v and any value of $t_{s,h}$, there are most ρ^{-1} such values of s. There are most ℓ timesteps, leading to the upper bound of $\ell\rho^{-1}$.

Each good seed is selected in P_h with probability δ. The probability that v is captured is $1-(1-\delta)^{k_v}$. Since we have by the choice of parameters that $\delta k_v < 1/2$, it follows this probability is at least $\delta k_v/2$.

If there are b good vertices that capture v, then v receives $4bd\beta^{-8}\ell^{-1/3}$ units of charge. The probability of b such vertices being chosen is $\binom{k_v}{b}\delta^b \leq (\delta k_v)^b$.

Using the fact that $\delta k_v < 1/2$,

$$\mathbf{E}[\text{charge on } v \text{ from phase } h|\mathcal{E}'_h] = \sum_{b\geq 0} 4bd\beta^{-8}\ell^{-1/3}(\delta k_v)^b \leq 8d\beta^{-8}\ell^{-1/3}(\delta k_v)$$

By (4.2),

$$\mathbf{E}[\text{charge on } v \text{ from phase } h|\mathcal{E}_{v,h}] \leq \frac{8d\beta^{-8}\ell^{-1/3}(\delta k_v)}{\delta k_v/2} = 16d\beta^{-8}\ell^{-1/3}$$

■

We can now break up the expected charge on vertex v into various conditional expectations.

$$\mathbf{E}[\text{charge on } v] = \sum_h \Pr[\mathcal{E}_h]\mathbf{E}[\text{charge on } v \text{ from phase } h|\mathcal{E}_h] + \Pr[\mathcal{F}_v]d \qquad (4.3)$$

By Claim 4.2.9, $\mathbf{E}[\text{charge on } v|\mathcal{E}_h] \leq 16d\beta^{-8}\ell^{-1/3}$. Thus, the expected charge is at most $16d\beta^{-8}\ell^{-1/3} + \Pr[\mathcal{F}_v]d$. Note that $\sum_v \Pr[\mathcal{F}_v] = \sum_v \mathbf{E}[X_v]$, where X_v is an indicator random variable for v being in some $|F_h| < \beta n$. But $\sum_v X_v < \beta n$, so $\sum_v \Pr[\mathcal{F}_v]d \leq \beta dn$. All in all, the expected total charge is at most $(16\beta^{-8}\ell^{-1/3}+\beta)dn$. By Claim 4.2.8, the expected number of edges cut is at most twice expected charge, which by the setting of parameters is at most $2\beta = 100\varepsilon$. ■

4.3 Random walks behavior on minor-free families

Like before, we use M to denote the transition matrix of the lazy random walk. Also, recall Definition 4.2.1. We will also need another definition.

Definition 4.3.1 *A vertex v is spreading if there exists $t \leq \ell$ such that $\|\widehat{M}^t(\vec{v}) - M\widehat{M}^{t-1}(\vec{v})\|_1 \geq \rho^{1/3}$.*

The following corollary based off of [21] appears in [7]

Corollary 4.3.1 *Suppose G is a graph with maximum degree that comes from a minor closed family. Then, there exists a subset of at most $\gamma\rho^{1/4}dn$ edges on deleting which G breaks up into connected components each with size at most $1/\sqrt{\rho}$ (where γ is an explicit constant which depends on the description of the minor closed family).*

Now, let us fix a graph G with maximum degree $\leq d$ that comes from a minor closed family and use Corollary 4.3.1 to obtain pieces of size $1/\sqrt{\rho}$ with a total of at most $\gamma\rho^{1/4}dn$ edges between them. Refer to these as AST pieces and the set of edges between pieces by R. We have the following claim which asserts that short enough random walks starting at any vertex are unlikely to leave the AST piece they start in.

Claim 4.3.2 *For at least $(1 - \rho^{1/8})$ fraction of vertices v, the probability that an ℓ-length walk starting at v encounters an edge in R is at most $\ell\gamma\rho^{3/8}$.*

(And thus the probability that the walk beginning at such v leaving its AST copy is also at most $\ell\gamma\rho^{3/8}$)

Proof Consider an ℓ-length walk in G starting from the stationary distribution (which is uniform). The probability that this walk encounters an edge in R at any step is exactly $|R|/2dn$. Let the random variable X_v be the number of edges of R encountered in an ℓ-length walk from v. Note that when $X_v = 0$, then the walk remains in the AST component containing v. Thus, letting E_v denote the event that walk from v leaves the AST component, we get

$$(1/n)\sum_v \Pr[E_v] \le \mathbf{E}_{v\sim\text{u.a.r.}}[X_v] = \ell|R|/2dn \le \gamma\ell\rho^{1/4}/2$$

By Markov's, we have that for at least $(1-\rho^{1/8})$ fraction of vertices the probability that an ℓ-length walk starting at v encounters an edge in R and thus leaves the AST piece is at most $\ell\gamma\rho^{3/8}$. ∎

We define the notion of good vertices.

Definition 4.3.2 *A vertex $v \in V(G)$ is called good if the probability that a random walk of length ℓ starting at v leaves its AST piece is at most $\ell\gamma\rho^{3/8}$.*

With this setup behind us, we can now prove the main lemma of this section.

Lemma 4.3.3 *At most $\rho^{1/8}n$ vertices are spreading.*

Proof By Claim 4.3.2, we know that there are at least $(1-\rho^{1/8})n$ good vertices in G. We will show that none of the good vertices are spreading. Fix a good vertex v. Let C denote the AST component for v. Fix any time step $t \in [\ell]$. Let $p_t = \widehat{M}^t(\vec{v}) - M\widehat{M}^{t-1}(\vec{v})$. Letting $\ell \ll 1/(\gamma\rho^{1/40})$, we have

$$\begin{aligned}
\|p_t\|_1 &= \sum_{w\in C} p_t(w) + \sum_{w\notin C} p_t(w) && (4.4)\\
&\overset{(\mathbf{1})}{\le} \rho\cdot(1/\sqrt{\rho}) + \sum_{w\notin C} p_t(w) && (4.5)\\
&\overset{(\mathbf{2})}{\le} \sqrt{\rho} + \Pr[\ell \text{ length walk from } v \text{ leaves } C] && (4.6)\\
&\overset{(\mathbf{3})}{\le} \sqrt{\rho} + \ell\gamma\rho^{3/8} && (4.7)\\
&\le \sqrt{\rho} + \rho^{7/20} < \rho^{1/3} && (4.8)
\end{aligned}$$

Here, $(\mathbf{1})$ holds because $|C| \le 1/\sqrt{\rho}$ and at most ρ amount of l_1 mass can be lost at any vertex w in a step of $\widehat{M}$. So, the total mass lost over all of C can be upperbounded by $\sqrt{\rho}$. (**2**) follows because of the following 2 observations. First, that the the value at any coordinate under the action of $\widehat{M}$ on a non-negative vector v cannot exceed the value at that coordinate under the action of standard random walk matrix M. And the other one being the fact that mass lost at any fixed $t \in [\ell]$ can be no bigger than mass lost over all the ℓ steps. (**3**) follows because v is a good vertex. The last step follows by upperbounding ℓ by $1/(\gamma\rho^{1/40})$. This indeed confirms that any good vertex v is not spreading because at any time step $t \in [\ell]$, $\|p_t\|_1 < \rho^{1/3}$. This means that the number of spreading vertices is at most $\rho^{1/8}n$ as desired.

■

Claim 4.3.4 *If v is a good vertex, then $\|\widehat{M}^\ell(\vec{v}) - M^\ell\vec{v}\|_1 \leq 2\ell\rho^{7/20}$.*

Proof We write

$$\|\widehat{M}^\ell(\vec{v}) - M^\ell\vec{v}\|_1 = \| \sum_{t=0}^{t=\ell-1} \left(M^t\widehat{M}^{\ell-t}(\vec{v}) - M^{t+1}\widehat{M}^{\ell-t-1}(\vec{v})\right) \|_1 \tag{4.9}$$

$$\overset{(\mathbf{1})}{\leq} \sum_{t=0}^{t=\ell-1} \|M^t\widehat{M}^{\ell-t}(\vec{v}) - M^{t+1}\widehat{M}^{\ell-t-1}(\vec{v})\|_1 \tag{4.10}$$

$$\overset{(\mathbf{2})}{\leq} \sum_{t=0}^{t=\ell-1} \|\widehat{M}^{\ell-t-1}(\vec{v}) - M\widehat{M}^{\ell-t-1}(\vec{v})\|_1 \tag{4.11}$$

$$\overset{(\mathbf{3})}{\leq} \ell \cdot 2\rho^{7/20} \tag{4.12}$$

Here (**1**) is just triangle inequality. (**2**) follows because multiplying with a stochastic (M) cannot increase the l_1-norm of a vector. The last step follows because v is a good vertex.

■

4.4 Correctness of `findr`

The proof has a number of moving parts. It will be convenient to collect all the values various parameters for ease of reference.

We start by doing technical calculations involving the Lovász-Simonovits technique.

Table 4.1.*Symbols Used*

Notation	Meaning	Where defined	Value chosen
β	Unclustered fraction cutoff	§4.2	50ε
α	Heavy Bucket Parameter	Definition 4.4.3	$\varepsilon^{4/3}/1000$
ℓ	Walk Length	§4.2	ε^{-30}
ρ	Min Probability	§4.2	ε^{1500}
δ	Phase rate	§4.2.4	ε^{1600}

4.4.1 The Lovász-Simonovits lemma

Our analysis closely follows that in [54] which is based on techniques from [3]. We reproduce the following definition.

Definition 4.4.1 *For a vector* $\mathbf{p}$ *over* V*, the function* $I(\mathbf{p}, x) : \mathbb{R}^n \times [n] \to [0, 1]$ *is defined as*

$$I(\mathbf{p}, x) = \max_{\substack{\mathbf{w} \in [0,1]^n \\ \sum \mathbf{w}(u) = x}} \sum_{u \in V} \mathbf{p}(u)\mathbf{w}(u)$$

This is equivalent to summing over the x *heaviest elements of* $\mathbf{p}$ *when* x *is an integer, and linearly interpolating between these points otherwise.*

For notational convenience, we define the following reparameterization:

$$I_{s,t}(x) = I(\widehat{M}^t(\vec{v}), x).$$

The fundamental lemma of Lovász-Simonovits is the following.

Lemma 4.4.1 *Let* $\overline{x} = \min(x, n - x)$*. Consider any non-negative vector* $\vec{p}$*, and let* S_x *denote the level set of* $M\vec{p}$ *with* x *vertices.*

$$I(M\vec{p}) \leq (1/2)(I(\vec{p}, x - 2\overline{x}\Phi(S_x)) + I(\vec{p}, x - 2\overline{x}\Phi(S_x)))$$

Note that this implies monotonicity, $I(M\vec{p}) \leq I(\vec{p})$. The application of this lemma to our setting leads to the following statement.

Lemma 4.4.2 *For all $t \le \ell$ and $x \le 1/\rho$,*

$$I_{s,t}(x) \le (1/2)(I_{s,t-1}(x(1-\Phi(L_{s,t,x}))) + I_{s,t-1}(x(1+\Phi(L_{s,t,x}))))$$

Let $f_{t,w,y}$ be the straight line between the points $(w, I_{s,t}(w))$ and $(y, I_{s,t}(y))$.

Lemma 4.4.3 *Let $t_0 < t_1 < \ldots < t_h$ be time steps. Suppose $\forall i \le h$ and $x \in [w,y]$, $\Phi(L_{s,t_i,x}) \ge \phi$. Then, $\forall i \le h, x \in [w,y]$,*

$$I_{s,t_i}(x) \le f_{t_0-1,w,y}(x) + \sqrt{\min(x-w, y-x)}(1-\phi^2/8)^i$$

Proof For convenience, let $\Delta(x) = \min(x-w, y-x)$. We prove by induction over i.

Base case, $i = 0$. We do a case analysis.

- Suppose $x = w$ or $x = y$. By monotonicity, $I_{s,t_0}(x) \le I_{s,t_0-1}(x)$. Since $x \in \{w,y\}$, the latter is exactly $f_{t_0,w,y}(x)$.
- Suppose $x \in [w+1, y-1]$. Then $\Delta(x) \ge 1$ and $I_{s,t_0}(x) \le 1 \le \sqrt{\Delta(x)}$.
- Suppose $x \in (w, w+1)$. Note that $\Delta(x) = w - x < 1$. By the definition of the LS curve, $I_{s,t_0}(x) = I_{s,t_0}(w) + (w-x)(I_{s,t_0}(w+1) - I_{s,t_0}(w)) \le I_{s,t_0-1}(w) + \sqrt{w-x}$ $\le f_{t_0-1,w,y}(x) + \sqrt{\Delta(x)}$.
- Suppose $x \in (y-1, y)$. An identical argument to the above holds.

Now for the induction. By Lemma 4.4.2,

$$\begin{aligned} I_{s,t_i}(x) &\le (1/2)[I_{s,t_i-1}(x(1-\Phi(L_{s,t_i,x}))) + I_{s,t_i-1}(x(1+\Phi(L_{s,t_i,x})))] && (4.13) \\ &\le (1/2)[I_{s,t_{i-1}}(x(1-\Phi(L_{s,t_i,x}))) + I_{s,t_{i-1}}(x(1+\Phi(L_{s,t_i,x})))] && (4.14) \end{aligned}$$

Note that $\Delta_x \le x$, so by concavity,

$$I_{s,t_i}(x) \le (1/2)[I_{s,t_{i-1}}(x - \Delta(x)\phi)) + I_{s,t_{i-1}}(x + \Delta(x)\phi))] \qquad (4.15)$$

Note that $x - \Delta(x)\phi$ and $x + \Delta(x)\phi$ both lie in $[w, y]$. Therefore, we can apply the induction hypothesis.

$$\begin{aligned}
I_{s,t_i}(x) &\leq (1/2)[f_{t_0-1,w,y}(x - \Delta(x)\phi) + \sqrt{\Delta(x - \Delta(x)\phi)}(1 - \phi^2/8)^{i-1} && (4.16)\\
&\quad + f_{t_0-1,w,y}(x + \Delta(x)\phi) + \sqrt{\Delta(x + \Delta(x)\phi)}(1 - \phi^2/8)^{i-1}] && (4.17)\\
&= (1/2)[f_{t_0-1,w,y}(x - \Delta(x)\phi) + f_{t_0-1,w,y}(x + \Delta(x)\phi)]\\
&\quad + (1/2)\Big[\sqrt{\Delta(x - \Delta(x)\phi)}(1 - \phi^2/8)^{i-1}\\
&\qquad + \sqrt{\Delta(x + \Delta(x)\phi)}(1 - \phi^2/8)^{i-1}\Big] && (4.18)
\end{aligned}$$

Since $f_{t_0-1,w,y}$ is a linear function, the first term is exactly $f_{t_0-1,w,y}(x)$. We analyze the second term.

We first assume that $\Delta(x) = x - w$ (instead of $y - x$).

$$\begin{aligned}
\Delta(x - \phi\Delta(x)) &= \min(x - \phi\Delta(x) - w, y - x + \phi\Delta(x)) && (4.19)\\
&= \min((1 - \phi)\Delta(x), y - x + \phi\Delta(x)) \leq (1 - \phi)\Delta(x) && (4.20)
\end{aligned}$$

Analogously,

$$\begin{aligned}
\Delta(x + \phi\Delta(x)) &= \min(x + \phi\Delta(x) - w, y - x - \phi\Delta(x)) && (4.21)\\
&= \min((1 + \phi)\Delta(x), y - x - \phi\Delta(x)) \leq (1 + \phi)\Delta(x) && (4.22)
\end{aligned}$$

Thus, the second term of (4.18) is at most $(1/2)(1 - \phi^2/8)\sqrt{\Delta(x)}(\sqrt{1 - \phi} + \sqrt{1 + \phi})$.

Now, we consider $\Delta(x) = y - x$.

$$\begin{aligned}
\Delta(x - \phi\Delta(x)) &= \min(x - \phi\Delta(x) - w, y - x + \phi\Delta(x)) && (4.23)\\
&= \min(x - \phi\Delta(x) - w, (1 + \phi)\Delta(x)) \leq (1 + \phi)\Delta(x) && (4.24)
\end{aligned}$$

Analogously,

$$\begin{aligned}
\Delta(x + \phi\Delta(x)) &= \min(x + \phi\Delta(x) - w, y - x - \phi\Delta(x)) && (4.25)\\
&= \min(x + \phi\Delta(x) - w, (1 - \phi)\Delta(x)) \leq (1 - \phi)\Delta(x) && (4.26)
\end{aligned}$$

In this case as well, the second term of (4.18) is at most

$$(1/2)(1 - \phi^2/8)\sqrt{\Delta(x)}(\sqrt{1 - \phi} + \sqrt{1 + \phi})$$

In both cases, we can upper bound (4.18) as follows. (We use the inequality $\frac{\sqrt{1-z}+\sqrt{1+z}}{2} \leq 1 - z^2/8$.

$$I_{s,t_i}(x) \leq f_{t_0-1,w,y}(x) + (1-\phi^2/8)^{i-1}\sqrt{\Delta(x)}\frac{\sqrt{1-\phi}+\sqrt{1+\phi}}{2}$$

which means $I_{s,t_i}(x) \leq f_{t_0-1,w,y}(x) + (1-\phi^2/8)^{i}\sqrt{\Delta(x)}$

■

4.4.2 From leaking timesteps to the dropping of the LS curve

We fix a source vertex s, and are looking at the evolution of $\widehat{M}^t(\vec{s})$. Therefore, we drop the dependence of s from much of the notation.

We use $\widehat{p}_t$ to denote $\widehat{M}^t(\vec{s})$. In this section, there will be a designated *free set* of vertices, denoted F.

Definition 4.4.2 *A timestep t is called* leaking for source s *if, for all $k \leq 2(\rho\alpha)^{-1}$ such that $|L_{s,t,k} \cap F| \geq \alpha^2 k/6$. $L_{s,t,k}$ has conductance at least $1/\ell^{1/3}$.*

If timestep t is not leaking for s, there exists $k \leq 2(\rho\alpha)^{-1}$ such that $|L_{s,t,k} \cap F| \geq \alpha^2 k/6$ and $\phi(L_{s,t,k}) < 1/\ell^{1/3}$. Such a k is denoted as an (s,t)-certificate of non-leakiness.

We will set $\alpha = \varepsilon^{4/3}/1000$.

Following the construction of the LS curve $I_{s,t}$, we will order each vector $\widehat{p}_t$ in decreasing order of coordinations, breaking ties by id. The *rank* of a vertex is its position in (the sorted version of) $\widehat{p}_t$.

Definition 4.4.3 *Let the* bucket *$B_{t,r}$ denote the set of vertices whose rank in $\widehat{p}_t$ is in the range $[2^r, 2^{r+1})$.*

A bucket $B_{t,r}$ is called heavy *if $\sum_{v \in B_{t,r} \cap F} \widehat{p}_t(v) \geq \alpha$. (The free part of the bucket has large probability.)*

Lemma 4.4.4 *Fix $r \geq 0$. Suppose $\ell' = \beta\ell/8$ and we have ℓ' leaking timesteps $t_0 < t_1 < \ldots < t_{\ell'}$ such that for all $0 \leq i \leq \ell'$, $B_{t_i,r}$ is heavy. Then, $I_{s,t_{\ell'}}(2^{r+1}) < I_{s,t_0}(2^{r+1}) - \alpha/4$.*

Proof Since $B_{t_0,r}$ is heavy, $I_{s,t_0}(2^r) < 1$. Since the support of $\widehat{p}_t$ is at most ρ^{-1}, this implies that $2^r < \rho^{-1}$ and $r \leq -\lg \rho$ (and this holds by the choice of parameters).

For all $v \in B_{t,r}$, $\widehat{p}_t(v) \leq 1/2^r$. Since $\sum_{v \in B_{t,r} \cap F} \widehat{p}_t(v) \geq \alpha$, $|B_{t,r} \cap F| \geq \alpha 2^r$. Call $w \in [2^r, 2^{r+1})$ a *balanced split* for t if $|L_{t,w} \cap F| \geq \alpha 2^r/3$ and $\sum_{v \in B_{t,r} \setminus L_{t,w}} \widehat{p}_t(v) \geq \alpha/3$. For convenience, let $T = \{t_0, t_1, \ldots, t_{\ell'}\}$.

Claim 4.4.5 *There exists w that is a balanced split for at least an $\alpha/3$-fraction of timesteps in T.*

Proof An averaging argument. Pick w uar in $[2^r, 2^{r+1})$. Let X_i be the indicator for w being a good split for t_i. Recall that $|B_{t_i,r} \cap F| \geq \alpha 2^r$. Sort the vertices of $B_{t_i,r} \cap F$ by increasing rank and consider the vertices in positions $\alpha 2^r/3$ and $2\alpha 2^r/3]$. Let the rank corresponding to these vertices by u_1 and u_2. We first argue that any rank $w \in [u_1, u_2]$ is a balanced split. We have $|L_{t,w} \cap F| \geq \alpha 2^r/3$ because $w \geq u_1$. For all $v \in B_{t_i,r}$, $\widehat{p}_{t_i}(v) \leq 1/2^r$. Thus, $\sum_{v \in L_{t_i,u_2}} \widehat{p}_{t_i}(v) \leq (1/2^r)(2\alpha 2^r/3) = 2\alpha/3$. Note that $\sum_{v \in B_{t_i,r}} \widehat{p}_t(v) \geq \alpha$, since the bucket is heavy Hence, for any $w \leq u_2$, $\sum_{v \in B_{t,r} \setminus L_{t,w}} \widehat{p}_t(v) \geq \alpha - 2\alpha/3 = \alpha/3$.

As a consequence, for any t_i, there are at least $\alpha 2^r/3$ values of w that are balanced splits. In other words, $\mathbf{E}[X_i] \geq \alpha/3$. By linearity of expectation, $\mathbf{E}[\sum_{i \leq \ell'} X_i] \geq \alpha\ell'/3$. Thus, there must exist some $w \in [2^r, 2^{r+1})$ that is a good split for at least $\alpha\ell'/3$ timesteps. ■

Pick such a w, as promised by Claim 4.4.5, and let $t_{i_1} < t_{i_2} < \ldots < t_{i_{\alpha\ell'/3}}$ be the timesteps for which w is a good split. Let $y = 2^{r + \lceil \lg(1/\alpha) \rceil} \in [2^r/\alpha, 2^{r+1}/\alpha]$. Since $r \leq -\lg \rho$, $y \leq 2(\rho\alpha)^{-1}$. Note that for all $j \leq \alpha\ell'/3$ and $x \in [w, y]$, $L_{t_{i_j},x}$ contains least $\alpha 2^r/3$ vertices of F. Thus, at least a $(\alpha 2^r/3)/(2^{r+1}/\alpha) \geq \alpha^2/6$-fraction of $L_{t_{i_j},x}$ is in F. Since t_{i_j} is leaking, $\Phi(L_{t_{i_j},x}) \geq 1/\ell^{1/3}$.

We apply Lemma 4.4.3. For all $x \in [w, y]$, $I_{s,t_{\ell'}}(x) \leq I_{s,t_{i_{\alpha\ell'/3}}}(x) \leq f_{t_{i_1}-1,w,y}(x) + \sqrt{x}(1 - 1/8\ell^{2/3})^{\alpha\ell'/3}$. By the choice of parameters, for sufficiently small ε (say $0 < \varepsilon < 1/50$), we have $\ell' = \beta\ell/8 \geq 1/\alpha^2\beta$ and therefore we have $(1 - 1/8\ell^{2/3})^{\alpha\ell'/3} \leq$

$\exp(-1/\alpha)$ By monotonicity of the LS curves, $I_{s,t_{\ell'}}(x) \leq f_{t_{i_1-1},w,y}(x) + \exp(-1/\alpha)$ $\leq f_{t_{i_0},w,y}(x) + \exp(-1/\alpha)$. Specifically, we get

$$I_{s,t_{\ell'}}(2^{r+1}) \leq f_{t_{i_0},w,y}(2^{r+1}) + \exp(-1/\alpha). \tag{4.27}$$

Since w is a good split, $I_{s,t_{i_0}}(2^{r+1}) \geq I_{s,t_{i_0}}(w) + \alpha/3$. Note that

$$\begin{aligned} f_{t_{i_0},w,y}(2^{r+1}) &= I_{s,t_{i_0}}(w) + (2^{r+1} - w)\left(\frac{I_{s,t_{i_0}}(y) - I_{s,t_{i_0}}(w)}{y - w}\right) \\ &\leq I_{s,t_{i_0}}(w) + 2^{r+1}/(y/2) \qquad (4.28) \\ &\leq I_{s,t_{i_0}}(w) + 2^{r+1} \times (\alpha^2/2^r) = I_{s,t_{i_0}}(w) + 2\alpha^2 \qquad (4.29) \end{aligned}$$

The first inequality above follows by upper bounding $I_{s,t_{i_0}}(y) - I_{s,t_{i_0}}(w)$ by 1, dropping the negative term and noting that $y - w \geq y/2$ for a sufficiently small α. Together with (4.27), we get

$$\begin{aligned} I_{s,t_{\ell'}}(2^{r+1}) &\leq f_{t_{i_0},w,y}(2^{r+1}) + \exp(-1/\alpha) \\ &\leq I_{s,t_{i_0}}(w) + 2\alpha^2 + \exp(-1/\alpha) \\ &\leq I_{s,t_{i_0}}(2^{r+1}) - \alpha/3 + 2\alpha^2 + \exp(-1/\alpha) \qquad (4.30) \end{aligned}$$

By monotonicity of the LS curve, $I_{s,t_{\ell'}}(2^{r+1}) < I_{s,t_0}(2^{r+1}) - \alpha/4$.

■

Now for the main lemma of this subsection. It is a direct corollary of the previous lemma.

Lemma 4.4.6 *Fix any r. There are at most $8\ell'/\alpha$ leaking timesteps t where $B_{t,r}$ is heavy.*

Proof We prove by contradiction. Suppose there are more than $8\ell'/\alpha > 4(\ell'+1)/\alpha$ leaking timesteps t where $B_{t,r}$ is heavy. We break these up into $4/\alpha$ contiguous blocks of $\ell' + 1$ leaking timesteps. By Lemma 4.4.4, after every such block of timesteps, $I_{s,t}(2^{r+1})$ reduces by more than $\alpha/4$. Note that $I_{s,0}(2^{r+1}) \leq 1$, and thus, after $4/\alpha$ blocks, $I_{s,t}(2^{r+1})$ becomes negative. Contradiction to the non-negativity of $I_{s,t}(2^{r+1})$.

■

4.4.3 From relevant vertices to the final proof

We will need the following simple "reverse Markov" inequality for bounded random variables.

Fact 4.4.7 *Let X be a random variable taking values in $[0, 1]$ such that $\mathbf{E}[X] \geq \delta$. Then $\Pr[X \geq \delta/2] \geq \delta/2$.*

Proof Let p be the probability that $\Pr[X \geq \delta/2]$.

$$\begin{aligned} \delta \leq \mathbf{E}[X] &= \Pr[X \geq \delta/2]\mathbf{E}[X|X \geq \delta/2] + \Pr[X < \delta/2]\mathbf{E}[X|X < \delta/2] \\ &\leq p + (1-p)(\delta/2) \leq p + \delta/2 \end{aligned}$$

■

Claim 4.4.8 *Assume that $|F| \geq \beta n$. There are at least $\beta^2 n/2$ vertices $s \in F$ such that: for at least $\beta\ell/8$ timesteps $t \in [\ell]$, $\widehat{M^t \vec{s}}(F) \geq \beta/16$.*

Proof Define $\theta_{s,t}$ as follows. For $s \in F$ and $t \in [\ell]$: if t is odd, $\theta_{s,t} = 0$. If t is even, then $\theta_{s,t}$ is the probability that a t-length (standard) random walk starting from s ends in F.

Let us pick a uar source vertex in $s \in F$, pick a uar length $t \in [\ell]$. We use the fact that M is a symmetric matrix.

$$\mathbf{E}_{s,t}[\theta_{s,t}] = \mathbf{1}_F^T \sum_{i=1}^{\ell/2} (M^{2i}/\ell)(\mathbf{1}_F/|F|) = \frac{\sum_{i \leq \ell/2}}{\ell|F|} \mathbf{1}_F^T M^{2i} \mathbf{1}_F = \frac{\sum_{i \leq \ell/2}}{\ell|F|} \|M^i \mathbf{1}_F\|_2^2 \quad (4.31)$$

Note that $\|M^i \mathbf{1}_F\|_1 = |F|$, so by Jensen's inequality, $\|M^i \mathbf{1}_F\|_2^2 \geq |F|^2/n$. Plugging in (4.31), $\mathbf{E}_{s,t}[\theta_{s,t}] \geq \ell^{-1} \times (\ell/2)|F|/n \geq \beta/2$. For any s, $\mathbf{E}_t[\theta_{s,t}] \leq 1$. By Fact 4.4.7, there are at least $\beta|F|/4$ vertices $s \in F$ such that $\mathbf{E}_t[\theta_{s,t}] \geq \beta/4$. Again applying Fact 4.4.7, for at least $\beta|F|/4$ vertices $s \in F$, there are at least $\beta\ell/8$ timesteps in $[\ell]$ such that $M^t\vec{s}(F) \geq \beta/8$.

By Lemma 4.3.3, there are at most $\rho^{1/8} n \leq \beta^2 n/8 \leq \beta|F|/8$ bad vertices. (Here, the first inequality follows by the choice of parameters). And moreover, for any good vertex, for all $t \in [\ell]$, $\|M^t\vec{s} - \widehat{M}^t\vec{s}\|_1 \leq 2\ell\rho^{7/20} \leq \beta/16$. Subtracting these bounds from the guarantees of the previous paragraph, there are at least $\beta|F|/8$ vertices $s \in F$ such that: for at least $\beta\ell/8$ timesteps in $[\ell]$, $\widehat{M}^t\vec{s}(F) \geq \beta/16$.

■

Lemma 4.4.9 *There are at least $\beta^2 n/2$ vertices $s \in F$, such that: there are at least $\beta\ell/16$ timesteps t in $[\ell]$ that are not leaking for s.*

Proof We fix any vertex s satisfying the conditions of Claim 4.4.8, and prove by contradiction. There are at least $\beta\ell/8 - \beta\ell/16 = \beta\ell/8$ timesteps t that are leaking for s, and such that $\widehat{M}^t\vec{s}(F) \geq \beta/16$. Fix any such timestep t, and consider the buckets $B_{t,r}$. There are at most $-\lg\rho$ buckets with non-zero probability mass, and by averaging, there exists $r \leq -\lg\rho$ such that $\sum_{v \in F \cap B_{t,r}} p_{s,t}(v) \geq \beta/(-16\lg\rho)$. By the choice of parameters, for any ε, we have $\sum_{v \in F \cap B_{t,r}} p_{s,t}(v) \geq \alpha$, and $B_{t,r}$ is heavy.

Thus, for each of the $\beta\ell/8$ leaking timesteps t obtained above, there exists some $r \leq -\lg\rho$ such that $B_{t,r}$ is heavy. By averaging, there exists some $r \leq -\lg\rho$ such that for $\beta\ell/(-8\lg\rho)$ leaking timesteps t, $B_{t,r}$ is heavy. But this means $\beta\ell/(-8\lg\rho) > 8\ell'/\alpha$, which contradicts Lemma 4.4.6. ■

Lemma 4.4.10 *Let $|F| \geq \beta n$. There exists a $r \leq \lg(2(\rho\alpha)^{-1})$ such that for at least $\beta^2 n/(2\lg(\rho\alpha)^{-1}$ vertices $s \in F$, the following holds. For $\geq \beta\ell/(32(\lg(\rho\alpha)^{-1}))$ timesteps t, there exists $k \in [2^r, 2^{r+1}]$ that is an (s,t)-certificate of non-leakiness.*

Proof This is an averaging argument. Apply Lemma 4.4.9. For each of the $\beta^2 n/2$ vertices $s \in F$, there are at least $\beta\ell/16$ timesteps t that are not leaking for s. Thus, for every such (s,t) pair, there exists $k_{s,t} \leq 2(\rho\alpha)^{-1}$ that is an (s,t)-certificate of non-leakiness. We basically bin the logarithm of the certificates. Thus, to every pair (s,t) (of the above form), we associate $r_{s,t} = \lfloor \lg k_{s,t} \rfloor$. By averaging, for each relevant s, there is a value r_s such that for at least $\beta\ell/(16\lg(2(\rho\alpha)^{-1}))$ timesteps t, there is an (s,t)-certificate in $[2^{r_s}, 2^{r_s+1}]$. By averaging again, there exists $r \leq \lg(2(\rho\alpha)^{-1})$ such that there are at least $\beta^2 n/\lg(2(\rho\alpha)^{-1})$ vertices $s \in F$ for which there exist at least $\beta\ell/(16\lg(2(\rho\alpha)^{-1}))$ timesteps t, such that there is an (s,t)-certificate for non-leakiness in $[2^r, 2^{r+1}]$. ■

Proof (of Theorem 4.2.5) The proof has two parts. In the first part (which uses Lemma 4.4.10), we argue that a non-zero r_h is output with high probability. The second part (which is a routine Chernoff bound calculation), we argue that any non-zero r_h that is output has the desired properties with high probability.

For convenience, we use $\xi := \lg(2(\rho\alpha)^{-1})$. By Lemma 4.4.10, there exists $r \leq \xi$ such that for at least $\beta^2 n/\xi$ vertices $s \in F_h$, there are at least $\beta\ell/(16\xi)$ timesteps t for which there is an (s,t)-certificate in $[2^r, 2^{r+1}]$. Fix this r. Refer to these timesteps as *safe* for s. For any s in consideration, the probability (over the choice of $t_{s,h}$) that $t_{s,h}$ is a safe timestep is at least $\beta/(16\xi)$. Call s in consideration *relevant* if $t_{s,h}$ is safe. Note that $F_h \subset V_h$. The probability that a uar vertex $s \in V_h$ (probability over the choice of $t_{s,h}$ and over choosing s) is relevant is at least $(\beta^2 n/\xi)/|V_h| \times \beta/(16\xi)$ $\geq \beta^3/(16\xi^2)$. Thus, the expected number of vertices in $s \in S_h$ such that $t_{s,h}$ is good is at least $\beta^3|S_h|/(16\xi^2) \geq \beta^{-2}$. By a multiplicative Chernoff bound (over the choices of various $t_{s,h}$s and of S_h), with probability $1 - \exp(-\beta^{-2}/12)$ there are at least $\beta^3|S_h|/(32\xi^2)$ vertices in S_h that are relevant.

Let us now uncoil the definition of relevant. We fixed r in the previous paragraph. Each relevant vertex s has the following property. There is an $(s, t_{s,h})$-certificate for non-leakiness in the range $[2^r, 2^{r+1}]$. Meaning, there is a $k \in [2^r, 2^{r+1}]$ such that such that $|L_{s,t,k} \cap F| \geq \alpha^2 k/6$ (which, by the choice of parameters, is at least $\beta^3 k$ for sufficiently small $\varepsilon > 0$) and $L_{s,t,k}$ has conductance at least $1/\ell^{1/3}$. Thus, a relevant vertex is labeled as an r-success in $\texttt{findr}(h)$, and there are at least $\beta^3|S_h|/(32\xi^2)$ vertices of S_h labeled as such. All in all, with probability at least $1 - \exp(-\beta^{-2}/12)$, $\texttt{findr}(h)$ returns a non-zero r_h.

We move to the second part of the proof, which asserts that (with high probability), an output non-zero r_h has the desired properties. Fix any $r \leq \xi$. Suppose that the number of vertices that are r-successes in V_h is at most $\beta^5 n$. Then, the expected number of r-successes in S_h is at most $\beta^5 n/|V_h| \times |S_h| \leq \beta^4|S_h|$.

(We use the lower bound $|V_h| \geq |F_h| \geq \beta n$.) By a Chernoff bound, for any $t > 2e\beta^4|S_h|$, $\Pr[\#$ r-successes in $S_h > t] < 2^{-t}$. The probabiility of there being more than $10\beta^4|S_h| \geq \beta^{-2}$ r-successes is $\exp(-\beta^{-2})$. By a union bound over all choices of r, with probability at least $1-\xi\exp(-\beta^{-2})$, if there are fewer than $\beta^5 n$ r-successes in V_h, then there are fewer than $10\beta^4|S_h|$ r-successes in S_h. Taking the contrapositive, with probability at least $1-\xi\exp(-\beta^{-2})$, if a non-zero r_h is output, there are at least $\beta^5 n$ r-successes in V_h.

By the first part, with probability at least $1-\exp(-\beta^{-2}/12)$, `findr`(h) returns a non-zero r_h. A union bound completes the proof. ■

4.5 Proofs of applications

The proofs here are quite straightforward and appear (in some form) in previous work. We sketch the proofs, and do not give out the specifics of the Chernoff bound calculations. Specifically, we mention Theorem 9.28 and its proof in [38], which contains these calculations.

Proof (of Theorem 4.0.2) For input graph G, we set up the partition oracle. Note that we can estimate the number of edges cut by random sampling. We pick a vertex u uniformly at random, pick a uar neighbor v, and call the partition oracle on u and v. If these lie in different components, the edge (u, v) is cut. By sampling $\Theta(1/\varepsilon)$, we can determine with high probability if more than εnd edges are cut by the partitioning (Chernoff bound). If so, we simply reject, since G is far from being in a minor-closed family.

Otherwise, we sample poly(ε^{-1}) uar vertices, and determine the component that each vertex belongs to. For each component, we directly determine if it belongs to $\mathcal{Q}$. (If there is an efficient algorithm, we can run that algorithm.) By a Chernoff bound, if G was ε-far from $\mathcal{Q}$, with high probability, one of the components would not be in $\mathcal{Q}$. Overall, the query complexity is poly$(d\varepsilon^{-1})$. ■

Proof (of Theorem 4.0.3) As with the previous proof, we set up the partition oracle. With high probability, at most $\varepsilon dn/c$ edges are cut by the partitioning given by the oracle, where c is the largest amount by which an edge addition/deletion changes f. We sample poly$(d\varepsilon^{-1})$ uar vertices and determine the component that each vertex belongs to. For each component, we compute f exactly. We take the sum of f-values, and rescale appropriately to get an additive εnd estimate for f. ■

5 SPECTRAL ALGORITHMS FOR COLORING PROBLEMS

5.1 Introduction

Given an undirected graph $G = (V, E)$, a k-coloring of G is a map $\chi : V \to [k]$ such that for all edges $\{u, v\} \in E$, we have $\chi(u) \neq \chi(v)$. Finding the minimum number of colors with which a graph can be colored, or even finding a coloring which uses few colors for a graph G which is promised to be 3-colorable has been a major open problem in the field of algorithm design. Starting from an early work of Wigderson [65] who showed how to color 3-colorable graphs with $O(\sqrt{n})$ colors, there has been a series of works using novel combinatorial ideas, as well as ideas based on semidefinite programming (SDP), which give algorithms for coloring 3-colorable graphs with fewer colors [66–72]. The most recent algorithm of [72] achieves a coloring with $o(n^{1/5})$ colors. On the hardness side, Dinur, Mossel and Regev [73] showed the hardness of coloring 3-colorable graphs with any constant number of colors, assuming a variant of the Unique Games Conjecture. It is also known [74, 75] to be NP-hard to find an independent set of size $n/9$ (which is a weaker goal than 9-coloring) in a graph G on n vertices, when G is promised to have a $(1 - \varepsilon)$-*partial* 3-coloring i.e., a coloring which properly colors the induced subgraph on at least $(1 - \varepsilon) \cdot n$ vertices.

There has also been a significant amount of research trying to recover a planted 3-coloring in special families of graphs [9, 76, 77]. Notably, an algorithm by Alon and Kahale [9], shows how to 3-color a random 3-colorable graph G (with high probability over the choice of G). The graphs in their model, denoted $\mathcal{G}_{3,p,n}$, are generated by dividing the vertices in three color classes of size $n/3$ each, and connecting each pair of vertices in distinct color classes independently with probability p. As their main result, [9] are able to recover the color classes in polynomial time with a small probability of failure even when p is as small as $3d/n$ for a constant d. Notice that in expectation a vertex in this graph will have degree $2d$ and will have d neighbors in each color class different from its own.

A generalization of the above result is to consider models with limited amount of randomness (semi-random) or arbitrary graphs with random-like properties (pseudorandom graphs). While both models seek to capture the minimal assumptions needed for the methods developed for random graphs, the study of pseudorandomness properties is also motivated by developing decompositions of worst-case objects into structured and pseudorandom parts [78]. The works by Blum and Spencer [76] already considered semi-random models for $p = n^{-(1-\delta)}$. Motivated by the notion of pseudorandomness and decompositions and pseudorandomness used in the subexponential algorithms for Unique Games [79, 80], Arora and Ge [71] showed how to find a large independent set in 3-colorable graphs with small threshold rank. A recent work of David and Feige [77] shows that even when the graph is pseudorandom (an expander) and the 3-coloring is arbitrary, one can recover the coloring on most vertices of the graph. They also show that when the coloring is random, it can be recovered for *all* vertices in the graph.

Our Results

In this work, we focus on showing that the first 2 steps in [9]'s proof can be adapted to (almost) 3-color a special family of pseudorandom graphs with a pseudorandom coloring (see Subsection 5.1.1 for a comparison with [77]). To state our results, we first define the relevant notions of pseudorandomness. Note that in the definitions below, we take the average degree of the graph to be $2d$, since we think of the degree in each color class as being d.

Definition 5.1.1 ***(Pseudorandom colorings)*** *Let* $G = (V, E)$ *be any graph with* $|V| = n$ *and* $|E| = d \cdot n$*. Let* $\chi\colon V \to \{1, 2, 3\}$ *be a (proper or improper) coloring of vertices with* 3 *colors. For a proper or improper coloring* χ*, we say*

Definition 5.1.2 *χ is a low variance coloring if for all $i, j \in \{1,2,3\}$, we have $\mathbf{var}_{v \in \mathrm{COL}_i} d_{ij}(v) \leq \varepsilon \cdot d^2$. If all balanced 3 colorings χ fail to have variance less than ε, we say that G only admits high variance colorings.*

χ is considered $(2d, \varepsilon)$-pseudorandom if

1. *χ is balanced i.e., all color classes ($1, 2, 3$ above) have the same size, $\frac{n}{3}$.*

2. *The coloring of G in the above step is a* low variance coloring.

The first family of pseudorandom graphs is low threshold-rank graphs, which (in the context of coloring) are defined as having a small number of negative eigenvalues with a large magnitude. This notion of pseudorandomness (with a different notion of the threshold) was also considered in the work of Arora and Ge [71].

Definition 5.1.3 ***(Threshold Rank)*** *A graph $G = (V, E)$ with $|E| = d \cdot n$. The threshold rank of G is defined to be the number of eigenvalues of the adjacency matrix that are smaller than $-9d/10$.*

We will also require the following notion of a partial coloring.

Definition 5.1.4 ***(Partial Coloring)*** *For a graph $G = (V, E)$, a function $\chi : V \to [3] \cup \{\perp\}$ is said to be a $(1-\gamma)$-partial 3-coloring if*

- *$|\chi^{-1}(\perp)| \ \leq \ \gamma \cdot n$.*

- *χ is a proper 3-coloring for the induced subgraph on $\chi^{-1}([3])$.*

Note that there exists a $(1-\gamma)$-partial 3-coloring χ of G if and only if there exits a *total* (but not necesarily proper) coloring $\chi' : V \to [3]$ such that χ' can be made a $(1-\gamma)$-partial 3-coloring by replacing the colors of γ fraction of vertices by $\perp$. Since our pseudorandomness properties are defined only for total colorings, we will abuse notation to say that there is a pseudorandom $(1-\gamma)$-partial 3-coloring if there exists a pseudorandom χ' which agrees with a $(1-\gamma)$-partial 3-coloring χ on $\chi^{-1}([3])$. We will refer to such a coloring as $(2d, \varepsilon)$-pseudorandom $(1-\gamma)$-partial 3-coloring. We now state the first theorem we prove.

Theorem 5.1.1 *There exists an algorithm which, given $G = (V, E)$ such that*

- $|E| = d \cdot n$ *and* $\mathsf{th}(G) = r$, *and*

- *there exists* χ, *which is* $(2d, \varepsilon)$*-pseudorandom and forms a* $(1-\gamma)$*-partial 3-coloring of* G,

runs in time $\left(\frac{\sqrt{r \cdot n}}{\varepsilon}\right)^r \cdot poly(n)$ *and w.h.p. returns a* $(1 - O(\varepsilon + \gamma))$*-partial* 3*-coloring of* G.

We also consider the futher specialized pseudorandom family, where the graphs will also be required to be expanding. Note that the graphs below capture the random family considered by Alon and Kahale [9]. For graphs in this family, we will recover all the color classes exactly.

Definition 5.1.5 ***(Expanding 3-colorable graphs)*** *A 3-colorable graph is said to be expanding if for some small positive constant* $\delta < 1$, $|\lambda_i| \leq \delta d$ *holds* $\forall\ 2 \leq i \leq n-2$ *i.e., if all eigenvalues other than the leading eigenvalue and the last two are small in magnitude.*

For graphs coming from this family, we have the following theorem.

Theorem 5.1.2 *There exists a polynomial-time algorithm which, given* $G = (V, E)$ *such that*

- G *is an expanding 3-colorable graph with* $|E| = d \cdot n$, *and*

- *there exists* $\chi : V \to [3]$, *which is* $(2d, \varepsilon)$*-pseudorandom and a proper 3-coloring of* G,

recovers χ.

Let G be an expanding 3-colorable graph. Note that this algorithm will recover a 3 coloring as long as there is some partition of $V(G)$ into color classes which have the low variance property (it does not have to be explicitly planted, as long as it exists Theorem 5.1.2 will find it.

5.1.1 Related work

There is a huge amount of literature devoted to find a complete or partial (legal) 3-coloring of an input graph under some assumptions on the graph and/or some assumptions on some 3-coloring of the given graph. Here we briefly examine works related to the current work.

The algorithm of Alon and Kahale.

In [9], Alon and Kahale described how to 3-color a random graph $G \sim G_{3,d/n,n}$. Thus, they have a random graph with a planted 3-coloring which they 3-color using a three phase algorithm. Their main result shows that graphs coming from $G_{3,d/n,n}$ have nice spectral properties which, in the first phase, can be exploited by a spectral clustering approach to find a good candidate coloring, χ_1, which misclassfies very few vertices. This candidate coloring is later refined to obtain a coloring χ_2 in the next phase. The second phase is a local search which locally improves the coloring on the set $H \subseteq V$ of vertices which have close to d neighbors in each color class according to the current coloring. This is then followed up by a cleanup phase which recolors vertices in $V \setminus H$ (all the components on the subgraph induced on these vertices have logarithmic sizes and can be brute forced upon).

We show how to construct χ_1 in low threshold-rank graphs which admit a pseudo-random partial coloring. When the graphs are also expanding admit a full coloring, we also show how to complete the second phase.

The results of David and Feige.

In [77], David and Feige considered extensions of the problem of coloring random 3-colorable graphs in several directions. In particular, they tried to relax the randomness assumption on the graph and the planted coloring inherent in [9]. To this end, they considered 4 models. They take as input an approximately d-regular (spectrally) expanding graph which can be either *adversarial* or *random* and a balanced planted coloring which again can be *adversarial or random.*

Our results are somewhat incomparable with theirs. Perhaps the most directly related setting from their work is the one where the graph is an arbitrary expander with an arbitrary (balanced) 3-coloring. In this case, they recover a $(1-\gamma)$ partial coloring. In this work, we start with additional pseudorandomness assumptions on the coloring (beyond balance), and can recover a partial coloring without assuming expansion in the input graph G, but making a different assumption about the negative eigenvalues. When G is also expanding in addition to these properties, we can recover the coloring completely.

Notation

Our notations are standard. We will write vectors and matrices in boldface (like $\boldsymbol{u}$ and $\boldsymbol{A}$ respectievely). For a graph G we will denote its adjacency matrix by $\boldsymbol{A} = \boldsymbol{A}(G)$. We denote a unit vector along the direction of $\boldsymbol{u}$ with $\hat{\boldsymbol{u}}$. And the transpose of a vector $\boldsymbol{u}$ is denoted $\boldsymbol{u}^T$. Also, $\mathbf{1}$ will denote the vector which is 1 in every coordinate. The stationaty distribution of random walks will be denoted $\boldsymbol{\mu}$. We denote the degree of a vertex $i \in V$ by $deg(i)$. The set of edges with one end point in a set S and the other in T is denoted $E(S,T)$.

5.2 Partially 3-coloring partially 3-colorable graphs

As a first step, we use lemma 5.2.1, to get a full coloring which only miscolors an $O(\varepsilon + \gamma)$ fraction of the vertices. In the next step, we will uncolor the incorrectly colored vertices. The first step is summarized by the following lemma.

Lemma 5.2.1 *There exists an algorithm which given a graph $G = (V, E)$ such that*

– $|E| = d \cdot n$ *and* $\mathsf{th}(G) = r$

– *there exists a coloring χ which is a $(2d, \varepsilon)$-pseudorandom coloring, and forms a $(1-\gamma)$-partial-3-coloring of G*

runs in time $O\left(\frac{\sqrt{r \cdot n}}{\varepsilon}\right)^r$ and returns a coloring which has $(1 - O(\varepsilon + \gamma))$ fraction of the vertices colored correctly, i.e., the graph induced on them has no monochromatic edges.

Following [9], we consider two special vectors which we call $\boldsymbol{x}$ and $\boldsymbol{y}$ as defined below.

$$\boldsymbol{x}(v) = \begin{cases} 2 & \text{if } v \in \mathrm{COL}_1, \\ -1 & \text{if } v \in \mathrm{COL}_2, \\ -1 & \text{if } v \in \mathrm{COL}_3, \end{cases} \qquad \boldsymbol{y}(v) = \begin{cases} 0 & \text{if } v \in \mathrm{COL}_1, \\ 1 & \text{if } v \in \mathrm{COL}_2, \\ -1 & \text{if } v \in \mathrm{COL}_3, \end{cases} \tag{5.1}$$

The point of these vectors is that they are both constant on all color classes and so are their linear combinations. Simlar to [9], we will try to find a vector which is close enough to some linear combination of $\boldsymbol{x}$ and $\boldsymbol{y}$ and use it to obtain a coloring of the kind Lemma 5.2.1 seeks. Now let us detail our algorithm.

Algorithm 1 Find Coloring

Require: Graph G with $\mathsf{th}(G) = r$ and the eigenvectors $\{\boldsymbol{v}_{n-r+1}, \boldsymbol{v}_{n-r+2}, \cdots \boldsymbol{v}_n\}$

1: Let $\boldsymbol{t} \leftarrow$ Subspace enumeration$\left(\sqrt{\frac{\varepsilon}{r}}\right)$

2: Let $\mathrm{COL}_1 = \{\ i \in V\ :\ \boldsymbol{t}_i > 1/2\ \}$, $\mathrm{COL}_2 = \{\ i \in V\ :\ \boldsymbol{t}_i < -1/2\ \}$ and $\mathrm{COL}_3 = V \setminus (\mathrm{COL}_1 \cup \mathrm{COL}_2)$

This algorithm relies on a procedure called Subspace Enumeration which is described below.

Algorithm 2 Subspace enumeration(τ)

Require: Graph G with $\mathsf{th}(G) = r$ and the eigenvectors $\{\boldsymbol{v}_{n-r+1}, \boldsymbol{v}_{n-r+2}, \cdots \boldsymbol{v}_n\}$

Let $B_r \leftarrow [-100\sqrt{n}, 100\sqrt{n}]^r$

$\triangleright$ B_r denotes a bounding box in the space of r eigenvectors above

Partition B_r into grid cells. Each cell has length τ in all dimensions.

$\triangleright$ The number of cells produced is $O\left(\frac{200\sqrt{n}}{\tau}\right)^r$

Let P_r denote the set of all corners of any grid cell.

$\triangleright$ Thus, $P_r = \{\, \boldsymbol{p} \in B_r \,:\, \tau$ divides all r coordinates in $\boldsymbol{p}\,\}$

Find in P_r a point $\boldsymbol{t}$ which has

- $med(\boldsymbol{t}) = 0$ and $\|\boldsymbol{t}'\|_2 = \Theta(\sqrt{n})$
- distance at most $\leq O\left(\sqrt{\varepsilon n + \gamma n + \tau^2 r}\right)$ from $span(\boldsymbol{x}, \boldsymbol{y})$.

$\triangleright$ We later show, in Claim 5.2.4 that such a vector exists.

return $\boldsymbol{t}$

To put this plan in motion, we make the following claims which are generalization of the corresponding claims in [9].

Claim 5.2.2 $\|\boldsymbol{A}\boldsymbol{x} + d(1-\gamma)\boldsymbol{x}\|_2^2, \|\boldsymbol{A}\boldsymbol{y} + d(1-\gamma)\boldsymbol{y}\|_2^2 \leq O\left(\varepsilon n d^2 + \gamma n d^2\right)$.

Claim 5.2.3 *There exist small shift vectors $\boldsymbol{s}_n$ and $\boldsymbol{s}_{n-1}$ with $\|\boldsymbol{s}_n\|_2^2, \|\boldsymbol{s}_{n-1}\|_2^2 \leq O\left(\varepsilon n + \gamma n\right)$ such that both $\boldsymbol{x} - \boldsymbol{s}_n$ and $\boldsymbol{y} - \boldsymbol{s}_{n-1}$ are the linear combinations of last r eigenvectors of $\boldsymbol{A}$.*

Claim 5.2.4 *Algorithm 2 finds a vector $\boldsymbol{t}$ in the span of $\{\boldsymbol{v}_{n-r+1}, \boldsymbol{v}_{n-r+2}, \ldots \boldsymbol{v}_n\}$ in time $O\left(\frac{\sqrt{n}}{\tau}\right)^r$ such that*

- *$med(\boldsymbol{t}) = 0$ and $\|\boldsymbol{t}'\|_2 = \Theta(\sqrt{n})$*
- *$\|\boldsymbol{t} - \boldsymbol{f}\|_2 \leq O\left(\sqrt{\varepsilon n + \gamma n}\right)$ where $\boldsymbol{f}$ is some vector that lies in $span(\boldsymbol{x}, \boldsymbol{y})$.*

Now, using these claims we will sketch how to establish Lemma 5.2.1. Taking cue from [9], we find a vector in the span of the last r eigenvectors, which we denote

as $span(\boldsymbol{v}_{n-r+1}, \boldsymbol{v}_{n-r+2}, \ldots, \boldsymbol{v}_n)$, a vector $\boldsymbol{t}$ which is close to a vector $\boldsymbol{f} \in span(\boldsymbol{x}, \boldsymbol{y})$ has length $\Theta(\sqrt{n})$ and median zero. In particular, the intuition is to have $\boldsymbol{t}$ (which the algorithm finds) be close to a vector $\boldsymbol{f}$ which has large positive entries indicating the first color class, large negative entries for the second color class and 0 entries for the last color class. In $\boldsymbol{t}$ hopefully the large positive entries and negative entries of $\boldsymbol{f}$ will remain away from zero and maintain their sign and the zero entries in $\boldsymbol{f}$ will hopefully remain close to 0. The remaining details of the proof are delegated to the last Section, section 5.5. The proof follows [9].
With the proof of Lemma 5.2.1, let us see how to prove Theorem 5.1.1.

Proof (Of Theorem 5.1.1)

Let $E_{bad} \subseteq E$ be the set of edges which have both endpoints with the same color in the coloring derived using the first step described in the proof of Lemma 5.2.1. Let $U_{bad} \subseteq V$ be the set of endpoints of these edges and consider the graph $G[U_{bad}]$ induced on these vertices. We note that the set U of vertices misclassified in Lemma 5.2.1 also forms a vertex cover in $G[U_{bad}]$. By the standard 2-approximation algorithm for vertex cover we can find a vertex cover $C \subseteq U_{bad}$ where $|C| \leq 2|U|$. And on "uncoloring" the vertices in the set C we obtain a partial coloring which omits only a $O(|U|)$ of the vertices – which is a $(1 - O(\varepsilon + \gamma))$-partial 3-coloring.

■

Now, let us prove Claim 5.2.2 and Claim 5.2.3.

Proof (Of Claim 5.2.2) Let us prove this for the vector $\boldsymbol{x}$. The proof with vector $\boldsymbol{y}$ is similar. Let $\boldsymbol{u} = \boldsymbol{A}\boldsymbol{x} + d(1-\gamma)\boldsymbol{x}$. Let $\boldsymbol{a}_i^T$ denote the i^{th} row in $\boldsymbol{A}$. Thus, $\boldsymbol{u}_i = \boldsymbol{a}_i^T \boldsymbol{x} + d\boldsymbol{x}_i$. Let us say that in the vector $\boldsymbol{x}$, $\boldsymbol{x}_i = 2$ for $i \in \mathrm{COL}_1$, and it is -1 otherwise. Note that for $i \in \mathrm{COL}_1$, we get $\boldsymbol{u}_i^2 = (\boldsymbol{a}_i^T \boldsymbol{x} + d(1-\gamma)\boldsymbol{x}_i)^2$

$$= \left(-d_{12}(i) - d_{13}(i) + 2d_{11}(i) + 2d(1-\gamma)\right)^2 \leq O\left(M_1(i)\right)$$

where $M_1(i) := (d_{12} - d)^2 + (d_{13} - d)^2 + (d_{11} - \gamma d)^2$. In a similar fashion, we see that for $i \in \mathrm{COL}_2$ $\boldsymbol{u}_i^2 \leq O\left(M_2(i)\right)$ where $M_2(i) = (d_{21}(i) - d)^2 + (d_{23}(i) - d)^2 + (d_{22} - \gamma d)^2$. And an analogous upper bound holds for $\boldsymbol{u}_i^2$ when $i \in \mathrm{COL}_3$. Thus,

$$\|\boldsymbol{u}\|_2^2 = \sum \boldsymbol{u}_i^2 \le \sum_{i\in \mathrm{COL}_1} M_1(i) + \sum_{i\in \mathrm{COL}_2} M_2(i) + \sum_{i\in \mathrm{COL}_3} M_3(i) \le \ O\left(\varepsilon n d^2 + \gamma n d^2\right)$$

as the $\mathbf{var}_{v\in \mathrm{COL}_i} d_{ii}(v) \le \varepsilon d^2$ as well. The proof for upperbound on $\|\boldsymbol{A}\boldsymbol{y} + d\boldsymbol{y}\|_2^2$ is similar. ■

And next, we prove Claim 5.2.3 as well. Here is a quick adaptation of [9]'s proof.

Proof (Of Claim 5.2.3) Write $\boldsymbol{x} = \sum c_i \boldsymbol{v}_i$ in the eigenbasis. Again let $\boldsymbol{u} = \boldsymbol{A}\boldsymbol{x} + d(1-\gamma)\boldsymbol{x}$.

$$\|\boldsymbol{u}\|_2^2 = \sum_{i=1}^{n} c_i^2(\lambda_i + d(1-\gamma))^2 \ge \sum_{i=1}^{n-r} c_i^2(\lambda_i + d(1-\gamma))^2 \ge \Omega(d^2)\sum_{i=1}^{n-r} c_i^2$$

The last step above follows because the first $n-r$ eigenvalues are all at least $-9d/10$. And together with the fact that $\|\boldsymbol{u}\|_2^2 \le O\left(\varepsilon n d^2 + \gamma n d^2\right)$, we get $\sum_{i=1}^{i=n-r} c_i^2 \le O\left(\varepsilon n + \gamma n\right)$. And this is the shift vector $\boldsymbol{s}_n$ we seek with the desired bound on its length. A similar argument can be made to find $\boldsymbol{s}_{n-1}$ and its length using the vector $\boldsymbol{y}$.

■

Proof (Of Claim 5.2.4)

Let I denote the set of indices that are left uncolored in the hidden partial coloring. Let us define a vector $\boldsymbol{z}$ as $\boldsymbol{z}(i) = -1$ for $i \in \mathrm{COL}_1$, $\boldsymbol{z}(i) = 0$ for $i \in I \cup \mathrm{COL}_2$ and $\boldsymbol{z}(i) = +1$ for $i \in \mathrm{COL}_3$. Note that there exists a vector (say $\boldsymbol{f}$) in $span(\boldsymbol{x}, \boldsymbol{y})$ such that $\boldsymbol{z}$ and $\boldsymbol{f}$ are pretty close – the distance is at most $\sqrt{\varepsilon n + \gamma n}$. Further, there also exists a vector $\boldsymbol{t} \in span(\boldsymbol{v}_{n-r+1}, \boldsymbol{v}_{n-r+2}, \cdots, \boldsymbol{v}_n)$ such that $\|\boldsymbol{t} - \boldsymbol{z}\|_2 \le O(\sqrt{\varepsilon n + \gamma n}$ by arguments similar to Claim 5.2.2 and Claim 5.2.3. And therefore, the distance $\|\boldsymbol{t} - \boldsymbol{f}\|_2 \le O\left(\sqrt{\varepsilon n + \gamma n}\right)$. And finally, we also note that $med(\boldsymbol{t}) = 0$ and $\|\boldsymbol{t}\|_2 = \Theta(\sqrt{n})$.

This only proves that such a vector exists. We need to algorithmically *find* one. To this end, we use the *subspace enumeration* procedure described earlier. Given any vector $\boldsymbol{p} \in B_r$ (which, recall, was defined in Algorithm 2), this procedure finds a vector $\boldsymbol{t}$ which is within a distance $\tau\sqrt{r}$ of $\boldsymbol{p}$. In particular, this also holds for $\boldsymbol{t}'$. Moreover, $\boldsymbol{t}'$ being close to $\boldsymbol{z}$ has several 0 entries one of which is the median. And the vector $\boldsymbol{t}$ being $\tau\sqrt{r}$ close to $\boldsymbol{t}'$ also has $med(\boldsymbol{t}) = 0$, length $\Theta(\sqrt{n})$ and has $\|\boldsymbol{t} - \boldsymbol{f}\|_2 \leq O\left(\sqrt{\varepsilon n + \gamma n + \tau^2 r}\right)$ as can be easily seen by triangle inequality.

■

5.3 Coloring expanding-3-colorable Graphs

In the case of $(2d, \varepsilon)$-expanding-3-colorable, $\gamma = 0$. So, the coloring obtained by the first step of the algorithm returns a $(1 - O(\varepsilon))$ partial coloring. We briefly review what [9] do in the second step for 3-coloring a random 3-colorable graph $G \in G_{3,d/n,3n}$. Their algorithm receives as input a 3-colored graph G and a set U of *bad* vertices (with $|U| = O(n/d)$) that have been incorrectly colored. This bad 3 coloring of G is improved via an iterative process. Each step in the iteration reduces the number of bad vertices by a constant factor. The algorithm is given below, and we use the same algorithm. However, our analysis is different.

Algorithm 3 Improve Coloring

Require: A $(1 - O(\varepsilon))$partial 3-coloring of G vertices.

1: Let the current color classes be denoted V_1^0, V_2^0 and V_3^0
2: Add the uncolored vertices arbitrarily to the partitions in any order.
3: **for** $i = 1 : \log n$ **do**
4: For $j \in \{1, 2, 3\}$, put $v \in V_j^i$ only if $|N(v) \cap V_j^{i-1}| \leq |N(v) \cap V_l^{i-1}|$ for all $l \neq j$
5: ▷ i.e.,, put v in the least popular color class among its neighbors from previous iteration
6: ▷ V_1^i, V_2^i, V_3^i denotes the color classes in the i^{th} iteration
7: **end for**

The above algorithm derives from the following intuition. Given a 3-colored graph $G \in G_{3,d/n,3n}$ with a small set U of bad vertices, a "local search procedure" can recolor those vertices in V which do not have way too many neighbors in U. Thus, another way to express the intuition is to say that a bad vertex at the beginning of iteration i remains bad after iteration i finishes only if it is surrounded by many bad vertices. To make this formal, let us consider the set

$$W = \{\, v \in V \;:\; deg_U(v) \geq d/4 \,\}\,.$$

W will be referred to as being *U-rich*. The main step in the argument is to show that the size of U-rich set is at most $\frac{|U|}{2}$ and that at the end of every iteration the set which remains incorrectly colored is only a subset of the U-rich set. We will now use this argument in the case of expanding graphs. This is done in the following lemma.

Lemma 5.3.1 *There exists a polynomial time algorithm which on input a $(2d, \varepsilon)$-expanding-3-colorable graph $G = (V, E)$ (recall that this means for some absolute positive constant $\delta < 1$, $|\lambda_i| \leq \delta d$ for $2 \leq i \leq n-2$) and which admits a $(1 - O(\varepsilon))$ partial 3-coloring of V, a proper 3 coloring of V.*

We will prove Lemma 5.3.1 by using the expander mixing lemma. The key here would be to again show that $|U|$-rich sets have small size if $|U|$ is small. This is done in the following claim.

Claim 5.3.2 *Let $U \subseteq V$ be a set with size at most $O(\varepsilon n)$. Let*

$$W = \{\, v \in V \;:\; deg_U(v) \geq 2d/5 \,\}\,.$$

Then $|W| \leq 0.99|U|$.

Proof (Of Claim 5.3.2) We begin by finding upper bounds and lower bounds for $|E(U, W)|$. We see that

$$|E(U, W)| \geq \frac{1}{2} \cdot \frac{2d|W|}{5} = \frac{d|W|}{5}.$$

The inequality here comes because there are $\frac{2d}{5}$ edges incident on every vertex in $|W|$ with at least one endpoint in U. The factor of $1/2$ compensates for the fact that both the endpoints of the edge may belong to U.

Now, we need to upper bound $|E(U,W)|$. To do this, we use expander mixing lemma. Let the indicator vector for U be denoted $\mathbb{1}_{\mathbf{U}}$ and the indicator vector for W be denoted $\mathbb{1}_{\mathbf{W}}$. Let us write $\mathbb{1}_{\mathbf{U}} = \sum \alpha_i \boldsymbol{v}_i$ and $\mathbb{1}_{\mathbf{W}} = \sum \beta_i \boldsymbol{v}_i$. We know

$$|E(U,W)| = \mathbb{1}_{\mathbf{U}}{}^T \boldsymbol{A} \mathbb{1}_{\mathbf{W}} = |\left(\sum \alpha_i \boldsymbol{v}_i^T\right) \boldsymbol{A} \left(\sum \beta_j \boldsymbol{v}_j\right)| = |\sum (\alpha_i \beta_i \lambda_i)|$$

And it is immediately seen that the following holds

$$|E(U,W)| \leq |\alpha_1 \beta_1 \lambda_1| + |\alpha_n \beta_n| \cdot |\lambda_n| \ + |\alpha_{n-1}\beta_{n-1}| \cdot |\lambda_{n-1}| + \sum_{i=2}^{n-2} |\alpha_i \beta_i \lambda_i| \tag{5.2}$$

Now, we upper bound the RHS of Equation 5.2. We show how to bound each of the summands seprately. Let us begin by bounding the first summand. For intuition sake consider the $\varepsilon = 0$ case. Then in fact we have a $2d$ regular graph and $\boldsymbol{v}_1 = \boldsymbol{u}_n$ where $\boldsymbol{u}_n = \frac{1}{\sqrt{n}} \cdot \mathbf{1}$ denotes the normalized uniform distribution vector.

$$\begin{aligned} |\alpha_1 \beta_1 \lambda_1| \leq & |\mathbb{1}_{\mathbf{U}}{}^T \boldsymbol{v}_1| \cdot |\mathbb{1}_{\mathbf{W}}{}^T \boldsymbol{v}_1| \cdot 2d \\ & = \frac{|U|}{\sqrt{n}} \cdot \frac{|W|}{\sqrt{n}} \cdot 2d = \frac{|U| \cdot |W| \cdot 2d}{n} \end{aligned} \tag{5.3}$$

In case $\varepsilon > 0$, we will show that something close holds. In particular, we will show that the vector $\boldsymbol{v}_1$ has a large dot product with $\boldsymbol{u}_n$. Denote by $\boldsymbol{\mu}$ the stationary distribution vector for random walks on G (and thus, $\boldsymbol{\mu}_i = \frac{deg(i)}{\sum deg(i)}$). Now, note

$$\begin{aligned} \left(\frac{\boldsymbol{v}_1^T \mathbf{1}}{\sqrt{n}}\right)^2 &= \left(\frac{1}{\sqrt{n}} \cdot \left(\frac{\boldsymbol{\mu}}{\|\boldsymbol{\mu}\|_2}\right)\right)^2 = \frac{1}{n} \cdot \frac{(\sum \boldsymbol{\mu}_i)^2}{\|\boldsymbol{\mu}\|_2^2} = \frac{1}{n} \cdot \frac{1}{\|\boldsymbol{\mu}\|_2^2} = \frac{(\sum deg(i))^2}{n \sum deg(i)^2} \\ &= \frac{(\mathbf{E} deg(i))^2}{\mathbf{E} deg(i)^2} = \frac{1}{1 + \dfrac{\mathbf{var}_{i \sim V} deg(i)}{\mathbf{E} deg(i)^2}} \geq \frac{1}{1 + 4\varepsilon d^2/d^2} = \frac{1}{1+4\varepsilon} \\ &\geq 1 - 8\varepsilon \end{aligned}$$

Here the first inequality follows because of Claim 5.3.3 which is proved later in this section.

Claim 5.3.3 $\mathbf{var}_{i \sim V} deg(i) \leq 4\varepsilon d^2$.

This means that the vector $\boldsymbol{v}_1$ is very close to the vector $\mathbf{1}/\sqrt{n}$. Now, we know that vector $\mathbb{1}_{\mathbf{U}}{}^T \frac{\mathbf{1}}{\sqrt{n}}$ is small. And we know that $\frac{\boldsymbol{v}_1^T \mathbf{1}}{\sqrt{n}}$ is large. It follows from this that $\mathbb{1}_{\mathbf{U}}{}^T \boldsymbol{v}_1$ will also be fairly small. In particular, we get

$$|\alpha_1 \beta_1 \lambda_1| \leq 5 \frac{|U|}{\sqrt{n}}(1+8\varepsilon) \cdot \frac{|W|}{\sqrt{n}}(1+8\varepsilon) \cdot 2d \leq \frac{|U| \cdot |W| \cdot 2d(1+20\varepsilon)}{n} \tag{5.4}$$

Now, we bound the second summand.

As we will see in Claim 5.5.1, we have $\sqrt{n}\boldsymbol{v}_n$ can be expressed as a linear combination of vectors $\boldsymbol{x} - \boldsymbol{s}_n$ and $\boldsymbol{y} - \boldsymbol{s}_{n-1}$ with coefficients $O(1)$ in absolute value. So, let us write $\sqrt{n}\boldsymbol{v}_n = a_1(\boldsymbol{x} - \boldsymbol{s}_n) + b_1(\boldsymbol{y} - \boldsymbol{s}_{n-1})$ with a_1 and b_1 being constants. A similar expression can be obtained for $\sqrt{n}\boldsymbol{v}_{n-1}$. Now, we claim

Claim 5.3.4 $|\alpha_n \beta_n \lambda_n| \leq O\left(d \cdot \left(\frac{|U||W|}{n} + \frac{|U|\sqrt{|W|}\sqrt{\varepsilon}}{\sqrt{n}} + \frac{|W|\sqrt{|U|}\sqrt{\varepsilon}}{\sqrt{n}} + \varepsilon\sqrt{|U||W|}\right)\right)$

Proof (Of Claim 5.3.4)

$$\begin{aligned} |\alpha_n \beta_n \lambda_n| &\leq \frac{1}{\sqrt{n}} |\mathbb{1}_{\mathbf{U}}{}^T \sqrt{n}\boldsymbol{v}_n| \cdot \frac{1}{\sqrt{n}} |\mathbb{1}_{\mathbf{W}}{}^T \sqrt{n}\boldsymbol{v}_n| \cdot 2d \\ &\leq \frac{|\mathbb{1}_{\mathbf{U}}{}^T a_1(\boldsymbol{x} - \boldsymbol{s}_n) + \mathbb{1}_{\mathbf{U}}{}^T b_1(\boldsymbol{y} - \boldsymbol{s}_{n-1})|}{\sqrt{n}} \\ &\quad \times \frac{|\mathbb{1}_{\mathbf{W}}{}^T a_2(\boldsymbol{x} - \boldsymbol{s}_n) + \mathbb{1}_{\mathbf{W}}{}^T b_2(\boldsymbol{y} - \boldsymbol{s}_{n-1})|}{\sqrt{n}} \cdot 2d \end{aligned} \tag{5.5}$$

We now need to understand how to bound terms like $\frac{1}{\sqrt{n}} |\mathbb{1}_{\mathbf{U}}{}^T a_1(\boldsymbol{x} - \boldsymbol{s}_n)|$. To this end, note that we have the following.

1. $|\mathbb{1}_{\mathbf{U}}{}^T a_1(\boldsymbol{x} - \boldsymbol{s}_n)| \leq |\mathbb{1}_{\mathbf{U}}{}^T a_1 \boldsymbol{x}| + |a_1| \|\boldsymbol{s}_n\|_2 \|\mathbb{1}_{\mathbf{U}}\|_2 \leq |\mathbb{1}_{\mathbf{U}}{}^T a_1 \boldsymbol{x}| + |a_1|\sqrt{\varepsilon}\sqrt{n}\sqrt{|U|}$.

2. $|\mathbb{1}_{\mathbf{U}}{}^T \boldsymbol{x}| = |2U_1 - U_2 - U_3| \leq 2|U|$

   Here U_1 refers to number of vertices colored with COL_1 in U. Other terms have analogous meaning.

Putting all of this together and absorbing all constants in the $O(.)$, we get

$$|\alpha_n \beta_n \lambda_n| \leq O\left(d \cdot \left(\frac{|U|}{\sqrt{n}} + \sqrt{\varepsilon}\sqrt{|U|}\right) \cdot \left(\frac{|W|}{\sqrt{n}} + \sqrt{\varepsilon}\sqrt{|W|}\right)\right) \tag{5.6}$$

■

The $3rd$ summand has an analogous bound to the one above. That is, we have

$$|\alpha_{n-1}\beta_{n-1}\lambda_{n-1}| \leq O(|\alpha_n \beta_n \lambda_n|) \tag{5.7}$$

The last summand can be simply bounded by using Cauchy Schwartz and the fact that all other eigenvalues are at most δd for some sufficiently small δ. This gives

$$\sum_{i=1}^{n-2} |\alpha_i \beta_i \lambda_i| \leq \delta d \cdot \sqrt{|U| \cdot |W|} \tag{5.8}$$

Putting all of these bounds Equation 5.4, Equation 5.6, Equation 5.7, Equation 5.8 together we get and upper bound on $|E(U, W)|$. Also, we already computed a lower bound on $|E(U, W)| \geq d|W|/5$. Chaining the upper bound and the lower bound thus obtained, using the fact that $|U| \leq O(\varepsilon n)$ we get

$$\begin{aligned}
\frac{\sqrt{|W|}}{5} &\leq \frac{2|U|\sqrt{|W|}(1+20\varepsilon)}{n} + O\left(\frac{|U|\sqrt{|W|}}{n} + \frac{\sqrt{\varepsilon|U||W|}}{\sqrt{n}} + \frac{\sqrt{\varepsilon}|U|}{\sqrt{n}} + \varepsilon\sqrt{|U|}\right) \\
&\quad + \delta\sqrt{|U|} \\
\implies \frac{\sqrt{|W|}}{10} &\leq O\left(\varepsilon\sqrt{|U|} + \delta\sqrt{|U|}\right) \\
\implies |W| &\leq O\left(\delta^2 + \varepsilon^2\right)|U|.
\end{aligned}$$

This finishes the proof of Claim 5.3.2. ■

Proof (Of Lemma 5.3.1) Observe that by Claim 5.3.2, it follows that the set of badly colored vertices shrinks significantly in each step of the local search. Thus, after $O(\log |U|)$ many steps, we do not have any bad vertices and we get a proper 3-coloring. This finishes the description of the algorithm for adapting the second step of [9] algorithm which finishes the proof of Lemma 5.3.1 ∎

Proof (Of Claim 5.3.3) Note that here we are taking variance in the degree of a random vertex over the full graph whereas by definition in $(2d, \varepsilon)$-expanding-3-colorable graph case we only know $\mathbf{var}_{v \in \mathrm{COL}_j} d_{ij}(v) \leq \varepsilon d^2$. So, a little more work is needed. Let us begin by noting that $\mathbf{E}deg(i)^2 = \sum_{j \in \{1,2,3\}} \mathbf{E}deg(i)^2 | i \in \mathrm{COL}_j/3$. Also, note $\mathbf{E}degi|i \in \mathrm{COL}_1 = \mathbf{E}deg(i)|i \in \mathrm{COL}_2 = \mathbf{E}deg(i)|i \in \mathrm{COL}_3 = d$. Let $M_1 = \mathbb{E}_{i \in \mathrm{COL}_1}[(d_{12}(i) + d_{23})^2]$ and $N_1 = \mathbb{E}_{i \in \mathrm{COL}_1}[(d_{12}(i)^2 + d_{13}(i)^2)]$ and similarly define M_2, N_2 and M_3, N_3. So, writing $\mathbf{var}_{i \sim V} deg(i) = \mathbf{E}deg(i)^2 - \mathbf{E}deg(i)^2$ and expanding by using the foregoing expressions, we get

$$
\begin{aligned}
\mathbf{var}_{i \sim V} &= \sum_{j \in \{1,2,3\}} \frac{\mathbf{E}deg(i)^2 | i \in \mathrm{COL}_j - 4d^2}{3} \\
&= \sum_{j \in \{1,2,3\}} \frac{M_j - 4d^2}{3} \\
&\leq \sum_{j \in \{1,2,3\}} \frac{2N_j - 4d^2}{3} \\
&= \frac{2(\mathbf{var}d_{12} + \mathbf{var}d_{13} + \ldots \mathbf{var}d_{32} + \mathbf{var}d_{31})}{3} \\
&\leq 4\varepsilon d^2
\end{aligned}
$$

∎

5.4 Future Work

Our work also suggests the following tantalizing conjecture for graph 3 coloring.

Conjecture 5.4.1 *Fix $\delta, \varepsilon > 0$. Then there exits an algorithm which when given an expanding graph G which as input, runs in time $\exp(n^\delta) \cdot \left(\frac{1}{\varepsilon}\right)^{1/\delta}$ and returns a coloring χ such that*

- *No edges are violated under χ.*
- *At most εn vertices are left uncolored.*

Suppose we are given a graph G as input which we know admits a low-variance coloring. Then by Theorem 5.1.2 we can find a partial 3-coloring of G in polynomial time. What about graphs which only admit high variance colorings Definition 5.1.2? The conjecture above states that this can be done in subexponential time in such graphs.

Acknowledgements We thank Elena Grigorescu for helpful discussions.

5.5 Missing Proofs

Proof (Of Lemma 5.2.1)

Obtain vectors $\boldsymbol{t}$ and $\boldsymbol{f}$ from Claim 5.2.4. These vectors have several useful properties which will be useful for us in what follows. Write $\boldsymbol{t} = \boldsymbol{f} + \boldsymbol{w}$. We have $\|\boldsymbol{w}\|_2 \leq O\left(\sqrt{\varepsilon n + \gamma n + \tau^2 r}\right)$.

Now, let $\alpha_i = \boldsymbol{f}\big|_{\mathrm{COL}_i}$ for $1 \leq i \leq 3$ – that is, α_i's are scalars and equal the constant value that $\boldsymbol{f}$ takes over the i^{th} color class. Assume $\alpha_1 \geq \alpha_2 \geq \alpha_3$. Using the fact that $\|\boldsymbol{w}\|^2 \leq O(\varepsilon n + \gamma n + \tau^2 r)$, it follows that only $O\left(\varepsilon n + \gamma n + \tau^2 r\right)$ coordinates in $\boldsymbol{w}$ can take on values large in magnitude (at least 0.01). Note that this means $|\alpha_2| \leq 1/4$. This is because, we have $\boldsymbol{t} - \boldsymbol{f} = \boldsymbol{w}$ and only a few entries in $\boldsymbol{w}$ can be big in magnitude. In more detail, suppose if $\alpha_2 > 1/4$ were to hold then at least $2n - O\left(\varepsilon n + \gamma n + \tau^2 r\right)$ entries in $\boldsymbol{t}$ will be big too and thus will have the same sign. This contradicts the 0 median assumption in $\boldsymbol{t}$. An analogous argument prevents $\alpha_2 < -1/4$.

Finally, recall that $\alpha_1 + \alpha_2 + \alpha_3 = 0$ (since $\boldsymbol{f} \in span(\boldsymbol{x}, \boldsymbol{y})$) and

$$\alpha_1^2 + \alpha_2^2 + \alpha_3^2 = \frac{\|\boldsymbol{f}\|^2}{n/3} \geq \frac{\|\boldsymbol{t}\|^2 - \|\boldsymbol{w}\|^2}{n/3} \geq 6 - O\left(\varepsilon + \gamma + \tau^2 r/n\right).$$

Together with the fact that $|\alpha_2| \leq 1/4$, this implies that $\alpha_1 \geq 3/4$ and $\alpha_3 \leq -3/4$. This means that $\boldsymbol{t}$ which is obtained by corrupting $\boldsymbol{f}$ on at most $O(\varepsilon n)$ entries coming from the noisy vector is a pretty good coloring. And the number of misclassified vertices in the coloring according to $\boldsymbol{t}$ is at most $O(\varepsilon n + \gamma n + \tau^2 r)$ – the ℓ_2^2 length of the noisy vector $\boldsymbol{w}$.

And now note that setting $\tau = \sqrt{\frac{\varepsilon}{r}}$ prodcues $\boldsymbol{w}$ with $\|w\|_2^2 = O\left(\varepsilon n + \gamma n\right)$. And the size of the discrete subspace that we search over is at most $O\left(\frac{\sqrt{r \cdot n}}{\varepsilon}\right)^r$ as claimed. And this finishes the proof.

■

Claim 5.5.1 *The vectors $\sqrt{n}\boldsymbol{v}_n$ and $\sqrt{n}\boldsymbol{v}_{n-1}$ can be expressed as a linear combination of vectors $\boldsymbol{x} - \boldsymbol{s}_n$ and $\boldsymbol{y} - \boldsymbol{s}_{n-1}$ with coefficients at most a constant in absolute value.*

Proof (Of Claim 5.5.1)

Let us first quickly see that $\boldsymbol{v}_n$ lies in $span(\boldsymbol{x} - \boldsymbol{s}_n, \boldsymbol{y} - \boldsymbol{s}_{n-1})$. By definition of $(2d, \varepsilon)$-expanding-3-colorable graphs, we have that all eigenvalues other than λ_1, λ_{n-1} and λ_n are small in absolute value. So, there exist tiny shift vectors $\boldsymbol{s}_n, \boldsymbol{s}_{n-1}$ (according to Claim 5.2.3) such that both $\boldsymbol{x} - \boldsymbol{s}_n, \boldsymbol{y} - \boldsymbol{s}_{n-1}$ lie in $span(\boldsymbol{v}_n, \boldsymbol{v}_{n-1})$. Thus, the space spanned by the last 2 eigenvectors is a tiny perturbation of $span(\boldsymbol{x}, \boldsymbol{y})$. So, any vector in $span(\boldsymbol{v}_n, \boldsymbol{v}_{n-1})$ also lies in $span(\boldsymbol{x} - \boldsymbol{s}_n, \boldsymbol{y} - \boldsymbol{s}_{n-1})$. And so, does $\sqrt{n}\boldsymbol{v}_n$.

Intuitively, the at most constant in absolute value part should follow because $\boldsymbol{x} - \boldsymbol{s}_n, \boldsymbol{y} - \boldsymbol{s}_{n-1}, \sqrt{n}\boldsymbol{v}_n, \sqrt{n}\boldsymbol{v}_{n-1}$ have length $\Theta(\sqrt{n})$ and $\boldsymbol{x} - \boldsymbol{s}_n, \boldsymbol{y} - \boldsymbol{s}_{n-1}$ are nearly orthogonal. This is because $\boldsymbol{x}$ and $\boldsymbol{y}$ are orthogonal and $\boldsymbol{s}_n$ and $\boldsymbol{s}_{n-1}$ are very tiny perturbations to these orthogonal vectors. In more detail,

$$
\begin{aligned}
&\sqrt{n}\boldsymbol{v}_n = \alpha(\boldsymbol{x} - \boldsymbol{s}_n) + \beta(\boldsymbol{y} - \boldsymbol{s}_{n-1}) \\
\Longrightarrow\ &\alpha\boldsymbol{x} + \beta\boldsymbol{y} = \ \sqrt{n}\boldsymbol{v}_n + \alpha\boldsymbol{s}_n + \beta\boldsymbol{s}_{n-1} \\
\Longrightarrow\ &\|\alpha\boldsymbol{x} + \beta\boldsymbol{y}\|_2 \leq \ \sqrt{n} + |\alpha| \cdot \|\boldsymbol{s}_n\|_2 + |\beta| \cdot \|\boldsymbol{s}_{n-1}\|_2
\end{aligned} \tag{5.9}
$$

On taking squared ℓ_2 norms on both sides of (5.9), it follows that

$$\frac{6n\alpha^2}{3} + \frac{2n\beta^2}{3} \leq n + O(\alpha^2 + \beta^2)\varepsilon n.$$

On simplifying it is clear that this means both $|\alpha|$ and $|\beta|$ are $O(1)$.

■

REFERENCES

[1] D. Spielman and S.-H. Teng. A local clustering algorithm for massive graphs and its application to nearly-linear time graph partitioning. *SIAM Journal on Computing*, 42(1):1–26, 2012. 1

[2] Luca Trevisan. Lecture notes on expansion, sparsest cut, and spectral graph theory. http://hortonworks.com/blog/snapshots-for-hdfs/. 1

[3] László Lovász and Miklós Simonovits. The mixing rate of markov chains, an isoperimetric inequality, and computing the volume. In *Foundations of Computer Science (FOCS)*, pages 346–354, 1990. 1, 13, 20, 46, 47, 73, 74, 83, 85, 102

[4] Sanjeev Arora, Boaz Barak, and David Steurer. Subexponential algorithms for unique games and related problems. *Journal of the ACM*, 62(5):42:1–42:25, 2015. 1, 4, 13

[5] Shayan Oveis Gharan and Luca Trevisan. Approximating the expansion profile and almost optimal local graph clustering. In *Foundations of Computer Science (FOCS)*, pages 187–196. IEEE Computer Society, 2012. 1

[6] Akash Kumar, C. Seshadhri, and Andrew Stolman. Finding forbidden minors in sublinear time: A $\mathrm{o}(\mathrm{n}^{1/2\,+\,\mathrm{o}(1)})$-query one-sided tester for minor closed properties on bounded degree graphs. In *Foundations of Computer Science (FOCS)*, pages 509–520, 2018. 1, 4, 57, 65, 67, 83

[7] Akash Kumar, C. Seshadhri, and Andrew Stolman. Random walks and forbidden minors II: a poly($d\,\epsilon^{-1}$)-query tester for minor-closed properties of bounded degree graphs. In *STOC 2019, Phoenix, AZ, USA, June 23-26, 2019.*, pages 559–567, 2019. 1, 3, 4, 83, 99

[8] Noga Alon, Michael Krivelevich, and Benny Sudakov. Finding a large hidden clique in a random graph. *Random Struct. Algorithms*, 13(3-4):457–466, 1998. 1

[9] Noga Alon and Nabil Kahale. A spectral technique for coloring random 3-colorable graphs. *SIAM J. Comput.*, 26(6):1733–1748, 1997. 1, 4, 113, 114, 116, 117, 119, 120, 121, 122, 123, 128

[10] Akash Kumar, Anand Louis, and Madhur Tulsiani. Finding pseudorandom colorings of pseudorandom graphs. In *37th IARCS FSTTCS 2017, December 11-15, 2017, Kanpur, India*, pages 37:1–37:12, 2017. 1, 6

[11] Boaz Barak, Samuel B. Hopkins, Jonathan A. Kelner, Pravesh K. Kothari, Ankur Moitra, and Aaron Potechin. A nearly tight sum-of-squares lower bound for the planted clique problem. *SIAM J. Comput.*, 2019. 2

[12] Samuel B. Hopkins, Pravesh K. Kothari, Aaron Potechin, Prasad Raghavendra, Tselil Schramm, and David Steurer. The power of sum-of-squares for detecting hidden structures. In *FOCS 2017, Berkeley, CA*, 2017. 2

[13] Vijay Bhattiprolu, Venkatesan Guruswami, and Euiwoong Lee. Sum-of-squares certificates for maxima of random tensors on the sphere. In *APPROX/RANDOM 2017*, 2017. 2, 7

[14] Vijay Bhattiprolu, Mrinalkanti Ghosh, Venkatesan Guruswami, Euiwoong Lee, and Madhur Tulsiani. Weak decoupling, polynomial folds and approximate optimization over the sphere. In *FOCS 2017,*, 2017. 2, 7

[15] N. Robertson and P. D. Seymour. Graph minors. XII. Distance on a surface. *Journal of Combinatorial Theory Series B*, 64(2):240–272, 1995. 2, 54, 78

[16] N. Robertson and P. D. Seymour. Graph minors. XIII. The disjoint paths problem. *Journal of Combinatorial Theory Series B*, 63(1):65–110, 1995. 2, 54, 78

[17] Itai Benjamini, Oded Schramm, and Asaf Shapira. Every minor-closed property of sparse graphs is testable. In *STOC 2008*, pages 393–402, 2008. 2, 3

[18] Avinatan Hassidim, Jonathan A. Kelner, Huy N. Nguyen, and Krzysztof Onak. Local graph partitions for approximation and testing. In *FOCS 2009*, 2009. 3, 4

[19] Reut Levi and Dana Ron. A quasi-polynomial time partition oracle for graphs with an excluded minor. In *ICALP*, 2013. 3, 4

[20] Alexandra Kolla and Madhur Tulsiani. Playing random and expanding unique games. 2007. 4

[21] Noga Alon, Paul D. Seymour, and Robin Thomas. A separator theorem for graphs with an excluded minor and its applications. In *STOC*, pages 293–299, 1990. 5, 63, 99

[22] Christian Wulff-Nilsen. Separator theorems for minor-free and shallow minor-free graphs with applications. In *FOCS 2011*, pages 37–46, 2011. 5

[23] Ken-ichi Kawarabayashi and Bruce A. Reed. A separator theorem in minor-closed classes. In *FOCS 2010*, pages 153–162, 2010. 5

[24] Ken-ichi Kawarabayashi, Yusuke Kobayashi, and Bruce A. Reed. The disjoint paths problem in quadratic time. *J. Comb. Theory, Ser. B*, 102(2):424–435, 2012. 6, 11, 22, 54, 61

[25] Suprovat Ghoshal, Anand Louis, and Rahul Raychaudhury. Approximation algorithms for partially colorable graphs. In *APPROX/RANDOM 2019*, pages 28:1–28:20, 2019. 6

[26] Peter Sarnak. Letter to B. Gross and J. Harris on ovals of random plane curves. pages 1–14, 2011. 7

[27] Antonio Lerario and Erik Lundberg. Statistics on Hilbert's 16th Problem. *International Mathematics Research Notices*, 2015(12):4293–4321, 05 2014. 7

[28] John Hopcroft and Robert Tarjan. Efficient planarity testing. *Journal of the ACM (JACM)*, 21(4):549–568, 1974. 9, 54, 78

[29] K. Kuratowski. Sur le problème des courbes gauches en topologie. *Fundamenta Mathematica*, 15:271–283, 1930. 9, 54, 78

[30] K. Wagner. Über eine eigenschaft der ebenen komplexe. *Mathematische Annalen*, 114:570–590, 1937. 9, 54, 78

[31] N. Robertson and P. D. Seymour. Graph minors. XX. Wagner's conjecture. *Journal of Combinatorial Theory Series B*, 92(1):325–357, 2004. 9, 10, 54, 55, 78

[32] I. Benjamini, O. Schramm, and A. Shapira. Every minor-closed property of sparse graphs is testable. In *Symposium on the Theory of Computing (STOC)*, pages 393–402, 2008. 9, 54, 56, 79, 82

[33] O. Goldreich and D. Ron. Property testing in bounded degree graphs. *Algorithmica*, 32(2):302–343, 2002. 9, 11, 54, 78

[34] Artur Czumaj, Oded Goldreich, Dana Ron, C Seshadhri, Asaf Shapira, and Christian Sohler. Finding cycles and trees in sublinear time. *Random Structures & Algorithms*, 45(2):139–184, 2014. 11, 12, 57, 83

[35] Hendrik Fichtenberger, Reut Levi, Yadu Vasudev, and Maximilian Wötzel. On testing minor-freeness in bounded degree graphs with one-sided error. *CoRR*, abs/1707.06126, 2017. 11, 12, 57, 83

[36] Reinhard Diestel. *Graph Theory, Fourth Edition*. Springer, 2010. 11, 56

[37] L. Lovász. Graph minor theory. *Bulletin of the American Mathematical Society*, 43(1):75–86, 2006. 11

[38] O. Goldreich. *Introduction to Property Testing*. Cambridge University Press, 2017. 11, 55, 56, 82, 111

[39] N. Alon, E. Fischer, I. Newman, and A. Shapira. A combinatorial characterization of the testable graph properties : it's all about regularity. *Symposium on the Theory of Computing (STOC)*, pages 251–260, 2006. 11

[40] Artur Czumaj, Asaf Shapira, and Christian Sohler. Testing hereditary properties of nonexpanding bounded-degree graphs. *SIAM Journal on Computing*, 38(6):2499–2510, 2009. 11, 82

[41] A. Hassidim, J. Kelner, H. Nguyen, and K. Onak. Local graph partitions for approximation and testing. In *Foundations of Computer Science (FOCS)*, pages 22–31, 2009. 12, 55, 56, 57, 78, 79

[42] Reut Levi and Dana Ron. A quasi-polynomial time partition oracle for graphs with an excluded minor. *ACM Transactions on Algorithms (TALG)*, 11(3):24, 2015. 12, 55, 56, 57, 79, 82

[43] Yuichi Yoshida and Hiro Ito. Testing outerplanarity of bounded degree graphs. *Algorithmica*, 73(1):1–20, 2015. 12, 55, 56, 83

[44] Alan Edelman, Avinatan Hassidim, Huy N. Nguyen, and Krzysztof Onak. An efficient partitioning oracle for bounded-treewidth graphs. pages 530–541, 2011. 12, 55, 56, 83

[45] O. Goldreich and D. Ron. A sublinear bipartite tester for bounded degree graphs. *Combinatorica*, 19(3):335–373, 1999. 12, 13, 57, 83

[46] O. Goldreich and D. Ron. On testing expansion in bounded-degree graphs. *ECCC*, TR00-020, 2000. 12

[47] Artur Czumaj and Christian Sohler. Testing expansion in bounded-degree graphs. *Combinatorics, Probability & Computing*, 19(5-6):693–709, 2010. 12

[48] S. Kale and C. Seshadhri. Testing expansion in bounded degree graphs. *Proc. 35th ICALP*, pages 527–538, 2008. 12

[49] A. Nachmias and A. Shapira. Testing the expansion of a graph. *ECCC*, TR07-118, 2007. 12

[50] Satyen Kale, Yuval Peres, and C. Seshadhri. Noise tolerance of expanders and sublinear expansion reconstruction. *SIAM J. Comput.*, 42(1):305–323, 2013. 12, 20, 44, 67

[51] Artur Czumaj, Pan Peng, and Christian Sohler. Testing cluster structure of graphs. In *Symposium on the Theory of Computing (STOC)*, pages 723–732, 2015. 12

[52] Andrei Z. Broder, Alan M. Frieze, and Eli Upfal. Static and dynamic path selection on expander graphs: A random walk approach. *Random Struct. Algorithms*, 14(1):87–109, 1999. 12

[53] Jon M. Kleinberg and Ronitt Rubinfeld. Short paths in expander graphs. In *Foundations of Computer Science (FOCS)*, pages 86–95. IEEE Computer Society, 1996. 12, 14

[54] D. Spielman and S.-H. Teng. A local clustering algorithm for massive graphs and its application to nearly-linear time graph partitioning. *SIAM Journal on Computing*, 42(1):1–26, 2012. 13, 19, 46, 58, 73, 83, 85, 102

[55] Luca Trevisan. Approximation algorithms for unique games. In *Foundations of Computer Science (FOCS)*, pages 197–205. IEEE, 2005. 13

[56] Shayan Oveis Gharan and Luca Trevisan. Approximating the expansion profile and almost optimal local graph clustering. In *Foundations of Computer Science (FOCS)*, pages 187–196. IEEE Computer Society, 2012. 13

[57] R. Andersen, F. R. K. Chung, and K. Lang. Local graph partitioning using pagerank vectors. *Foundations of Computer Science (FOCS)*, pages 475–486, 2006. 19

[58] David Aldous and James Allen Fill. Reversible markov chains and random walks on graphs, 2002. Unfinished monograph, recompiled 2014, available at `http://www.stat.berkeley.edu/~aldous/RWG/book.html`. 44

[59] D. Spielman. Lecture notes on spectral graph theory. http://www.cs.yale.edu/homes/spielman/eigs/. 46, 73, 74

[60] Ilan Newman and Christian Sohler. Every property of hyperfinite graphs is testable. *SIAM Journal on Computing*, 42(3):1095–1112, 2013. 56, 79, 80

[61] Akash Kumar, C. Seshadhri, and Andrew Stolman. Finding forbidden minors in sublinear time: a o(n$^{1/2 + o(1)}$)-query one-sided tester for minor closed properties on bounded degree graphs. *CoRR*, abs/1805.08187, 2018. 67, 68, 69, 73, 74

[62] D. P. Dubhashi and A. Panconesi. *Concentration of measure for the analysis of randomized algorithms*. Cambridge, 2009. 69, 70, 71

[63] Richard J. Lipton and Robert Endre Tarjan. Applications of a planar separator theorem. *SIAM J. Comput.*, 9(3):615–627, 1980. 78

[64] Noga Alon, Paul D. Seymour, and Robin Thomas. Planar separators. *SIAM J. Discrete Math.*, 7(2):184–193, 1994. 78

[65] Avi Wigderson. Improving the performance guarantee for approximate graph coloring. *Journal of the ACM (JACM)*, 30(4):729–735, 1983. 113

[66] Avrim Blum. New approximation algorithms for graph coloring. *J. ACM*, 41(3):470–516, 1994. 113

[67] David R. Karger, Rajeev Motwani, and Madhu Sudan. Approximate graph coloring by semidefinite programming. *J. ACM*, 45(2):246–265, 1998. 113

[68] Avrim Blum and David R. Karger. An õ(nˆ{3/14})-coloring algorithm for 3-colorable graphs. *Inf. Process. Lett.*, 61(1):49–53, 1997. 113

[69] Sanjeev Arora and Eden Chlamtac. New approximation guarantee for chromatic number. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing, Seattle, WA, USA, May 21-23, 2006*, pages 215–224, 2006. 113

[70] Eden Chlamtac. Approximation algorithms using hierarchies of semidefinite programming relaxations. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2007), October 20-23, 2007, Providence, RI, USA, Proceedings*, pages 691–701, 2007. 113

[71] Sanjeev Arora and Rong Ge. New tools for graph coloring. In *RANDOM 2011, Princeton, NJ, USA, August 17-19, 2011. Proceedings*, pages 1–12, 2011. 113, 114, 115

[72] Ken-ichi Kawarabayashi and Mikkel Thorup. Coloring 3-colorable graphs with less than $n^{1/5}$ colors. *J. ACM*, 64(1):4:1–4:23, 2017. 113

[73] Irit Dinur, Elchanan Mossel, and Oded Regev. Conditional hardness for approximate coloring. *SIAM Journal on Computing*, 39(3):843–873, 2009. 113

[74] Irit Dinur, Subhash Khot, Will Perkins, and Muli Safra. Hardness of finding independent sets in almost 3-colorable graphs. In *Foundations of Computer Science (FOCS), 2010 51st Annual IEEE Symposium on*, pages 212–221. IEEE, 2010. 113

[75] Subhash Khot and Rishi Saket. Hardness of finding independent sets in almost q-colorable graphs. In *Foundations of Computer Science (FOCS), 2012 IEEE 53rd Annual Symposium on*, pages 380–389. IEEE, 2012. 113

[76] Avrim Blum and Joel Spencer. Coloring random and semi-random k-colorable graphs. *J. Algorithms*, 19(2):204–234, 1995. 113, 114

[77] Roee David and Uriel Feige. On the effect of randomness on planted 3-coloring models. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 77–90, 2016. 113, 114, 117

[78] Terence Tao. *Structure and randomness: pages from year one of a mathematical blog*. American Mathematical Soc., 2008. 114

[79] Sanjeev Arora, Boaz Barak, and David Steurer. Subexponential algorithms for unique games and related problems. In *Foundations of Computer Science (FOCS), 2010 51st Annual IEEE Symposium on*, pages 563–572. IEEE, 2010. 114

[80] Sanjeev Arora, Boaz Barak, and David Steurer. Subexponential algorithms for unique games and related problems. *Journal of the ACM (JACM)*, 62(5):42, 2015. 114