2018 IEEE 59th Annual Symposium on Foundations of Computer Science

Finding forbidden minors in sublinear time: a $n^{1/2+o(1)}$-query one-sided tester for minor closed properties on bounded degree graphs

Akash Kumar
Department of Computer Science
Purdue University
West Lafayette, IN
Email: akumar@purdue.edu

C. Seshadhri
Department of Computer Science
University of California, Santa Cruz
Santa Cruz, CA
Email: sesh@ucsc.edu

Andrew Stolman
Department of Computer Science
University of California, Santa Cruz
Santa Cruz, CA
Email: astolman@ucsc.edu

***Abstract*—Let G be an undirected, bounded degree graph with n vertices. Fix a finite graph H, and suppose one must remove εn edges from G to make it H-minor free (for some small constant $\varepsilon > 0$). We give an $n^{1/2+o(1)}$-time randomized procedure that, with high probability, finds an H-minor in such a graph. As an application, suppose one must remove εn edges from a bounded degree graph G to make it planar. This result implies an algorithm, with the same running time, that produces a $K_{3,3}$ or K_5 minor in G. No prior sublinear time bound was known for this problem.**

By the graph minor theorem, we get an analogous result for any minor-closed property. Up to $n^{o(1)}$ factors, this resolves a conjecture of Benjamini-Schramm-Shapira (STOC 2008) on the existence of one-sided property testers for minor-closed properties. Furthermore, our algorithm is nearly optimal, by an $\Omega(\sqrt{n})$ lower bound of Czumaj et al (RSA 2014).

Prior to this work, the only graphs H for which non-trivial one-sided property testers were known for H-minor freeness are the following: H being a forest or a cycle (Czumaj et al, RSA 2014), $K_{2,k}$, $(k \times 2)$-grid, and the k-circus (Fichtenberger et al, Arxiv 2017).

***Keywords*-Planarity testing, random walks, minor-closed properties, one-sided graph property testing**

I. Introduction

Deciding if an n-vertex graph G is planar is a classic algorithmic problem solvable in linear time [1]. The Kuratowski-Wagner theorem asserts that any non-planar graph must contain a K_5 or $K_{3,3}$-minor [2], [3]. 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 [4], 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 [5] 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 [6]. 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 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 provide a certificate of rejection.

We are interested in property $\mathcal{P}$ 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-

(Supported in part by NSF CCF-1319080.)
(Supported by NSF TRIPODS grant CCF-1740850)
(Supported by NSF TRIPODS grant CCF-1740850)

2575-8454/18/$31.00 ©2018 IEEE
DOI 10.1109/FOCS.2018.00055

sided property testers as sublinear time procedures that find forbidden minors. Our main theorem follows.

Theorem I.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 $dn^{1/2+O(\delta r^4)} + d\varepsilon^{-2\exp(2/\delta)/\delta}$. 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 [4] 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 I.1 is the following.

Corollary I.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 I.1 was known are the following: $O^*(1)$ time for H being a forest, $O^*(\sqrt{n})$ for H being a cycle [7], and $O^*(n^{2/3})$ for H being $K_{2,k}$, the $(k \times 2)$-grid, and the k-circus [8]. No sublinear time bound was known for planarity.

Corollary I.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 I.1. Suppose bounded degree 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.

A. Related work

Graph minor theory is a deep topic, and we refer the reader to Chapter 12 of Diestel's book [9] and Lovász' survey [10]. For our purposes, we use as a black-box polynomial time algorithms that find fixed minors in a graph. A result of Kawarabayashi-Kobayashi-Reed provides an $O(n^2)$ time algorithm [11].

Property testing on graphs is an immensely rich area of study, and we refer the reader to Goldreich's recent textbook for more details [12]. 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 [13]. 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 [6]. Czumaj-Sohler-Shapira proved that hereditary properties of non-expanding graphs are testable [14]. 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 [15]. A result of Levi-Ron [16] 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 [17], or Edelman et al generalize for bounded treewidth graphs [18].

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 [7] made the first step by giving an $\widetilde{O}(\sqrt{n})$ one-sided tester for the property of being C_k-minor free [7]. 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 [19]. (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 [7]. 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) [8]. 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 [19] and was then (naturally) used in testing expansion properties and clustering structure [20]–[25]. 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 [26], [27]. We use techniques from Kale-Seshadhri-Peres to analyze random walks on projected Markov Chains [24]. We also employ the local partitioning methods of Spielman-Teng [28], which is in turn derived from the Lovász-Simonovits analysis technique [29].

II. Main Ideas

We give an overview of the proof strategy and discuss the various moving parts of the proof. For convenience, 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 [19]. 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 [30], [31]. Famously, the Arora-Barak-Steurer algorithm [32] 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 behavior. 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.

A. 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 $n/\mathrm{poly}(\log n)$ edges [27].) 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 $v_1, \ldots, v_5$ 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, and $\mathbf{p}_{v_i,\ell/2}$ is exactly the distribution the algorithm picks the v_i's 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, it turns out 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_is could be contracted to get a K_5-minor. We can roughly quantify how far the v_is will be by the variance of $\mathbf{p}_{v,\ell/2}$. Thus, the third and fourth problem are coupled.

B. *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 R of vertices. A careful analysis of these walks provides to 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 $s \in R$ 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.

C. *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 [28], [33], 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 [24]. 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 [29]. 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 strata, we have a lower bound on the random walks norm. This allows for local partitioning around s.

III. 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 $\texttt{FindMinor}(G, \varepsilon, H)$, 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:

- $\texttt{LocalSearch}(s)$: This procedure perform a small number of short random walks to find the piece described in §II-C. This produces a small subgraph of G, where an exact H-minor finding algorithm is used.
- $\texttt{FindPath}(u, v, k, i)$: 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.
- $\texttt{FindBiclique}(s)$: This is the main procedure mostly as described in §II-A. 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 $\texttt{FindPath}$ 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.
- $\texttt{KKR}(F, H)$: This refers to an exact H-minor finding process (in F). For concreteness, we use the quadratic time procedure of Kawarabayashi-Kobayashi-Reed [11].

```
FindMinor(G, ε, H)
  1) If ε < ε_CUTOFF, query all of G, and output
     KKR(G, H)
  2) Else
     a) Repeat ε^{-2} n^{35δr^4} times:
        i) Pick uar s ∈ V
        ii) call LocalSearch(s)
        iii) call FindBiclique(s).
```

```
LocalSearch(s)
  1) Initialize set B = ∅.
  2) For h = 1, ..., n^{7δr^4}:
     a) Perform ε^{-1} n^{30δr^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, ..., 1/δ + 4:
     a) Perform 2r^2 independent random walks of
        length 2^{i+1}ℓ 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 ∈ A, b ∈ B:
        i) Run FindPath(a, b, n^{δ(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 ℓ from u and
     v.
  2) If a walk from u and v terminate at the same vertex,
     return these paths. (Otherwise, return nothing.)
```

Theorem III.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 $dn^{1/2+O(\delta r^4)} + d\varepsilon^{-2\exp(2/\delta)/\delta}$.*

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.)

IV. Technical contributions

A. Stratification and important consequences

In this subsection and the following, we summarize one of our most important contributions - the analysis of `FindBiclique`. For a full proof, we refer the reader to the full version of the paper [34].

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

In order to analyze the behavior of `FindBiclique`, we develop the notion of returning walks and stratification.

Definition IV.2. *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.*

This definition of returning walks gives rise to the stratification process. Stratification results in a collection of disjoint sets of vertices denoted by $S_0, S_1, \ldots$ which are called *strata*. The corresponding *residue* sets denoted by $R_0, R_1, \ldots$. 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 IV.3. *Suppose R_i has been constructed. A vertex $s \in R_i$ is placed in S_i if $\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_2^2 \geq 1/n^{\delta i}$.*

The utility of stratification is perhaps best summarized in the following key lemma. Roughly speaking, the intuition is as follows. Fix some $s \in S_i$. By the definition of returning walks, 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 $\boldsymbol{q}^{(i)}_{[R_i],s}$. But if a large set of vectors is correlated with a fixed vector $\boldsymbol{q}^{(i)}_{[R_i],s}$, then we expect that many of these vectors are correlated among themselves.

Definition IV.4. *For $s \in R_i$, the distribution $\mathcal{D}_{s,i}$ has support R_i, and the probability of $u \in R_i$ is $\hat{q}^{(i+1)}_{[R_i],s}(v) = q^{(i+1)}_{[R_i],s}(v)/\|\boldsymbol{q}^{(i+1)}_{[R_i],s}\|_1$. (For convenience, we will drop the subscript i in $\mathcal{D}_{s,i}$ when it is apparent.)*

Lemma IV.5. *Fix arbitrary $s \in S_i$.*

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

We refer the reader to section 4 of [34] for the proof of this lemma.

Another important aspect of the stratification process is that it manages to stratify almost all of the vertices of the graph.

Lemma IV.6. $|R|^{-1} \sum_{s \in R} \|\boldsymbol{q}^{(i)}_{[R],s}\|_1 \geq (|R|/2n)^{2^i+1}$

Proof: We will express $\sum_{s \in R} \|\boldsymbol{q}^{(i)}_{[R],s}\|_1 = \mathbf{1}^T(\mathbb{P}_R^T M^\ell \mathbb{P}_R)^{2^i}\mathbf{1}$. First, let us prove for $i = 0$. Observe that $\sum_{s \in R} \|\boldsymbol{q}^{(0)}_{[R],s}\|_1 = \mathbf{1}_R^T M^\ell \mathbf{1}_R = ((M^T)^{\ell/2}\mathbf{1}_R)^T(M^{\ell/2}\mathbf{1}_R) = \|M^{\ell/2}\mathbf{1}_R\|_2^2$. 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 $1 \leq \lambda_1 \leq \lambda_2 \ldots \lambda_s$, with corresponding eigenvectors $\mathbf{u_1}, \mathbf{u_2}, \ldots, \mathbf{u_s}$. We can express $\mathbf{1} = \sum_{k \leq s} \alpha_k \mathbf{u_k}$, where $\sum_k \alpha_k^2 = |R|$. Writing out in the eigenbasis, $\mathbf{1}^T N \mathbf{1} = \sum_k \alpha_k^2 \lambda_k \geq |R|^2/n$. Partition the eigenvalues into the heavy and lights set as follows: $H = \{k | \lambda_k \geq |R|/2n\}$ and $L = \overline{H}$.

$$|R|^2/n \leq \sum_k \alpha_k^2 \lambda_k \leq (|R|/2n) \sum_{k \in L} \alpha_k^2 + \sum_{k \in H} \alpha_k^2$$
$$\implies \sum_{k \in H} \alpha_k^2 \geq |R|^2/2n$$

Now, we deal with general i.

$$\begin{aligned}\mathbf{1}^T N^{2^i} \mathbf{1} &= \sum_k \alpha_k^2 \lambda_k^{2^i} \\ &\geq \sum_{k \in H} \alpha_k^2 \lambda_k^{2^i} \\ &\geq (|R|/2n)^{2^i}(|R|^2/2n) = |R| \times (|R|/2n)^{2^i+1}\end{aligned}$$

■

Lemma IV.7. *Let $\varepsilon \geq \varepsilon_{\mathsf{CUTOFF}}$. Then, at most $\varepsilon n/\log n$ vertices are in $R_{1/\delta+3}$.*

Proof: Suppose for the sake of contradiction that $R_{1/\delta+3}$ has at least $\varepsilon \frac{n}{\log n}$ vertices. By Lemma IV.6,

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

Thus, the bound holds for some specific $s \in R_{1/\delta+3}$, and thus

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

Observe that $\varepsilon \geq \varepsilon_{\mathsf{CUTOFF}} \geq n^{-\delta/\exp(1/\delta)}$. For sufficiently small δ, the latter is greater than $(\log n)n^{-2\delta/(2^{1/\delta+1}+2)}$. 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 $v \in S_{1/\delta+2}$, which is a contradiction. ■

B. Analysis of `FindBiclique`

The proof of Theorem IV.1 essentially contains two parts. In the first part, summarized in the lemma below, we show that with high probability if s lies in a stratum numbered $5r^4$ and above, then `FindBiclique` succeeds in connecting each pair of vertices, $a \in A$ and $b \in B$, by a path. In the second part, we show that with some care, we can use these paths with high probability to reveal a K_{r^2,r^2} minor.

Lemma IV.8. *Suppose $s \in S_i$, for some $i \leq 2/\delta$. Then, with probability $\Omega((2n)^{-2\delta r^4})$ conditioned on the event that $A, B \subseteq R_i$, `FindBiclique`(s) finds a path between every $a \in A, b \in B$.*

Proof sketch: 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`$(a, b, n^{1/2+4\delta}, i)$ succeeds in finding a path between a and b. The probability of success for `FindBiclique`(s, k) given that $A, B \subseteq R_i$ is at least

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

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

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

Both inequalities used before are Jensen's. In [34], it is shown that for every $a\in R_i$, $\|\boldsymbol{q}^{(i)}_{[R_i],a}\|_2^2\leq 1/n^{\delta(i-1)}$ (similarly for $b\in R_i$). Also shown in [34] is that if $\boldsymbol{q}^{(i)}_{[R_i],a}\cdot\boldsymbol{q}^{(i)}_{[R_i],b}\geq 1/2n^{\delta(i+1)}$, then $\tau_{a,b}=\Omega(1)$. By Lemma IV.5, $\mathbf{E}_{a,b}[\boldsymbol{q}^{(i)}_{[R_i],a}\cdot\boldsymbol{q}^{(i)}_{[R_i],b}]\geq 1/n^{\delta(i+1)}$. We now apply Cauchy-Schwartz, and $\boldsymbol{q}^{(i)}_{[R_i],a}\cdot\boldsymbol{q}^{(i)}_{[R_i],b}\leq 1/n^{\delta(i-1)}$. Let p be the probability (over a,b) that $\boldsymbol{q}^{(i)}_{[R_i],a}\cdot\boldsymbol{q}^{(i)}_{[R_i],b}\geq 1/2n^{\delta(i+1)}$.

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

Thus, $p\geq 1/2n^{2\delta}$, and with this probability $\tau_{a,b}=\Omega(1)$. Plugging into (3), the probability of success is at least $(1/2n^{2\delta})^{r^2}$.

■

C. Local partitioning

In order to deal with the cases when `FindBiclique` will not work, we prove the following lemma.

Lemma IV.9. *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

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

For the proof of this lemma, we refer the reader to section 6 of [34]. It proceeds by analyzing a projected Markov chain using techniques from [35]. On this projected Markov chain, techniques from [29] and [28] are employed to show that we can perform local partitioning on this projected Markov chain to get low conductance cuts from the original graph.

D. Proof of Theorem III.1

We have the tools required to complete the proof of Theorem III.1. We 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 $\alpha=\varepsilon/(40r^4\log n)$.

> `Decompose`(G)
> 1) Initialize $S=V$ and $\mathcal{P}=\emptyset$.
> 2) For $i=1,\dots,5r^4$:
>   a) Assign $S':=\Big\{s\in S:\|\boldsymbol{q}^{(i+1)}_{[S],s}\|_2^2\geq 1/n^{\delta i}\Big\}$
>   b) While $|S'|\geq\alpha n$:
>   i) Choose arbitrary $s\in S''$, and let P_s be as in Lemma IV.9.
>   ii) Add P_s to $\mathcal{P}$ and assign $S:=S\setminus P_s$
>   iii) Assign $S':=\Big\{s\in S:\|\boldsymbol{q}^{(i+1)}_{[S],s}\|_2^2\geq 1/n^{\delta i}\Big\}$
>   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 output of `Decompose` is a collection of sets: those in $\mathcal{P}$, X, and S. Each set in $\mathcal{P}$ is a low conductance cut, as given by Lemma IV.9. The set X is an "excess" set, obtained by the union of all S' sets that are too small for Lemma IV.9. The set S (which is what is left of V after Step 2 finishes) is contained in strata numbered at least $5r^4$.

We will also define the *ball* around a vertex s, denoted B_s.

$$B_s=\left\{v\in V:\exists t\leq\frac{160n^{6\delta r^4}}{\alpha}\text{ s.t. }p_{s,t}(v)\geq\frac{\alpha}{n^{11\delta r^4}}\right\}$$

Lemma IV.10. *Assume $\varepsilon>\varepsilon_{\text{CUTOFF}}$. Suppose `FindMinor`$(G,\varepsilon,H)$ outputs an H-minor with probability $<2/3$. Then, the output of `Decompose` satisfies the following conditions.*

- $|X|\leq\varepsilon n/10$.
- $|S|\leq\varepsilon n/10$.
- *For all $P_s\in\mathcal{P}$, $P_s\subseteq B_s$*
- *There are at most $\varepsilon n/10$ edges that go between different P_s sets.*

Proof: Except for the bound on S, all other conditions follow from Lemma IV.9 and the parameter settings.

First, consider the X_i's formed by `Decompose`. Each of these has size at most αn, 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 IV.9. Consider the number of edges that go between P_s and the rest of S, when P_s was constructed (in `Decompose`). By Lemma IV.9 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). Observe that any edge that goes between different P_s sets must be such a cut edge, proving the fourth condition.

The second condition on $|S|$ is where we require the behavior of `FindMinor`. Suppose, for contradiction's sake, that $|S| > \varepsilon n/10$. Consider the stratification process with $R_0 = S$. By construction of S, $\forall s \in S$, $||\boldsymbol{q}^{(5r^4+1)}_{[S],s}|| \leq 1/n^{4\delta r^4}$. Thus, all of these vertices will lie in strata numbered $5r^4$ or above. Since $\varepsilon > \varepsilon_{\mathsf{CUTOFF}}$, by Lemma IV.7, at most $\varepsilon n/\log n$ vertices are in strata numbered more than $1/\delta + 3$. By Theorem IV.1, for at least $\varepsilon n/10 - \varepsilon n/\log n \geq n^{1-\delta}$ 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 $n^{-5\delta r^4}$. The probability that $n^{20\delta r^4}$ calls do not find such a minor is $< 1/6$, a contradiction. Thus, $|S| \leq \varepsilon n/10$. ∎

Claim IV.11. *Assume $\varepsilon > \varepsilon_{\mathsf{CUTOFF}}$. Suppose* `FindMinor`$(G, \varepsilon, H)$ *outputs an H-minor with probability $< 2/3$. Then, for at most $n^{1-30\delta r^4}$ vertices s, B_s induces an H-minor.*

Proof: Suppose not. Observe that `LocalSearch` is called on at least $n^{35\delta r^4}$ vertices, so with probability at least $1 - (1 - n^{-30\delta r^4})^{n^{35\delta r^4}} > 5/6$, the algorithm samples vertex s where B_s induces an H-minor. There are at most $160\alpha^{-1}n^{6\delta r^4} \cdot \alpha^{-1}n^{11\delta r^4} \leq n^{18\delta r^4}$ vertices in B_s. The probability of ending at such a vertex v (from s) is at least $\alpha n^{-11\delta r^4} \geq n^{-12\delta r^4}$. Since `LocalSearch`$(s)$ performs $n^{30\delta r^4}$ random walks from s, the probability of not adding v to B is at most $\exp(-n^{15\delta r^4})$. Taking a union bound over all of B_s, with probability at least $5/6$, the set B discovered by `LocalSearch`(s) is a superset of B_s. In this situation, `LocalSearch`(s) finds an H-minor. Thus, the overall probability of outputting an H-minor is at least $(5/6)^2 > 2/3$, a contradiction. ∎

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

Claim IV.12. *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 IV.10 to partition V into $\mathcal{P}, X, S$ as given.

Call $P_s \in \mathcal{P}$ bad, if P_s induces an H-minor. Similarly, call B_s bad if it induces an H-minor. Observe that for all $P_s \in \mathcal{P}$, $P_s \subseteq B_s$. Thus, the union of bad P_s sets is contained in the union on bad B_s sets. The size of any B_s is at most $160\alpha^{-1}n^{6\delta r^4} \times \alpha^{-1}n^{11\delta r^4} \leq n^{18\delta r^4}$. There are at most $n^{1-30\delta r^4}$ bad B_s sets, by Claim IV.11. So, the total union is at most $n^{1-12\delta r^4} \leq \varepsilon n/10$.

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 IV.10 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 IV.13. *The running time of* `FindMinor`(G, ε, H) *is*

$$dn^{1/2+O(\delta r^4)} + d\varepsilon^{-2\exp(2/\delta)/\delta}.$$

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 performed by a single call to `FindBiclique` is

$$\sum_{i=5r^4}^{1/\delta+3}\left(2r^2 2^{i+1} n^{5\delta} + 2r^2 n^{\delta i/2+9\delta} 2^i n^{5\delta}\right)$$
$$= r^2 n^{1/2+O(\delta)}.$$

While this is the total number of vertices encountered, we note that the calls made to $\texttt{KKR}(F, H)$ 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 $\texttt{KKR}(F, H)$ 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 IV.6.

REFERENCES

[1] J. Hopcroft and R. Tarjan, "Efficient planarity testing," *Journal of the ACM (JACM)*, vol. 21, no. 4, pp. 549–568, 1974. 1

[2] K. Kuratowski, "Sur le problème des courbes gauches en topologie," *Fundamenta Mathematica*, vol. 15, pp. 271–283, 1930. 1

[3] K. Wagner, "Über eine eigenschaft der ebenen komplexe," *Mathematische Annalen*, vol. 114, pp. 570–590, 1937. 1

[4] N. Robertson and P. D. Seymour, "Graph minors. XX. Wagner's conjecture," *Journal of Combinatorial Theory Series B*, vol. 92, no. 1, pp. 325–357, 2004. 1, 2

[5] I. Benjamini, O. Schramm, and A. Shapira, "Every minor-closed property of sparse graphs is testable," in *Symposium on the Theory of Computing (STOC)*, 2008, pp. 393–402. 1

[6] O. Goldreich and D. Ron, "Property testing in bounded degree graphs," *Algorithmica*, vol. 32, no. 2, pp. 302–343, 2002. 1, 2

[7] A. Czumaj, O. Goldreich, D. Ron, C. Seshadhri, A. Shapira, and C. Sohler, "Finding cycles and trees in sublinear time," *Random Structures & Algorithms*, vol. 45, no. 2, pp. 139–184, 2014. 2

[8] H. Fichtenberger, R. Levi, Y. Vasudev, and M. Wötzel, "On testing minor-freeness in bounded degree graphs with one-sided error," *CoRR*, vol. abs/1707.06126, 2017. 2

[9] R. Diestel, *Graph Theory, Fourth Edition*. Springer, 2010. 2

[10] L. Lovász, "Graph minor theory," *Bulletin of the American Mathematical Society*, vol. 43, no. 1, pp. 75–86, 2006. 2

[11] K. Kawarabayashi, Y. Kobayashi, and B. A. Reed, "The disjoint paths problem in quadratic time," *J. Comb. Theory, Ser. B*, vol. 102, no. 2, pp. 424–435, 2012. 2, 6

[12] O. Goldreich, *Introduction to Property Testing*. Cambridge University Press, 2017. 2

[13] 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)*, pp. 251–260, 2006. 2

[14] A. Czumaj, A. Shapira, and C. Sohler, "Testing hereditary properties of nonexpanding bounded-degree graphs," *SIAM Journal on Computing*, vol. 38, no. 6, pp. 2499–2510, 2009. 2

[15] A. Hassidim, J. Kelner, H. Nguyen, and K. Onak, "Local graph partitions for approximation and testing," in *Foundations of Computer Science (FOCS)*, 2009, pp. 22–31. 2

[16] R. Levi and D. Ron, "A quasi-polynomial time partition oracle for graphs with an excluded minor," *ACM Transactions on Algorithms (TALG)*, vol. 11, no. 3, p. 24, 2015. 2

[17] Y. Yoshida and H. Ito, "Testing outerplanarity of bounded degree graphs," *Algorithmica*, vol. 73, no. 1, pp. 1–20, 2015. 2

[18] A. Edelman, A. Hassidim, H. N. Nguyen, and K. Onak, "An efficient partitioning oracle for bounded-treewidth graphs," in *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques RANDOM 2011, Princeton, NJ, USA, August 17-19, 2011. Proceedings*, 2011, pp. 530–541. 2

[19] O. Goldreich and D. Ron, "A sublinear bipartite tester for bounded degree graphs," *Combinatorica*, vol. 19, no. 3, pp. 335–373, 1999. 2, 3

[20] ——, "On testing expansion in bounded-degree graphs," *ECCC*, vol. TR00-020, 2000. 3

[21] A. Czumaj and C. Sohler, "Testing expansion in bounded-degree graphs," *Combinatorics, Probability & Computing*, vol. 19, no. 5-6, pp. 693–709, 2010. 3

[22] S. Kale and C. Seshadhri, "Testing expansion in bounded degree graphs," *Proc. 35th ICALP*, pp. 527–538, 2008. 3

[23] A. Nachmias and A. Shapira, "Testing the expansion of a graph," *ECCC*, vol. TR07-118, 2007. 3

[24] S. Kale, Y. Peres, and C. Seshadhri, "Noise tolerance of expanders and sublinear expansion reconstruction," *SIAM J. Comput.*, vol. 42, no. 1, pp. 305–323, 2013. 3, 6

[25] A. Czumaj, P. Peng, and C. Sohler, "Testing cluster structure of graphs," in *Symposium on the Theory of Computing (STOC)*, 2015, pp. 723–732. [Online]. Available: http://doi.acm.org/10.1145/2746539.2746618 3

[26] A. Z. Broder, A. M. Frieze, and E. Upfal, "Static and dynamic path selection on expander graphs: A random walk approach," *Random Struct. Algorithms*, vol. 14, no. 1, pp. 87–109, 1999. 3

[27] J. M. Kleinberg and R. Rubinfeld, "Short paths in expander graphs," in *Foundations of Computer Science (FOCS)*. IEEE Computer Society, 1996, pp. 86–95. 3

[28] 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*, vol. 42, no. 1, pp. 1–26, 2012. 3, 5, 9

[29] L. Lovász and M. Simonovits, "The mixing rate of markov chains, an isoperimetric inequality, and computing the volume," in *Foundations of Computer Science (FOCS)*, 1990, pp. 346–354. 3, 6, 9

[30] L. Trevisan, "Approximation algorithms for unique games," in *Foundations of Computer Science (FOCS)*. IEEE, 2005, pp. 197–205. 3

[31] S. O. Gharan and L. Trevisan, "Approximating the expansion profile and almost optimal local graph clustering," in *Foundations of Computer Science (FOCS)*. IEEE Computer Society, 2012, pp. 187–196. 3

[32] S. Arora, B. Barak, and D. Steurer, "Subexponential algorithms for unique games and related problems," *Journal of the ACM*, vol. 62, no. 5, pp. 42:1–42:25, 2015. 3

[33] R. Andersen, F. R. K. Chung, and K. Lang, "Local graph partitioning using pagerank vectors," *Foundations of Computer Science (FOCS)*, pp. 475–486, 2006. 5

[34] A. Kumar, C. Seshadhri, and A. Stolman, "Fiding forbidden minors in sublinear time: a $O(n^{1/2+o(1)})$-query one-sided tester for minor closed properties on bounded degree graphs," *ArXiv e-prints*, May 2018. 7, 8, 9

[35] S. Kale and C. Seshadhri, "An expansion tester for bounded degree graphs," *SIAM J. Comput.*, vol. 40, no. 3, pp. 709–720, 2011. 9