Exact Recovery Algorithm for Planted Bipartite Graph in Semi-Random Graphs

Akash Kumar ✉
School of Computer and Communication Sciences, EPFL, Lausanne, Switzerland

Anand Louis ✉
Computer Science and Automation Department, IISc, Bangalore, India

Rameesh Paul ✉
Computer Science and Automation Department, IISc, Bangalore, India

Abstract

The problem of finding the largest induced balanced bipartite subgraph in a given graph is NP-hard. This problem is closely related to the problem of finding the smallest Odd Cycle Transversal.

In this work, we consider the following model of instances: starting with a set of vertices V, a set $S \subseteq V$ of k vertices is chosen and an arbitrary d-regular bipartite graph is added on it; edges between pairs of vertices in $S \times (V \setminus S)$ and $(V \setminus S) \times (V \setminus S)$ are added with probability p. Since for $d = 0$, the problem reduces to recovering a planted independent set, we don't expect efficient algorithms for $k = o\left(\sqrt{n}\right)$. This problem is a generalization of the planted balanced biclique problem where the bipartite graph induced on S is a complete bipartite graph; [46] gave an algorithm for recovering S in this problem when $k = \Omega\left(\sqrt{n}\right)$.

Our main result is an efficient algorithm that recovers (w.h.p.) the planted bipartite graph when $k = \Omega_p\left(\sqrt{n \log n}\right)$ for a large range of parameters. Our results also hold for a natural semi-random model of instances, which involve the presence of a monotone adversary. Our proof shows that a natural SDP relaxation for the problem is integral by constructing an appropriate solution to it's dual formulation. Our main technical contribution is a new approach for construction the dual solution where we calibrate the eigenvectors of the adjacency matrix to be the eigenvectors of the dual matrix. We believe that this approach may have applications to other recovery problems in semi-random models as well.

When $k = \Omega\left(\sqrt{n}\right)$, we give an algorithm for recovering S whose running time is exponential in the number of small eigenvalues in graph induced on S; this algorithm is based on subspace enumeration techniques due to the works of [42, 8, 41].

2012 ACM Subject Classification Theory of computation → Semidefinite programming; Theory of computation → Graph algorithms analysis; Theory of computation → Approximation algorithms analysis

Keywords and phrases SDP duality, Planted models, Semi-random models, Exact recovery, Threshold rank, Spectral embedding, Subspace enumeration

Digital Object Identifier 10.4230/LIPIcs.ICALP.2022.84

Category Track A: Algorithms, Complexity and Games

Related Version *Full Version*: https://arxiv.org/abs/2205.03727

Funding *Akash Kumar*: Received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 759471).
Anand Louis: Supported in part by SERB Award ECR/2017/003296 and a Pratiksha Trust Young Investigator Award.

1 Introduction

Given a graph $G = (V, E)$, the problem of finding the largest induced bipartite subgraph of G is well known to be NP-hard [64]. The problem is equivalent to the Odd Cycle Transversal problem. The problem is also related to the balanced biclique problem, where the task is

49th International Colloquium on Automata, Languages, and Programming (ICALP 2022).
Editors: Mikołaj Bojańczyk, Emanuela Merelli, and David P. Woodruff;
Article No. 84; pp. 84:1–84:20
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

that of finding the largest induced balanced complete bipartite subgraph. This problem has a lot of practical application in computational biology [19], bioinformatics [65] and VLSI design [7].

For the worst-case instance of the problem, the work [3] gives an algorithm that computes a set with at least $\left(1-\mathcal{O}\left(\varepsilon\sqrt{\log n}\right)\right)$ fraction of vertices which induces a bipartite graph, when it is promised that the graph contains an induced bipartite graph having $(1-\varepsilon)\,n$ fraction of the vertices. The work [32] gives an efficient randomized algorithm that computes an induced bipartite subgraph having $\left(1-\mathcal{O}\left(\sqrt{\varepsilon \log d}\right)\right)$ fraction of the vertices where d is the bound on the maximum degree of the graph. They also give a matching (up to constant factors) Unique Games hardness for certain regimes of parameters. We refer to Section 1.2 for more details about these related problems.

In an effort to better understand the complexity of various computationally intractable problems, a lot of work has been focused on the special cases of the problem, and towards studying the problem in various *random* and *semi-random* models. Here, one starts with solving the problem for random instances (for graph problems this is often $G_{n,p}$ Erdős-Rényi graphs[1]). The analysis in random instances is often much simpler, and one can give algorithms with "good" approximation guarantees. The next goal in this direction is to plant a solution that is "clearly optimal" in an ambient random graph and then attempt to recover this planted solution. We, therefore, build towards the worst-case instances of the problem by progressively weakening our assumptions. We refer to the book [59] for a more detailed discussion of these models in the context of other problems like planted clique, planted bisection, k-coloring, Stochastic Block Models, and Matrix completion problems.

We start our discussion with the problem of computing a maximum clique/independent set, since it has been extensively studied in such planted models. In the planted clique/independent set problem we plant a clique/independent set of size k in an otherwise random $G_{n,p}$ graph. The work [6] presents an algorithm, which, given a graph $G \sim \mathcal{G}(n, 1/2)$ with a planted clique/independent set of size k, recovers the planted clique when $k > c_1\sqrt{n}$ (where c_1 is a constant). We will refer to the planted independent set/clique problem at various points throughout the introduction

Such *random planted models* have been studied in context of other problems as well such as the planted 3-coloring problem [14, 5], planted dense subgraph problem [34, 35, 36], planted bisection and planted Stochastic Block models [16, 22, 37, 17, 21, 2], to state a few. We define a similar *random planted model* to study our problem, as stated below.

▶ **Definition 1** (Random planted model). *Given n, k, d, p, our planted bipartite graph is constructed as follows,*

1. *Let V be a set of n vertices. Fix an arbitrary subset $S \subset V$ such that $|S| = k$.*
2. *Add edges arbitrarily inside S such that the resulting graph is a connected d-regular bipartite graph. Let S_1, S_2 denote the bipartite components.*
3. *For each pair of vertices in $S \times (V \setminus S)$, add an edge independently with probability p.*
4. *For each pair of vertices in $(V \setminus S) \times (V \setminus S)$, add an edge independently with probability p.*

For planted cliques, a lot of work has been done in the special case of $p = 1/2$. However, people have studied other problems such as the planted bisection problems [24], and exact recovery problems in Stochastic Block Models [2] in the harder $p = o(1)$ regimes. Therefore, we also aim to solve our problem in $p = o(1)$ regimes.

[1] For each pair of vertices, an edge is added independently with probability p.

We note that this problem is a generalization of the planted independent set and the planted balanced biclique problem. For $d = 0$, it reduces to recovering a planted independent set and hence we do not expect efficient algorithms for $k = o\left(\sqrt{n}\right)$ [28, 11]. For $k = \Omega\left(\sqrt{n}\right)$, both these special cases i.e the planted independent set problem [6, 26], and the planted balanced biclique problem [46] admit a polynomial-time recovery algorithm. So it is natural to consider $k = \Omega\left(\sqrt{n}\right)$ as a benchmark for recovery and look for algorithms in this regime. The other consideration for interesting regimes to study the problem comes by viewing this problem as a special case of the densest k-subgraph (DkS) problem. When $d \gg pk$, the problem can be viewed as the densest k-subgraph (DkS) and for $d \ll pk$, the problem can be viewed as sparsest k-subgraph problem (studying the complement of this graph would be an instance of DkS problem). However, this general DkS problem is information-theoretically unsolvable for $d = pk$ [18]. Formally, this follows from Theorem 2.1 in the work [18], by setting $d = qk = pk$ and setting $r = 1$ where q is the edge probability within the vertices of planted subgraph and a p is the edge probability when at least one of the vertex does not belong to the planted subgraph and r is the number of clusters. Therefore we focus our attention to the case when $d \approx pk$ (also including $d = pk$). In our problem, we can hope to use the specifics of the bipartite structure in hand and recover the planted set exactly.

1.1 Our models and results

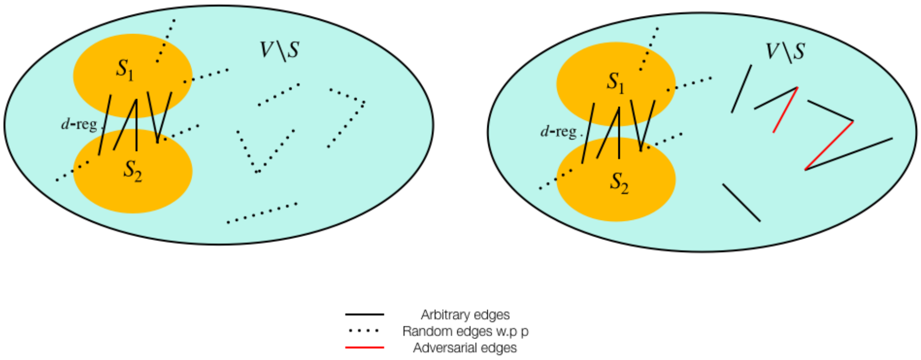

Figure 1 Random Planted model Definition 1 (left) and semi-random model Definition 2 (right).

We start by introducing our semi-random model which attempts to robustify the random planted model from Definition 1.

▶ **Definition 2** (Semi-random model). *Fix n, k, d, p, we now describe how a graph G from our semi-random model is generated,*

1. *Let V be a set of n vertices. Fix an arbitrary subset $S \subset V$ such that $|S| = k$.*
2. *Add edges arbitrarily inside S such that the resulting graph is a connected d-regular bipartite graph. Let S_1, S_2 denote the bipartite components.*
3. *For each pair of vertices in $S \times (V \setminus S)$, add an edge independently with probability p.*
4. *Arbitrarily add edges in $(V \setminus S) \times (V \setminus S)$ such that smallest eigenvalue of the matrix $\left(A_{(V\setminus S)\times(V\setminus S)} - p\mathbb{1}_{V\setminus S}\mathbb{1}_{V\setminus S}^T\right)$ is greater than $-\left((1/2-\alpha)/(1/2+\alpha)\right)d$ where α is a small[2] positive constant (throughout this paper we assume $\alpha \leq 1/6$).*
5. *Allow a monotone adversary to add edges in $(V \setminus S) \times (V \setminus S)$ arbitrarily.*

[2] Note that the smaller the value of α, the weaker is this assumption.

▶ **Observation 3.** *Definition 2 also captures Definition 1; since in the case when $V \backslash S$ is chosen to be a $G_{(n-k),p}$ random graph, $\left(A_{(V\backslash S)\times(V\times S)} - p\mathbb{1}_{V\backslash S}\mathbb{1}_{V\backslash S}^T\right) = A_{(V\backslash S)\times(V\backslash S)} - \mathbb{E}\left[A_{(V\backslash S)\times(V\backslash S)}\right]$, and therefore the smallest eigenvalue of $\left(A_{(V\backslash S)\times(V\times S)} - p\mathbb{1}_{V\backslash S}\mathbb{1}_{V\backslash S}^T\right)$ is greater than $-2\sqrt{n}$ (as follows from the work [63]).*

Models stronger than *random planted models* have also been considered in the literature for planted problems. The work [26] studies the planted clique problem in what they call the "sandwich model". The model is constructed as per the random planted model in Definition 1, but an adversary is allowed to act on the top of that in a fashion similar to step 5 of Definition 2.

The work [24] introduced a strong adversarial *semi-random* model (referred to as the *Fiege and Kilian* model). They gave recovery algorithms for the planted clique ($k = \Omega(n)$ regimes) and for the planted bisection and planted k-coloring in this model. The work [54] further shows that one can recover the planted clique for $k = \Omega_p\left(n^{2/3}\right)$ [3] in [24] model.

In the Feige-Kilian model, step 4 allows for any arbitrary graph in $(V \setminus S) \times (V \setminus S)$. However, with no further assumptions on graph induced on $V \setminus S$, even for the special case of planted independent set problem ($d = 0$), the best known algorithm [54] works only for $k = \Omega_p\left(n^{2/3}\right)$. However, since our benchmark is $k = \Omega\left(\sqrt{n}\right)$, we look at a model with stronger assumptions than the Feige-Kilian model. In order to uniquely identify the planted graph, we need to assume that $V \setminus S$ is far from having any induced bipartite subgraphs of degree at least d. Our condition in step 4 implies that this indeed holds. This is because if the smallest eigenvalue is greater than $-d/2 + 2\sqrt{n}$, the graph is indeed far from having an induced bipartite subgraph of smallest degree d. Since otherwise, a vector having entries 1 for one side of the bipartition and -1 on the other side and 0 elsewhere achieves a Rayleigh Quotient of value $-d$ (and hence the smallest eigenvalue is at most $-d$).

We now present our main result which holds for both the random planted model (Definition 1) and semi-random model (Definition 2).

▶ **Theorem 4.** *For n, k, d, p satisfying $k = \Omega_p\left(\sqrt{n \log n}\right)$ and $p = \Omega\left(\log k/k\right)^{1/6}$ and $d \geq 2pk/3$, there exists a deterministic algorithm that takes as input an instance generated by Definition 2, and recovers the arbitrary planted set S exactly, in polynomial time and with high probability (over the randomness of the input).*

Achieving exact recovery for $k = \Omega_p\left(\sqrt{n}\right)$ is still an open problem. To the best of our knowledge, nothing is known about this problem in full generality. For the planted clique problem, recovery for $k = \Omega\left(\sqrt{n \log n}\right)$ is trivial [43]. However, such techniques don't work for our problem when $d = pk$. We prove Theorem 4 by showing that an SDP relaxation for the problem is integral, by constructing an optimal dual solution. We give an outline of the proof in Section 1.4. We leave the proof of a formal version of this theorem to the full version of the paper.

Our proofs use the spectral properties of bipartite graphs and random graphs to show the existence of an optimal dual solution having large rank. Our main technical contribution is a new approach for constructing a dual solution where we *calibrate the eigenvectors* of the adjacency matrix to be the eigenvectors of the dual matrix. We believe that this approach may have applications to other recovery problems in semi-random models as well.

[3] Ω_p hides $\mathsf{poly}\left(1/p\right)$ factors.

▶ **Theorem 5.** *For n, k, d, p, satisfying $k = \Omega_p(\sqrt{n})$, there exists a deterministic algorithm that takes as input an instance generated as per Definition 1, and recovers the arbitrary planted set S exactly with high probability (over the randomness of the input) in time exponential in the number of small eigenvalues of the adjacency matrix (eigenvalues smaller than $-d/2 + 2\sqrt{n}$) of the graph induced on S.*

We leave the proof of a formal version of this theorem to the full version of the paper.

▶ **Observation 6.** *For and many special classes of instances such as, (i) when the probability $p = \Omega(1)$, (ii) when the planted graph is a complete bipartite graph like in the balanced biclique problem (iii) when the planted bipartite graph is a d-regular random graph or (iv) more generally when the planted graph is a d-regular expander graph; the number of these small eigenvalues is a constant in the regimes of $d = \Omega(pk)$ and Theorem 5 allows efficient recovery (running time of the algorithm is polynomial in n).*

1.2 Related Work

Odd Cycle Transversal problem

The odd cycle transversal problem asks to find the smallest set of vertices in the graph such that the set has an intersection with every odd cycle of the graph. Removing these vertices will result in a bipartite graph, and hence this problem is equivalent to finding the largest induced bipartite graph. Owing to the hereditary nature of the bipartiteness property, the problem is NP-hard, as follows from the work of Yannakakis [64]. The work [64] shows that for a broad class of problems that have a structure that is hereditary on induced subgraphs, finding such a structure is NP-Complete. The optimal long code test by Khot and Bansal [10] rules out any constant factor approximation for this problem. On the algorithmic front, casting the problem as a 2-CNF deletion problem, [4] gives a reduction to the min-multicut problem. This reduction gives us an $\mathcal{O}(\log n)$ approximation due to the work [30], which was further improved to $\mathcal{O}\left(\sqrt{\log n}\right)$ in the work [3]. The work [32] gives an efficient randomized algorithm that removes only $\mathcal{O}\left(n\sqrt{\mathsf{OPT}\log d}\right)$ vertices where d is the bound on the maximum degree of the graph and OPT denotes the fraction of vertices in the optimal set. They also give a matching (up to constant factors) Unique Games hardness for certain regimes of parameters.

The problem is equivalent to finding the largest 2-colorable subgraph of a given graph and is known as the partial 2-coloring problem. The work [33] studies the problem in the Feige-Kilian semi-random model [24], where a 2-colorable graph of size $(1-\varepsilon)n$ is planted. They give an algorithm that outputs a set $\mathcal{S}'$ such that $|\mathcal{S}'| \geq \left(1 - \varepsilon c/p^2\right) n$ for $p = \Omega\left(\sqrt{\log n/n}\right)$ and $\varepsilon \leq p^2$ where c is a positive constant. Their algorithm is a partial recovery algorithm and works for the regimes when ε is small. Our results in Theorem 4 hold when $1-\varepsilon$ is small and give complete recovery for a large range of p. However, since our model in Definition 2 makes stronger assumptions than the [24] model, we don't make any comparisons.

Balanced Biclique problem

In the balanced complete bipartite subgraph problem (also called the balanced biclique problem), we are given a graph on n vertices and a parameter k, and the problem then asks whether there is a complete bipartite subgraph that is balanced with k vertices in each of the bipartite components. The problem was studied when the underlying graph is a bipartite graph, and shown to be NP-complete by a reduction from the CLIQUE problem in the works [29, 38]. They additionally note that the balanced constraint is what makes the problem hard. If we remove the balanced constraint, the problem can be reduced to finding a maximum independent set in a bipartite graph. The latter problem admits a polynomial-time solution using the matching algorithm. The work [25] shows that this problem of finding a maximum balanced biclique is hard to approximate within a factor of $2^{(\log n)^{\delta}}$ for some $\delta > 0$, under the assumption that $\mathsf{3SAT} \notin \mathsf{DTIME}\left(2^{n^{3/4+\varepsilon}}\right)$ for some $\varepsilon > 0$. Recently, the work [53] showed that one cannot find a better approximation than $n^{1-\varepsilon}$, assuming the *Small Set Expansion Hypothesis* and that $\mathsf{NP} \nsubseteq \mathsf{BPP}$ for every constant $\varepsilon > 0$.

A related problem is the maximum edge biclique problem, where we are asked to find whether G contains a biclique with at least k edges. This problem was also shown to be NP-hard in the work [58].

Given these intractability results for general graphs, there has been some success in special classes of graphs. In graphs with constant arboricity, the work [23] gives a linear time algorithm that lists all maximal complete bipartite subgraphs. In a degree bounded graph, the work [60] gives a combinatorial algorithm for the balanced biclique problem that runs in time $\mathcal{O}\left(n2^{d}\right)$. Another systematic approach, however, is to consider planted and semi-random models for the problem. In the work [46], they study the planted version of the problem, which, they call "hidden biclique problem". Their model is similar to our model in Definition 1; however, we consider an arbitrary d-regular bipartite graph instead of a complete bipartite graph. They give a linear-time combinatorial algorithm that finds the planted hidden biclique with high probability (over the randomness of the input instance) for $k = \Omega\left(\sqrt{n}\right)$. Their algorithm builds on the "Low Degree Removal" algorithm, due to Feige and Ron [27] which finds a planted clique in linear time.

Graph problems in Semi-random and Pseudorandom models

A wide variety of random graph models and their relaxations have been a rich source of algorithmic problems on graphs. Alon and Kahale [5] sharpened the results of Blum and Spencer [14] and gave algorithms that recover a planted 3-coloring in a natural family of random 3-colorable instances. [44] extended this result and showed how to recover a 3-coloring when the input graph is pseudorandom (has some mild expansion properties) and is known to admit a random like 3-coloring. A unified spectral approach by McSherry [55] gives a single shot recovery algorithm for many problems in these random planted models. One can use the [55] framework to recover a planted random bipartite graph; however, it is not known if it will work if S is an arbitrary bipartite graph.

On the other side, we have semi-random models. Notably, the Feige-Kilian model [24] is one of the strongest semi-random models. In [24], they also give recovery algorithms for planted clique, planted k-colorable, and planted bisection problem in this model. In [54], they give a recovery algorithm for the independent set problem for large regimes of parameters. The work [40] generalizes these results to r-uniform hypergraphs in this model. There are other works [51, 52, 49, 50] that study graph partitioning in semi-random models.

A host of work has been done in various random and semi-random models for the more general densest k-subgraph problem. The works by Hajek, Wu, and Xu [34, 35, 36] study the problem when the planted dense subgraph is random and gives algorithms for exact recovery using SDP relaxations for some range of parameters. They complement these results by providing information-theoretic limits for regimes where recovery is impossible. The work by [13] studies this problem when the planted graph is arbitrary. They analyze an SDP-based method to distinguish the dense graphs from the family of $G_{n,p}$ graphs when $k \geq \sqrt{n}$. The work [39] studies the problem of densest k-subgraph in some semi-random model and gives a partial recovery algorithm for some regimes of d, k, n, p.

SDP has been the tool of choice for exact recovery in semi-random models. Starting from the fundamental works of exact recovery for the planted clique problem [26], for the planted bisection problem [24], for Stochastic Block Models [2] etc., (and many other works as have been mentioned above), are based on SDP relaxations. A natural way to analyze these SDP relaxations is by constructing an optimal dual solution to prove integrality of the primal relaxation. This idea has been explored in the works of [24, 20, 13, 1, 2, 49], to state a few. We note that the task of constructing an optimal dual solution is problem-specific, and there is no generic way of doing this.

1.3 Preliminaries

We start with some essential notation to understand the proof overview and review some well-known facts about random perturbation matrices. Then, we write our SDP relaxation to the problem and the accompanying dual SDP. We follow this up with a discussion on some well known tools from spectral graph theory such as the *threshold rank* and *spectral embedding*. We will build on these ideas in our Proof Overview Section 1.4 to show that the primal SDP is an optimal one and the primal matrix is a rank-one matrix.

1.3.1 Notation

We let $[M]_{n\times n}$ denote a matrix M of size $n \times n$. For some set of indices $R_1, R_2 \subseteq [n]$, $M_{R_1\times R_2}$ denotes a matrix of size $n \times n$ constructed out of matrix M of size $n \times n$ by copying the entries for $(i, j) \in R_1 \times R_2$ and setting rest of the entries to be 0. We let $M|_{R_1\times R_2}$ denote the matrix of size $|R_1| \times |R_2|$ constructed from a matrix M of size $n \times n$ by taking rows corresponding to R_1 and columns corresponding to R_2. The eigenvalues of a matrix M are sorted as $\lambda_1(M) \leq \lambda_2(M) \leq \ldots \leq \lambda_n(M)$. We will drop the matrix M wherever it is clear from the context. The eigenvectors are also sorted by their corresponding eigenvalues.

1.3.2 Spectral bounds on Perturbation matrices

We let A denote the adjacency matrix of the graph obtained using Definition 1. We can express the matrix A as sum of "simpler" matrices,

$$A = A_{S\times S} + A_{V\setminus S\times V\setminus S} + p\left(\mathbb{1}\mathbb{1}^T - \mathbb{1}_S\mathbb{1}_S^T - \mathbb{1}_{V\setminus S}\mathbb{1}_{V\setminus S}^T\right) + R \quad \left(R_{ij} \stackrel{\text{def}}{=} A_{ij} - \mathbb{E}[A_{ij}]\right) \tag{1}$$

where $A_{S\times S}$ represents the matrix corresponding to the planted bipartite graph, the term $p\left(\mathbb{1}\mathbb{1}^T - \mathbb{1}_S\mathbb{1}_S^T - \mathbb{1}_{V\setminus S}\mathbb{1}_{V\setminus S}^T\right)$ is the expected adjacency matrix for the random graph and R as defined above is the perturbation matrix corresponding to the random part of the graph.

▶ **Proposition 7.** *For the perturbation matrix R as defined in equation (1) we have that $\|R\| \leq 2\sqrt{n}$ almost surely.*

Proof. R is a symmetric random matrix and the entries R_{ij} can be treated as random variables, bounded between -1 and 1, with expectation 0 and variance $p(1-p) \leq 1/4$. Also the entries R_{ij} are independent and hence, by Theorem 1.1 in the work [63], we have $\|R\| \leq 2\sqrt{n}$ almost surely. ◀

1.3.3 SDP Relaxation

Our main results are based on analyzing the following SDP relaxation SDP 8. We construct its dual SDP 9.

▶ **SDP 8** (Primal).

$$\min \sum_{\{i,j\}\in E} 2\left\langle \mathbf{x}_i, \mathbf{x}_j \right\rangle$$

subject to

$$\sum_{i\in V} \left\|\mathbf{x}_i\right\|^2 = 1 \tag{2}$$

$$\left\|\mathbf{x}_i\right\|^2 \leq 1/k \qquad \forall i \in V \tag{3}$$

$$\left\langle \mathbf{x}_i, \mathbf{x}_j \right\rangle \leq 0 \qquad \forall \{i,j\} \in E\,. \tag{4}$$

▶ **SDP 9** (Dual).

$$\max\ \beta - \sum_{i\in V} \gamma_i$$

subject to

$$Y = A - \beta I + k \sum_{i\in V} \gamma_i D_i + \sum_{\{i,j\}\in E} B_{ij}\left(\mathbb{1}_{ij} + \mathbb{1}_{ji}\right) \tag{5}$$

$$B_{ij} \geq 0, \qquad \forall \{i,j\} \in E \tag{6}$$

$$Y \succeq 0\,. \tag{7}$$

In SDP 9, the Lagrange multipliers β_i's,γ_i's and B_{ij}'s are our dual variables and Y is the dual SDP matrix. By $\mathbb{1}_{ij}$ we mean an indicator matrix which is one for (i,j) entry and zero elsewhere. Similarly, D_i is an indicator matrix which is one for (i,i) entry and zero elsewhere. For clarity, we will denote $\sum_{\{i,j\}\in E} B_{ij}\left(\mathbb{1}_{ij} + \mathbb{1}_{ji}\right)$ by a matrix B.

Intended solution

We denote the primal SDP matrix by X and let $\mathbf{x}_i$ denote the vector corresponding to vertex i such that $X_{ij} = \langle \mathbf{x}_i, \mathbf{x}_j \rangle$.

Our intended integral solution to the SDP is $X = \mathbf{g}\mathbf{g}^T$, where $\mathbf{g} \in \mathbb{R}^n$ s.t $g_i = 1/\sqrt{k}$ for $i \in S_1$, $g_i = -1/\sqrt{k}$ for $i \in S_2$ and 0 otherwise. This solution is obtained by setting,

$$\mathbf{x}_i^* = \begin{cases} \hat{e}/\sqrt{k} & \text{if } i \in S_1 \\ -\hat{e}/\sqrt{k} & \text{if } i \in S_2 \\ 0 & \text{otherwise,} \end{cases} \tag{8}$$

where $\hat{e}$ is some unit vector.

Weak Duality for fixing dual variables

Let $\mathsf{SDPOPT}(G)$ denote the optimal value of the primal SDP; then from the proposed integral solution we have that,

$$\mathsf{SDPOPT}(G) \leq -2 \sum_{\{i,j\}\in E} \left\langle \mathbf{x}_i^*, \mathbf{x}_j^* \right\rangle = \left\langle A, \mathbf{g}\mathbf{g}^T \right\rangle = \mathbf{g}^T A \mathbf{g} = -d\,.$$

For any feasible solution to the dual SDP 9, by weak duality, we know that

$$\beta - \sum_{i\in V} \gamma_i \leq \mathsf{SDPOPT}(G) \leq -d\,.$$

We note that the upper bound is achievable by setting $\beta = -d$ and $\gamma_i = 0, \forall i \in V$.

We will show later that the remaining dual variables B_{ij}'s can be chosen in a way that the choice of $\beta = -d$ and $\gamma_i = 0, \forall i \in V$ yields a feasible dual solution.

▶ **Fact 10** (Folklore, also see Lemma 2.3 in [49]). *The primal solution $X = \mathbf{g}\mathbf{g}^T$ is the unique solution to SDP 8 if there exists a dual matrix Y such that it satisfies constraints in SDP 9, with $\beta = -d$ and $\gamma_i = 0, \forall i \in V$ and having* $\mathsf{rank}(Y) = n - 1$ *(i.e.* $\lambda_2(Y) > 0$*).*

1.3.4 Threshold rank eigenvectors

▶ **Definition 11** (Threshold rank of a graph). *For $\tau \in [0, d]$, we define threshold rank of a graph with adjacency matrix G (denoted by* $\mathsf{rank}_{\leq -\tau}(G)$*) as,*

$$\mathsf{rank}_{\leq -\tau}(G) = |\{i : \lambda_i(G) \leq -\tau\}| \ .$$

We let $P_{-\tau} = \left\{\mathbf{v}^{(1)}, \mathbf{v}^{(2)}, \ldots, \mathbf{v}^{(L_{-\tau})}\right\}$ (the bottom $L_{-\tau}$ vectors) denote the set of orthonormal eigenvectors of $A|_{S\times S}$ with eigenvalues smaller than the threshold $-\tau$, breaking ties arbitrarily where $L_{-\tau} = \mathsf{rank}_{\leq -\tau}\left(A|_{S\times S}\right)$. We call these vectors as *τ-threshold rank eigenvectors* of $A|_{S\times S}$. Next, we recall a well known fact about the threshold rank of a graph.

▶ **Fact 12** (Folklore). $\mathsf{rank}_{\leq -\tau}\left(A|_{S\times S}\right) \leq \dfrac{kd}{2\tau^2}$.

Proof. Since, $A|_{S\times S}$ is the adjacency matrix of a bipartite graph, it's eigenvalue spectrum is symmetric around 0. Therefore the number of eigenvalues with absolute value greater then or equal to τ is given by $2\,\mathsf{rank}_{\leq -\tau}\left(A|_{S\times S}\right)$ and are bounded as,

$$2\tau^2 \mathsf{rank}_{\leq -\tau}(A|_{S\times S}) \leq \sum_i \lambda_i^2(A|_{S\times S}) \leq \left\| A|_{S\times S}\right\|_F^2 = kd. \qquad ◀$$

We note that a similar notion of threshold rank has appeared in other works [8, 9, 12, 31] etc.

1.3.5 Spectral embedding vectors

▶ **Definition 13** (Spectral embedding vectors). *Given the planted bipartite graph S and the matrix of bottom $L_{-\tau}$ orthonormal eigenvectors $W_{-\tau}^T = \begin{bmatrix}\mathbf{v}^{(1)} & \mathbf{v}^{(2)} & \ldots & \mathbf{v}^{(L_{-\tau})}\end{bmatrix}$, we define the spectral embedding of a vertex $i \in S$ as the $L_{-\tau}$-dimensional vector given by $\mathbf{w}^{(i)} = W_{-\tau}\mathbf{e}_i$ where $\mathbf{e}_i$ is a vector with one in the i^{th} coordinate and zero elsewhere.*

Informally, these are the vectors obtained by looking at the subspace of the columns of $W_{-\tau}^T$ where the vertex i is mapped to the i^{th} column of $W_{-\tau}^T$. These spectral embedding vectors have been explored in various works on graph partitioning as [57, 45, 48] etc. It is known that these spectral embedding vectors are "well spread", formally referred to as being in an isotropic[4] position. We define these set of vectors to be in an isotropic position if $\sum_{i\in S} \mathbf{w}^{(i)} = 0$ and $\sum_{i\in S} \mathbf{w}^{(i)}\mathbf{w}^{(i)^T} = I$ where I is an $L_{-\tau} \times L_{-\tau}$ sized identity matrix. The condition that $\sum_{i\in S} \mathbf{w}^{(i)}\mathbf{w}^{(i)^T} = I$ can equivalently be written as $\sum_{i\in S} \left\langle \mathbf{y}, \mathbf{w}^{(i)}\right\rangle^2 = 1, \forall \mathbf{y}$ with $\|\mathbf{y}\|_2 = 1$.

▶ **Lemma 14** (Folklore). *The spectral embedding vectors are in an isotropic position.*

For a proof, we refer the reader to the work [47].

[4] Typically, isotropicity is a property of distribution. We say a distribution is isotropic if the mean of a random variable sampled from the distribution is zero and it's covariance matrix is an identity matrix.

1.4 Proof Overview

For the sake of simplicity, we will assume that the graph is sampled as per the random planted model (Definition 1). We will also allow an action of a monotone adversary (as in step 5) on this model; but we analyze its action separately (in Section 1.4.6). The main ideas for the semi-random model (Definition 2) are essentially the same, and the additional steps to handle them is just a technical adjustment.

In this section we give an overview of how Theorem 4 and Theorem 5 are proven. Because of page limits, the detailed proofs are deferred to the full version of the paper. In what follows, we present the main ideas which go inside these proofs.

1.4.1 Spectral Approaches

We start with some natural spectral approaches for recovering the planted set. These approaches have found some success, e.g. in recovering planted cliques/independent sets, planted bisection, planted k-colorable graphs (refer work [55] for details). We recall from our earlier discussion, that the interesting regimes for this problem are $k = \Omega\left(\sqrt{n}\right)$ and $d \approx pk$.

Detecting planted bipartitions and why it is easy

We note that the *detection problem* i.e. detecting the presence of bipartite graph as constructed in the random planted model (Definition 1) against the null hypothesis of Erdős-Rényi graph $G_{n,p}$, is easy when $k = \Omega\left(\sqrt{n}\right)$. Formally one notes that given two distributions

$$H_0 : G \sim G\left(n, p\right) \text{ against } H_1 : G \sim G\left(n, k, d, p\right) \text{ as per Definition 1,}$$

the spectral test, which outputs H_1 when $\lambda_1\left(G\right) \leq -d$ and H_0 otherwise, is correct almost surely for $d \approx pk$ and $k \geq c\sqrt{n}/p$ where $c > 0$ is a large enough constant. This is because for a $G_{n,p}$ graph, the smallest eigenvalue is greater than $-2\sqrt{n}$ almost surely (Claim 7), while for a graph with planted bipartite subgraph, the smallest eigenvalue is smaller than $-d$ since the vector $\mathbb{1}_{S_1} - \mathbb{1}_{S_2}$ already achieves Rayleigh Quotient of value $-d$.

The challenges in exact recovery

However, as expected, the exact recovery problem is more challenging. There are some works that look at these planted problems on an individual basis ([15],[6]). They typically rely on the spectral bounds of perturbation matrices and the framework of Davis-Kahan theorem (refer [61]) to identify eigenvector(s) indicating the planted set. However, we need a sufficient eigengap[5] to apply these results from perturbation theory. Since our planted graph S in the random planted model is an arbitrary bipartite graph, it can have any number of eigenvalues close to the smallest eigenvalue $-d$ and hence we may not have such an eigengap.

A unified spectral framework for random planted models was given by McSherry [55] (further refined in the work [62]). Here, one can check that we cannot satisfy the conditions in Observation 11 of this work [55] if the planted set has size $o(n)$. Again, the reason is because the planted bipartite graph is arbitrary. Since the planted bipartite graph can have arbitrary rank we cannot get the constants γ_1 in [55] to be small enough to recover in $o(n)$ regimes. It is also easy to verify that this framework works if the planted bipartite graph is also a random graph, for regimes of $k = \Omega\left(\sqrt{n}\right)$ and $d \approx pk$ (say by choosing edge probability for the random planted bipartite graph as $p' = 2p$).

[5] Typically around the bottom eigenvector(s) or the top eigenvector(s).

A subspace enumeration style approach

Another spectral approach, inspired from the works of [42, 8, 41, 44] is to apply the *subspace enumeration* technique to recover a large fraction of planted set S. Here we first identify that the vector $\mathbf{u} = \mathbb{1}_{S_1} - \mathbb{1}_{S_2}$ has a large projection on the space spanned by τ'-threshold rank eigenvectors of A (for choice of $\tau' = d/2$). Note that this vector $\mathbf{u}$ identifies the planted set (as well as the planted bipartition), and therefore we call it the *signed indicator vector*. We then do a standard ε-net construction to find a vector $\mathbf{y}$ close to $\mathbf{u}$ and use $\mathbf{y}$ to recover a large fraction of planted set S. We can recover the remaining set of vertices by an argument due to the work [33], where they distinguish vertices by the size of matching in induced neighborhoods. Putting all this together, we can prove Theorem 5.

The running time of the procedure described above is exponential in $L_{-\tau}$ where $L_{-\tau} = \mathcal{O}\left(1/p\right)$ for $\tau = \Omega\left(d\right)$ (follows from Fact 12). Therefore, for many special classes of instances such as, (i) when the probability $p = \Omega(1)$ and $d = \Omega(pk)$, (ii) when the planted graph is a complete bipartite graph (this is the balanced biclique problem) and $d = \Omega(pk)$, (iii) when the planted bipartite graph is d-regular random graph for $d = \Omega(pk)$ or (iv) more generally when the planted graph is a d-regular expander graph for $d = \Omega(pk)$ we have $L_{-\tau} = \mathcal{O}\left(1\right)$ and this already gives us a polynomial-time algorithm.

However, as stated earlier, we want to solve the problem in $p = o(1)$ regimes. To accomplish this, we shift our focus to the SDP formulation we mentioned in SDP 8. Also, for other problems in this literature (planted clique, planted bisection, planted k-colorable, Stochastic Block models etc), only SDP's have provable guarantees of working in the presence of such monotone adversaries (refer Chapter 10 in [59] for more intuition on this).

1.4.2 Traditional SDP Analysis

Now we overview our SDP-based approach to solving the problem. We will see that the difficulties in the spectral approach will translate to showing the feasibility of the dual SDP solution. However, we have more freedom here since we have the dual variables to work with and we can use them and try to enforce the optimality of the dual solution.

Characterizing dual variables through optimality conditions

A standard technique for analyzing SDP relaxations (like our SDP 8) is to show optimality by constructing a dual solution that matches the $\mathsf{SDPOPT}\left(G\right)$ value of the primal in a manner that the dual matrix Y is positive semi-definite and has rank $n-1$, (see Fact 10).

These impose a "wish list" of desired conditions, which can be used to characterize our dual variables

1. $\beta = -d$ (Optimal objective)
2. $\left\langle \mathbf{g}\mathbf{g}^T, Y\right\rangle = 0$ (Complementary slackness)
3. $Y \succeq 0$ (Dual feasibility)
4. $\lambda_2\left(Y\right) > 0$ (Strong duality).

Using weak duality we set $\beta = -d$ and $\gamma_i = 0, \forall i \in V$ to match the optimal primal objective value of $\mathsf{SDPOPT} = -d$. We expand upon the complementary slackness condition as,

$$\left\langle \mathbf{g}\mathbf{g}^T, Y\right\rangle = \mathbf{g}^T Y \mathbf{g} = \mathbf{g}^T\left(A + dI\right)\mathbf{g} + \mathbf{g}^T B\mathbf{g} = 0 + \mathbf{g}^T B\mathbf{g} = \mathbf{g}^T B\mathbf{g}\,.$$

Therefore the complementary slackness condition gives us that,

$$\sum_{i \in S} \sum_{\substack{j \in S \\ \{i,j\} \in E}} B_{ij} = 0 \tag{9}$$

and since the SDP dual requires that $B_{ij} \geq 0$, it implies that $B_{ij} = 0$ for all $(i, j) \in E(S_1, S_2)$. Now using the characterization of dual variables from conditions (1) and (2), one tries to show the feasibility of the dual and the strong duality rank condition. Typically, this characterization turns out to be rather weak. So, we refer to the dual variables set so far (ensuring condition (1) and (2) are satisfied) as *weakly characterized.*

Showing optimality of dual solution through weakly characterized dual variables

For certain problems in semi-random models, such as the planted clique problem [26], community detection in SBM [2], the weak characterization above suffices. We are able to show that the weakly characterized dual solution satisfies conditions (3) and (4). This is typically done by invoking some standard results for random matrix bounds and concentration inequalities. In our setup, satisfying condition (3) requires that

$$\lambda_{\min}(Y) \geq 0 \text{ which is implied if } \lambda_{\min}(A) + d + \lambda_{\min}(B) = \lambda_{\min}(A) + d \geq 0\,. \tag{10}$$

However, in our random planted model, the smallest eigenvalue of A can be smaller than $-d - 2\sqrt{pn}$ and condition (3) may not hold (as per choice of B_{ij}'s dictated from equation (9)). Thus we need a stronger characterization of dual variables to satisfy the conditions (3) and (4). In our problem, we need to make use of the large number of unused dual variables $B'_{ij}s$ for $\{i, j\} \in E \cap \{(V \times V) \setminus (S \times S)\}$

Guessing/Constructing the dual certificate

Now we discuss an approach of making the dual matrix satisfy conditions (3) and (4) by guessing the dual variables thus giving an explicit setting of dual variables. This is typically done by assigning some sort of meaning to dual variables and guessing their values based on the input instance. This approach has found reasonable success in other recovery problems like the planted bisection problem [24], coloring semi-random graphs [20], decoding binary node labels from censored edge measurements [1], and planted sparse vertex cuts [49].

Therefore, we may expect to guess a nice setting of dual variables that satisfy equation (10). However, if one takes a deeper look at this approach, the task again reduces to applying results from perturbation theory. Again, such an approach would work if the planted bipartite graph were also a random graph or an expander, since there would only be a single eigenvector whose corresponding eigenvalue disobeys equation (10), and one could choose the dual variables constructively to handle it and make it satisfy condition (3).

However, for an arbitrary planted bipartite graph, we can have a lot of eigenvalues in the interval $[-d, -d - 2\sqrt{n}]$ (and hence the entire graph A can have a lot of eigenvalues in the interval $[-d - 2\sqrt{n}, -d]$). Therefore, we need a more principled approach to deal with the corresponding eigenvectors of the planted graph having eigenvalue close to $-d$ (as we pointed out earlier, there can be $\mathcal{O}(1/p)$ such eigenvectors).

1.4.3 Calibrating the eigenvectors

Now we present our approach towards satisfying conditions (3) and (4), which is to *calibrate the eigenvectors.* We will see that, this calibration will further complicate our requirements on the dual variables, however we will argue in Section 1.4.4 on how we manage that.

Obtaining optimality of Primal SDP by assuming existence of a certifying B

It is now clear that our Achilles' heel are the eigenvalues (and corresponding threshold rank eigenvectors[6]) of the planted graph in the interval $[-d, -d - 2\sqrt{n}]$. If we were allowed to ignore these vectors it's easy to see that equation (10) and hence condition (3) holds.

Our core idea is to extend (by padding with 0's so that they are the right length) the threshold rank eigenvectors of $A|_{S\times S}$ to be the eigenvectors of the dual matrix Y. Recall, the eigenvalues of $A|_{S\times S}$ lie in the interval $[-d, d]$. Now take a threshold rank eigenvector of $A|_{S\times S}$ (say with eigenvalue λ_l). We wish to calibrate such a threshold rank eigenvector to be an eigenvector of Y with eigenvalue $d + \lambda_l$. If we are able to achieve this calibration, we need not bother about the $2\sqrt{n}$ term since now these eigenvectors have a non-negative quadratic form[7].

The only thing at our disposal for this calibration are the unused (so far) dual variables B_{ij}'s. Denote this set of threshold rank eigenvectors of $A|_{S\times S}$ as $P_{-\tau}$. Given this set $P_{-\tau}$, achieving this calibration can be expressed as satisfying the system of equations,

$$\sum_{i\in S} \mathbf{v}_i^{(l)} \left(A_{ij} + B_{ij}\right) = 0, \quad \forall j \in V \setminus S, \forall\, \mathbf{v}^{(l)} \in P_{-\tau}\,. \tag{11}$$

- L different system of equations $\mathcal{E}_l$, one for each $\mathbf{v}^{(l)}$.
- Each system $\mathcal{E}_l$ involves $|S| \times |V \setminus S|$ variables B_{ij} where $i \in S$ and $j \in V \setminus S$.

Now, all we need is a setting of B_{ij}'s such that the system of equations is satisfied. However, this will still not be enough. We note that if this system of equations were to have a solution, we would have set some of these B_{ij} variables to non-zero values. Therefore our equation (10) would now need to be modified to showing that $\lambda_{\min}(A) + d + \lambda_{\min}(B) \geq 0$.

The way we deal with this is by noting that we can tune $\tau > 0$ apriori to be sufficiently large for this calibration such that $\lambda_{\min}(A) + d \geq \eta$ where $\eta > 0$; and now impose an additional constraint on the matrix of dual variables that $\|B\| \leq \eta$. At this point, it seems highly suspicious as whether such B_{ij}'s exist. However, if we table these considerations aside and for choice of $\tau = 2d/3$ and $\eta = \tilde{\mathcal{O}}(pk)$ we can indeed show that condition (3) and condition (4) of our "wish list" are met and we get the desired integral primal solution.

1.4.4 Setting of dual variables

In this section we show that there exists a matrix of non-negative dual variables B_{ij}'s that satisfies the system of equations (11) and $\|B\|_2 = \tilde{\mathcal{O}}(pk)$.

An LP formulation and Farkas Lemma based approach

We start by observing that the condition $\|B\|_2 = \tilde{\mathcal{O}}(pk)$ is implied by a condition that $B_{ij} \leq t = \mathcal{O}_p\left(\sqrt{\log k/k}\right), \forall (i,j) \in (S \times (V \setminus S)) \cup ((V \setminus S) \times S)$. Also, since these system of equations (11) only concerns the non-negative dual variables B_{ij}'s with $\{i,j\} \in (S \times (V \setminus S)) \cup ((V \setminus S) \times S)$, we set the rest of them to 0.

[6] For an appropriate choice of τ, which we decide later, these will be the threshold rank eigenvectors.
[7] The quadratic form of vector $\mathbf{x}$ with a matrix Y is a number given by $\mathbf{x}^T Y \mathbf{x}$

We now reorganize our collection of linear systems in (11) as follows.

- For $j \in V \setminus S$, define a system of equations $\mathcal{F}_j$.
- In all, this gives a collection of systems $\{\mathcal{F}_j\}_{j \in V \setminus S}$. Each system contains $L_{-\tau} \times |S|$ variables. In particular, the system $\mathcal{F}_j$ is expressed in the standard form $W_{-\tau}\mathbf{x} = \mathbf{b}$, where $W_{-\tau} \in \mathbb{R}^{L_{-\tau} \times k}$ is a matrix formed by stacking the vectors $\mathbf{v}^{(l)} \in P_{-\tau}$ as rows.

Fix $j \in V \setminus S$ and consider the system $\mathcal{F}_j$. The vector $\mathbf{b}$ in this system is a row vector of size $L_{-\tau} \times 1$ and has entries given by $b_l = -\sum_{i \in S} A_{ij} v_i^{(l)}, \forall l \in [L_{-\tau}]$ and $\mathbf{x}$ here is a row vector of size $k \times 1$ where the entry $x_i = B_{ij}$ (recall that we have fixed a $j \in V \setminus S$). However since B_{ij}'s are not arbitrary variables but dual variables of SDP 9, these are required to be non-negative and should only be defined for $i \in N(j)$. Since the graph on $S \times (V \setminus S)$ is random, the choice of random edges while choosing $N(j)$ (in model construction) corresponds to setting those $B_{ij} = 0$ whenever the edge is not chosen. Let $\tilde{W}_{-\tau}$ denote the submatrix after removing the columns corresponding to $i \notin N(j)$ and recall t is our upper bound on the entries of B matrix as mentioned above. We then consider the following feasibility LP formulation for this problem of finding appropriate B_{ij}'s.

▶ **LP 15.**

$$\tilde{W}_{-\tau}\mathbf{x} = \mathbf{b} \tag{12}$$

$$0 \leq \mathbf{x} \leq t\mathbb{1}\,. \tag{13}$$

For simplicity, consider the case $p = 1$, i.e. when $A_{ij} = 1$ for all $i \in S, j \notin S$. Then we have that for any vector $\mathbf{y} \in \mathbb{R}^{L_{-\tau}}$,

$$\mathbf{b}^T\mathbf{y} = \sum_{r \in [L_{-\tau}]} b_r y_r = -\sum_{r \in [L_{-\tau}]} \sum_{i \in S} v_i^{(r)} y_r = -\sum_{i \in S} \sum_{r \in [L_{-\tau}]} v_i^{(r)} y_r \tag{14}$$

$$= -\sum_{i \in S} \sum_{r \in [L_{-\tau}]} w_r^{(i)} y_r = -\sum_{i \in S} \left\langle \mathbf{w}^{(i)}, y \right\rangle = -\left\langle \sum_{i \in S} \mathbf{w}^{(i)}, \mathbf{y} \right\rangle = 0\,. \tag{15}$$

Using the standard variant of Farkas' Lemma, this immediately implies the existence of a solution to equation (11). However, in general, for $p < 1$, we need to do more work here.

We apply a more general version of Farkas' Lemma, and we have that satisfying this LP in the general case corresponds to showing that for some $t > 0$, the following holds.

$$\forall\, \mathbf{y} \in \mathbb{R}^{L_{-\tau}}, \forall\, \mathbf{z} \geq 0,\ \tilde{W}_{-\tau}^T \mathbf{y} + \mathbf{z} \geq 0 \implies \mathbf{b}^T\mathbf{y} + t\,\langle \mathbf{z}, \mathbb{1} \rangle \geq 0\,. \tag{16}$$

The first term in the expression, $\mathbf{b}^T\mathbf{y}$, can be expanded as in equation (14) to obtain

$$\mathbf{b}^T\mathbf{y} = -\sum_{i \in N(j)} \left\langle \mathbf{w}^{(i)}, \mathbf{y} \right\rangle \text{ and using } \left(\tilde{W}_{-\tau}^T \mathbf{y}\right)_i = \left\langle \mathbf{w}^{(i)}, \mathbf{y} \right\rangle \text{ we have } z_i \geq -\left\langle \mathbf{w}^{(i)}, \mathbf{y} \right\rangle.$$

We can give a proof by contradiction for equation (16). By contradiction there exists a $\mathbf{y}$ and a $\mathbf{z}$ such that $\mathbf{b}^T\mathbf{y} + t\,\langle \mathbf{z}, \mathbb{1} \rangle < 0$. We choose $\mathbf{z}' \leq \mathbf{z}$ by setting $z'_i = \max\left\{0, -\left\langle \mathbf{w}^{(i)} \right\rangle, \mathbf{y}\right\}$ and argue that it is enough to show contradiction for $t, \mathbf{y}$ and $\mathbf{z}'$. Using the expressions for $\mathbf{b}^T\mathbf{y}$ as above, this translates to showing that

$$\sum_{i \in N(j)} \left\langle \mathbf{w}^{(i)}, \mathbf{y} \right\rangle + t \sum_{i \in N(j)} \min\left\{0, \left\langle \mathbf{w}^{(i)}, \mathbf{y} \right\rangle\right\} > 0, \tag{17}$$

has no solution. We show that equation (17) does not hold for our desired choice of $t = \mathcal{O}_p\left(\sqrt{\log k/k}\right)$.

We note that the second term in equation (17) is ≥ 0, and we wish the inequality to not hold for as small a value of t as possible; therefore, we seek an upper bound on both terms.

Structure of threshold rank/spectral embedding vectors

To upper bound the first term, it might be helpful to understand the structure of the spectral embedding vectors $\mathbf{w}^{(i)}$. Since these are intimately connected to the threshold rank eigenvectors $\mathbf{v}^{(l)}$, we use these eigenvectors to characterize them. For convenience we let $\mathbf{v}^{(l)} \in P_{-\tau}$ have unit norm, then we show that $\left\|\mathbf{v}^{(l)}\right\|_\infty = \left\|\mathbf{w}^{(i)}\right\|_\infty \leq 2/\sqrt{d}$. Since $i \in N(j)$ are sampled randomly, we can use the Hoeffding bounds to upper bound the l_∞ norm for $\sum_{i \in N(j)} \mathbf{w}^{(i)}$ and hence upper bound our first term by $\mathcal{O}_p\left(\sqrt{\log k}\right)$ with high probability. We choose our parameters such that the vectors in $P_{-\tau}$ are orthogonal to $\mathbb{1}_S$. For the embedding vectors, this translates to saying that $\sum_{i \in S} \mathbf{w}^{(i)} = 0$.

Towards bounding the second term, we use these *spectral embedding vectors.* The spectral embedding vectors are isotropic for $p = 1$ (already where we can easily show that equation (17) does not hold and we are done). However, for $p < 1$, we have $i \in N(j)$ (and corresponding embedding vectors) being sampled randomly as per $G_{n,p}$ distribution. Here, we show that by using Matrix Bernstein concentration we can get close to isotropic vectors,

$$\sum_{i \in N(j)} \left\langle \mathbf{y}, \mathbf{w}^{(i)} \right\rangle^2 \geq p/2 \quad \text{(This shows that embedding vectors are } p/2\text{-isotropic .)} \qquad (18)$$

Showing existence of a solution to LP 15

Now, we look at two cases; the first case where the negative terms dominate the summand in equation (17), then we use the eigenvector structure that $\left\|\mathbf{w}^{(i)}\right\| \leq 2/\sqrt{d}$ and we are done; for the other case where the positive terms dominate, we relate the positive terms to the negative terms again using the bound we obtained from the eigenvector structure $\left\|\sum_{i \in N(j)} \mathbf{w}^{(i)}\right\| = \mathcal{O}_p\left(\sqrt{\log k}\right)$.

Therefore, we argue that we can upper bound the second term by $-\Omega_p\left(\sqrt{k}\right)$. Therefore for a choice of t as we obtained above of $t = \mathcal{O}_p\left(\sqrt{\log k / k}\right)$, equation (17) does not hold. As discussed earlier this implies that the there exists a dual such that conditions (1)-(4), equation (11) and $\|B\|_2 = \tilde{\mathcal{O}}\left(pk\right)$ holds which further implies that the primal SDP is feasible. Further, if the graph is connected; the *signed indicator vector* would be the only eigenvector with eigenvalue $-d$ (after padding to make these the eigenvectors of Y), this would be the only eigenvector of Y with eigenvalue 0. Using Fact 10, this implies that the proposed integral solution in equation (8) is the only integral solution and hence Cholesky Decomposition of our SDP matrix returns the *signed indicator vector* and thus our planted set.

1.4.5 Low degree regimes

The discussion above about SDP holds only where $d = \gamma pk$ where $\gamma \geq 2/3$. Note that this covers our interesting regimes when $d \approx pk$ where the problem is non-trivial. The other case where $d \leq 2pk/3$, can actually be trivially solved for $k = \Omega_p\left(\sqrt{n \log n}\right)$ using a degree counting argument along the lines of [43] as we discuss below.

Now we consider the regimes when $d = \gamma pk$ with $\gamma \leq 2/3$. We show that a simple algorithm that collects the bottom k degrees of the graph will work in these regimes since the vertices in S will have smaller degrees compared to vertices in $V \setminus S$.

▶ **Lemma 16.** *For $k \geq 6\sqrt{6n \log n/p}$, Algorithm 1 returns the planted set S with high probability (over the randomness of the input).*

Algorithm 1 Kucera's algorithm for recovery in low degree regimes.

```
Input: $G = (V, E)$, sampled as per Definition 1 with adversary as per step 5.
Output: The set of vertices in planted bipartite graph (with high probability) $S$.
1: Sort the degrees of the vertices in $G$.
2: Return $\mathcal{S}$ be the set of bottom $k$ degrees after sorting..
```

Proof. For a vertex $v \in S$ the expected degree is $d + p(n-k)$. We note that this is smaller than pn since $d \leq 2pk/3$. We can upper bound the degree of v (denoted $d(v)$), with high probability (over the randomness of the input) using Chernoff bounds (Lemma 4.5, [56]) as,

$$\mathbb{P}\left[d(v) \geq d + p(n-k) + \sqrt{6pn\log n}\right] \leq \exp\left(\frac{-pn(\sqrt{6\log n}/\sqrt{pn})^2}{3}\right) = \frac{1}{n^2}.$$

Using a union bound over all $v \in S$, we have that for an any $v \in S$,

$$\mathbb{P}\left[d(v) \geq d + p(n-k) + \sqrt{6pn\log n}\right] \leq \frac{1}{n}.$$

Therefore we have with high probability (over the randomness of the input) that $d(v) \leq d + p(n-k) + \sqrt{6pn\log n}$. Similarly for a vertex $v' \notin S$ the degree can be lower bounded with high probability (over the randomness of the input) using Chernoff bounds (Lemma 4.4, [56]) as,

$$\mathbb{P}\left[d(v') \leq pn - \sqrt{6pn\log n}\right] \leq \exp\left(\frac{-pn(\sqrt{6\log n}/\sqrt{pn})^2}{2}\right) = \exp\left(-3\log n\right) = \frac{1}{n^3}.$$

Now using a union bound over all $v' \notin S$, we have with high probability (over the randomness of the input) that $d(v') \geq pn - \sqrt{6pn\log n}$. Therefore, with high probability (over the randomness of the input), the degrees differ by,

$$d(v') - d(v) \geq pk - d - 2\sqrt{6pn\log n} \geq \frac{pk}{3} - 2\sqrt{6n\log n}\,. \tag{19}$$

where we have used $d \leq 2pk/3$. It is also evident from equation (19) that for $k \geq 6\sqrt{6n\log n/p}$, with high probability (over the randomness of the input), the degree for a vertex $v \in S$ is smaller than degree of any vertex $v' \notin S$. ◀

1.4.6 Action of Adversary

Finally, we discuss the action of adversary (allowed to add edges in $(V \setminus S) \times (V \setminus S)$ for $d \geq 2pk/3$). We show that the inductive argument given by [24] also works for our case. This argument also extends to the semi-random model in Definition 11. We leave these details to the full version of the paper.

For $d \leq 2pk/3$ regimes, Algorithm 1 continues to return the planted set, since the action of adversary only amplifies the difference of degree for a vertex $v \in S$ and vertices $v' \notin S$. This argument does not extend to the semi-random model in Definition 11.

References

1 Emmanuel Abbe, Afonso S. Bandeira, Annina Bracher, and Amit Singer. Decoding binary node labels from censored edge measurements: phase transition and efficient recovery. *IEEE Trans. Network Sci. Eng.*, 1(1):10–22, 2014. doi:10.1109/TNSE.2014.2368716.

2 Emmanuel Abbe, Afonso S. Bandeira, and Georgina Hall. Exact recovery in the stochastic block model. *IEEE Trans. Inform. Theory*, 62(1):471–487, 2016. doi:10.1109/TIT.2015.2490670.

3 Amit Agarwal, Moses Charikar, Konstantin Makarychev, and Yury Makarychev. $O(\sqrt{\log n})$ approximation algorithms for Min UnCut, Min 2CNF deletion, and directed cut problems. In *STOC'05: Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pages 573–581. ACM, New York, 2005. doi:10.1145/1060590.1060675.

4 A. Agrawal, P. Klein, S. Rao, and R. Ravi. Approximation through multicommodity flow. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 726–737 vol.2, Los Alamitos, CA, USA, October 1990. IEEE Computer Society. doi:10.1109/FSCS.1990.89595.

5 Noga Alon and Nabil Kahalé. A spectral technique for coloring random 3-colorable graphs. *SIAM J. Comput.*, 26(6):1733–1748, 1997.

6 Noga Alon, Michael Krivelevich, and Benny Sudakov. Finding a large hidden clique in a random graph. In *Proceedings of the Eighth International Conference "Random Structures and Algorithms" (Poznan, 1997)*, volume 13, pages 457–466, 1998. doi:10.1002/(SICI)1098-2418(199810/12)13:3/4<457::AID-RSA14>3.3.CO;2-K.

7 Claudio Arbib and Raffaele Mosca. Polynomial algorithms for special cases of the balanced complete bipartite subgraph problem. *JCMCC. The Journal of Combinatorial Mathematics and Combinatorial Computing*, 30, January 1999.

8 Sanjeev Arora, Boaz Barak, and David Steurer. Subexponential algorithms for unique games and related problems. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science—FOCS 2010*, pages 563–572. IEEE Computer Soc., Los Alamitos, CA, 2010.

9 Sanjeev Arora and Rong Ge. New tools for graph coloring. In *Approximation, randomization, and combinatorial optimization*, volume 6845 of *Lecture Notes in Comput. Sci.*, pages 1–12. Springer, Heidelberg, 2011. doi:10.1007/978-3-642-22935-0_1.

10 Nikhil Bansal and Subhash Khot. Optimal long code test with one free bit. In *2009 50th Annual IEEE Symposium on Foundations of Computer Science—FOCS 2009*, pages 453–462. IEEE Computer Soc., Los Alamitos, CA, 2009. doi:10.1109/FOCS.2009.23.

11 Boaz Barak, Samuel B. Hopkins, Jonathan Kelner, Pravesh Kothari, Ankur Moitra, and Aaron Potechin. A nearly tight sum-of-squares lower bound for the planted clique problem. In *57th Annual IEEE Symposium on Foundations of Computer Science—FOCS 2016*, pages 428–437. IEEE Computer Soc., Los Alamitos, CA, 2016.

12 Boaz Barak, Prasad Raghavendra, and David Steurer. Rounding semidefinite programming hierarchies via global correlation. In *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science—FOCS 2011*, pages 472–481. IEEE Computer Soc., Los Alamitos, CA, 2011. doi:10.1109/FOCS.2011.95.

13 Aditya Bhaskara, Moses Charikar, Eden Chlamtac, Uriel Feige, and Aravindan Vijayaraghavan. Detecting high log-densities—an $O(n^{1/4})$ approximation for densest k-subgraph. In *STOC'10—Proceedings of the 2010 ACM International Symposium on Theory of Computing*, pages 201–210. ACM, New York, 2010.

14 Avrim Blum and Joel Spencer. Coloring random and semi-random k-colorable graphs. *J. Algorithms*, 19(2):204–234, 1995.

15 Ravi B. Boppana. Eigenvalues and graph bisection: An average-case analysis. In *28th Annual Symposium on Foundations of Computer Science (sfcs 1987)*, pages 280–285, 1987. doi:10.1109/SFCS.1987.22.

16 T. N. Bui, S. Chaudhuri, F. T. Leighton, and M. Sipser. Graph bisection algorithms with good average case behavior. *Combinatorica*, 7(2):171–191, 1987. doi:10.1007/BF02579448.

17 Ted Carson and Russell Impagliazzo. Hill-climbing finds random planted bisections. In *Proceedings of the Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms (Washington, DC, 2001)*, pages 903–909. SIAM, Philadelphia, PA, 2001.

18 Yudong Chen and Jiaming Xu. Statistical-computational tradeoffs in planted problems and submatrix localization with a growing number of clusters and submatrices. *J. Mach. Learn. Res.*, 17:Paper No. 27, 57, 2016.

19 Yizong Cheng and George M. Church. Biclustering of expression data. In Philip E. Bourne, Michael Gribskov, Russ B. Altman, Nancy Jensen, Debra A. Hope, Thomas Lengauer, Julie C. Mitchell, Eric D. Scheeff, Chris Smith, Shawn Strande, and Helge Weissig, editors, *ISMB*, pages 93–103. AAAI, 2000.

20 Amin Coja-Oghlan. Colouring semirandom graphs. *Combin. Probab. Comput.*, 16(4):515–552, 2007. doi:10.1017/S0963548306007917.

21 Anne Condon and Richard M. Karp. Algorithms for graph partitioning on the planted partition model. *Random Structures Algorithms*, 18(2):116–140, 2001. doi:10.1002/1098-2418(200103)18:2<116::AID-RSA1001>3.0.CO;2-2.

22 M. E. Dyer and A. M. Frieze. The solution of some random NP-hard problems in polynomial expected time. *J. Algorithms*, 10(4):451–489, 1989. doi:10.1016/0196-6774(89)90001-1.

23 David Eppstein. Arboricity and bipartite subgraph listing algorithms. *Inform. Process. Lett.*, 51(4):207–211, 1994. doi:10.1016/0020-0190(94)90121-X.

24 Uriel Feige and Joe Kilian. Heuristics for semirandom graph problems. *Journal of Computing and System Sciences*, 63:639–671, 2001.

25 Uriel Feige and Shimon Kogan. Hardness of approximation of the balanced complete bipartite subgraph problem. Technical report, Weizmann Institute, 2004.

26 Uriel Feige and Robert Krauthgamer. Finding and certifying a large hidden clique in a semirandom graph. *Random Structures Algorithms*, 16(2):195–208, 2000. doi:10.1002/(SICI)1098-2418(200003)16:2<195::AID-RSA5>3.3.CO;2-1.

27 Uriel Feige and Dorit Ron. Finding hidden cliques in linear time. In *21st International Meeting on Probabilistic, Combinatorial, and Asymptotic Methods in the Analysis of Algorithms (AofA'10)*, Discrete Math. Theor. Comput. Sci. Proc., AM, pages 189–203. Assoc. Discrete Math. Theor. Comput. Sci., Nancy, 2010.

28 Vitaly Feldman, Elena Grigorescu, Lev Reyzin, Santosh S. Vempala, and Ying Xiao. Statistical algorithms and a lower bound for detecting planted cliques. In *STOC'13—Proceedings of the 2013 ACM Symposium on Theory of Computing*, pages 655–664. ACM, New York, 2013. doi:10.1145/2488608.2488692.

29 Michael R. Garey and David S. Johnson. *Computers and intractability.* A Series of Books in the Mathematical Sciences. W. H. Freeman and Co., San Francisco, Calif., 1979. A guide to the theory of NP-completeness.

30 Naveen Garg, Vijay Vazirani, and Mihalis Yannakakis. Approximate max-flow min-(multi)cut theorems and their applications. *SIAM Journal on Computing*, 25, January 1998. doi:10.1137/S0097539793243016.

31 Shayan Oveis Gharan and Luca Trevisan. Improved arv rounding in small-set expanders and graphs of bounded threshold rank, 2013. arXiv:1304.2060.

32 Suprovat Ghoshal and Anand Louis. Approximation algorithms and hardness for strong unique games. In Dániel Marx, editor, *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms, SODA 2021, Virtual Conference, January 10 - 13, 2021*, pages 414–433. SIAM, 2021. doi:10.1137/1.9781611976465.26.

33 Suprovat Ghoshal, Anand Louis, and Rahul Raychaudhury. Approximation algorithms for partially colorable graphs. In *Approximation, randomization, and combinatorial optimization. Algorithms and techniques*, volume 145 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. No. 28, 20. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2019.

34 Bruce Hajek, Yihong Wu, and Jiaming Xu. Achieving exact cluster recovery threshold via semidefinite programming. *IEEE Trans. Inform. Theory*, 62(5):2788–2797, 2016. doi:10.1109/TIT.2016.2546280.

35 Bruce Hajek, Yihong Wu, and Jiaming Xu. Achieving exact cluster recovery threshold via semidefinite programming: extensions. *IEEE Trans. Inform. Theory*, 62(10):5918–5937, 2016. doi:10.1109/TIT.2016.2594812.

36 Bruce Hajek, Yihong Wu, and Jiaming Xu. Semidefinite programs for exact recovery of a hidden community, 2016. arXiv:1602.06410.

37 Mark Jerrum and Gregory B. Sorkin. The Metropolis algorithm for graph bisection. *Discrete Appl. Math.*, 82(1-3):155–175, 1998. doi:10.1016/S0166-218X(97)00133-9.

38 David S Johnson. The np-completeness column: An ongoing guide. *Journal of Algorithms*, 8(3):438–448, 1987. doi:10.1016/0196-6774(87)90021-6.

39 Yash Khanna and Anand Louis. Planted models for the densest k-subgraph problem. In *40th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science*, volume 182 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. 27, 18. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2020.

40 Yash Khanna, Anand Louis, and Rameesh Paul. Independent sets in semi-random hypergraphs. In Anna Lubiw, Mohammad Salavatipour, and Meng He, editors, *Algorithms and Data Structures*, pages 528–542, Cham, 2021. Springer International Publishing.

41 Alexandra Kolla. Spectral algorithms for unique games. *Comput. Complexity*, 20(2):177–206, 2011. doi:10.1007/s00037-011-0011-7.

42 Alexandra Kolla and Madhur Tulsiani. Playing random and expanding unique games, 2007.

43 Ludek Kucera. Expected complexity of graph partitioning problems. *Discret. Appl. Math.*, 57(2-3):193–212, 1995. doi:10.1016/0166-218X(94)00103-K.

44 Akash Kumar, Anand Louis, and Madhur Tulsiani. Finding pseudorandom colorings of pseudorandom graphs. In *37th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science*, volume 93 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. No. 37, 12. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2017.

45 James R. Lee, Shayan Oveis Gharan, and Luca Trevisan. Multi-way spectral partitioning and higher-order Cheeger inequalities. In *STOC'12—Proceedings of the 2012 ACM Symposium on Theory of Computing*, pages 1117–1130. ACM, New York, 2012. doi:10.1145/2213977.2214078.

46 Yevgeny Levanzov. On finding large cliques in random and semi-random graphs. Master's thesis, Weizmann Institute of Science, January 2018.

47 Anand Louis, Prasad Raghavendra, Prasad Tetali, and Santosh Vempala. Algorithmic extensions of Cheeger's inequality to higher eigenvalues and partitions. In *Approximation, randomization, and combinatorial optimization*, volume 6845 of *Lecture Notes in Comput. Sci.*, pages 315–326. Springer, Heidelberg, 2011. doi:10.1007/978-3-642-22935-0_27.

48 Anand Louis, Prasad Raghavendra, Prasad Tetali, and Santosh Vempala. Many sparse cuts via higher eigenvalues. In *STOC'12—Proceedings of the 2012 ACM Symposium on Theory of Computing*, pages 1131–1140. ACM, New York, 2012. doi:10.1145/2213977.2214079.

49 Anand Louis and Rakesh Venkat. Semi-random graphs with planted sparse vertex cuts: algorithms for exact and approximate recovery. In *45th International Colloquium on Automata, Languages, and Programming*, volume 107 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. No. 101, 15. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2018.

50 Anand Louis and Rakesh Venkat. Planted models for k-way edge and vertex expansion. In *39th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science*, volume 150 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. No. 23, 15. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2019.

51 Konstantin Makarychev, Yury Makarychev, and Aravindan Vijayaraghavan. Approximation algorithms for semi-random partitioning problems. In *STOC'12—Proceedings of the 2012 ACM Symposium on Theory of Computing*, pages 367–384. ACM, New York, 2012. doi:10.1145/2213977.2214013.

52 Konstantin Makarychev, Yury Makarychev, and Aravindan Vijayaraghavan. Constant factor approximation for balanced cut in the PIE model. In *STOC'14—Proceedings of the 2014 ACM Symposium on Theory of Computing*, pages 41–49. ACM, New York, 2014.

53 Pasin Manurangsi. Inapproximability of maximum edge biclique, maximum balanced biclique and minimum k-cut from the small set expansion hypothesis. In *44th International Colloquium on Automata, Languages, and Programming*, volume 80 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. No. 79, 14. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2017.

54 Theo McKenzie, Hermish Mehta, and Luca Trevisan. A new algorithm for the robust semi-random independent set problem. In Shuchi Chawla, editor, *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, pages 738–746. SIAM, 2020. doi:10.1137/1.9781611975994.45.

55 Frank McSherry. Spectral partitioning of random graphs. In *42nd IEEE Symposium on Foundations of Computer Science (Las Vegas, NV, 2001)*, pages 529–537. IEEE Computer Soc., Los Alamitos, CA, 2001.

56 Michael Mitzenmacher and Eli Upfal. *Probability and computing.* Cambridge University Press, Cambridge, second edition, 2017. Randomization and probabilistic techniques in algorithms and data analysis.

57 Andrew Y. Ng, Michael I. Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In Thomas G. Dietterich, Suzanna Becker, and Zoubin Ghahramani, editors, *Advances in Neural Information Processing Systems 14 [Neural Information Processing Systems: Natural and Synthetic, NIPS 2001, December 3-8, 2001, Vancouver, British Columbia, Canada]*, pages 849–856. MIT Press, 2001. URL: https://proceedings.neurips.cc/paper/2001/hash/801272ee79cfde7fa5960571fee36b9b-Abstract.html.

58 René Peeters. The maximum edge biclique problem is NP-complete. *Discrete Appl. Math.*, 131(3):651–654, 2003. doi:10.1016/S0166-218X(03)00333-0.

59 Tim Roughgarden. *Beyond the Worst-Case Analysis of Algorithms.* Cambridge University Press, 2021. doi:10.1017/9781108637435.

60 A. Tanay, R. Sharan, and R. Shamir. Discovering statistically significant biclusters in gene expression data. *Bioinformatics*, 18 Suppl 1:S136–44, 2002.

61 Roman Vershynin. *High-dimensional probability*, volume 47 of *Cambridge Series in Statistical and Probabilistic Mathematics.* Cambridge University Press, Cambridge, 2018. An introduction with applications in data science, With a foreword by Sara van de Geer. doi:10.1017/9781108231596.

62 Van Vu. A simple SVD algorithm for finding hidden partitions. *Combin. Probab. Comput.*, 27(1):124–140, 2018. doi:10.1017/S0963548317000463.

63 Van H. Vu. Spectral norm of random matrices. *Combinatorica*, 27(6):721–736, 2007. doi:10.1007/s00493-007-2190-z.

64 Mihalis Yannakakis. Node-and edge-deletion np-complete problems. In *Proceedings of the Tenth Annual ACM Symposium on Theory of Computing*, STOC '78, pages 253–264, New York, NY, USA, 1978. Association for Computing Machinery. doi:10.1145/800133.804355.

65 Yun Zhang. Elissa J. Chesler, Michael A. Langston: On finding bicliques in bipartite graphs: a novel algorithm with application to the integration of diverse biological data types. In *41st Hawaii International International Conference on Systems Science (HICSS-41 2008), Proceedings, 7-10 January 2008, Waikoloa, Big Island, HI, USA*, page 473. IEEE Computer Society, 2008. doi:10.1109/HICSS.2008.507.