Optimal Robust Community Detection

Samuel B. Hopkins* Yiding Hua† Sidhanth Mohanty‡ Amit Rajaraman§
David Steurer¶

November 18, 2026

Abstract

We study community detection in the sparse two-community stochastic block model when an adversary may corrupt a small constant fraction of the edges of the observed graph. For every choice of parameters above the Kesten–Stigum threshold and every $\varepsilon > 0$, we give a nearly-linear-time algorithm which estimates the communities with accuracy within ε of the information-theoretically optimal accuracy, while tolerating a small constant fraction of adversarially corrupted edges. Previously, efficient algorithms with optimal accuracy were known only in the absence of corruptions, while robust algorithms either achieved only weak recovery or approached optimal accuracy only in the limit of large signal-to-noise ratio.

Our algorithm is simple: approximate the second eigenvector of the nonbacktracking matrix, using a semidefinite program to cope with the corruptions, and use it to initialize belief propagation. Along the way, we show that in the absence of corruptions, belief propagation initialized at this eigenvector achieves optimal accuracy, resolving an open problem of Zdeborová in the two-community case. The proof of this fact is short and independent of our robustness results—its key ingredient is a local characterization of the second eigenvector of the nonbacktracking matrix in terms of constant-depth nonbacktracking walks.

*MIT EECS and CSAIL. Email: samhop@mit.edu. Supported by NSF award no. 2238080, MLA@CSAIL, FinTechAI@CSAIL, MIT Research Support Committee, MIT-Google Program for Computing Innovation.

†ETH Zurich. Email: yiding.hua@inf.ethz.ch.

‡Northwestern University. Email: sidhanth.mohanty@northwestern.edu.

§MIT EECS and CSAIL. Email: amit_r@mit.edu.

¶ETH Zurich. Email: dsteurer@inf.ethz.ch.

Contents

1 Introduction **1**
- 1.1 Related work . . . 4

2 Technical overview **5**

3 Preliminaries **12**
- 3.1 Stochastic block model and community detection . . . 12
- 3.2 Nonbacktracking matrix . . . 13
- 3.3 Broadcast trees, belief propagation, and optimal recovery . . . 13

4 Initialization of belief propagation at the second eigenvector of the nonbacktracking matrix **15**
- 4.1 The eigenvector is close to a constant-depth walk score . . . 18
- 4.2 BP from the walk score is near-optimal . . . 19
- 4.3 Stability of BP . . . 20

5 Robustly estimating the second eigenvector of the nonbacktracking matrix **24**

6 Negative eigenvalues of the Bethe Hessian **28**
- 6.1 Reduction to the broadcast tree . . . 30
- 6.2 Proof of the tree estimates . . . 32

7 Analysis of degree-cutoff preprocessing **34**

8 SDP recovery and implementation **37**
- 8.1 Recovery analysis . . . 39
- 8.2 Fast implementation . . . 40

9 Analysis of postprocessing by lifting **43**

10 AI methodology **46**

References **48**

A Proof of Lemma 6.9 **54**

B Deferred proofs for fast implementation **57**

1 Introduction

The *stochastic block model* is a long-studied random graph model for graphs with *community structure*. In the n-vertex, degree-d two-community stochastic block model (SBM), $\mathrm{SBM}(n, d, \lambda)$, we first assign to each vertex $u \in [n]$ a random community label $\boldsymbol{\sigma}^\star_u \sim \{\pm 1\}$, then add each possible edge $\{u, v\}$ independently with probability $\frac{d}{n}(1 + \lambda \boldsymbol{\sigma}^\star_u \boldsymbol{\sigma}^\star_v)$ to form a graph $\boldsymbol{G}$, where $\lambda \in (-1, 1)$ captures the degree of (dis)affinity within communities.

The SBM arose independently in several research communities – algorithms [BCLS87, DF89, Bop87], statistics and the social sciences [HLL83], statistical physics [Söd02, DKMZ11b, DKMZ11a], and mathematics [BJR07]. Across all of them, it is an important testbed for algorithmic and analytical techniques, since it provides a (relatively) simple model organism for a wide range of phenomena spanning clustering, spin glasses, social networks, and more. In computer science, studying the SBM has led to new clustering algorithms [DKMZ11a, KMM+13, SKZ14, GV16, MS16, HS17, AS18], new insights about average-case computational complexity [DKMZ11a, HS17, Hop18, BBK+21, KWB22], novel understanding of the differences between random and semirandom input models in beyond-worst-case analysis [MPW16], and much more.

We focus on the sparse graph setting, with constant average degree d, where the most interesting algorithmic and probabilistic phenomena occur. A long line of work has tackled the problem of estimating the community labels $\boldsymbol{\sigma}^*$ after observing the sparse graph $\boldsymbol{G}$ via computationally efficient algorithms [CO10, DKMZ11a, KMM+13, Mas14, SKZ14, AS18, Abb18]. A major theme has been the pursuit of algorithms which work for all d, λ obeying $d\lambda^2 > 1$, the *Kesten–Stigum threshold* – this is the best that can be hoped for, since if $d\lambda^2 < 1$ estimating $\boldsymbol{\sigma}^\star$ with any constant advantage over oblivious guessing is information-theoretically impossible [MNS15].

Some highlights include [Mas14, MNS18, BLM18], which gave the first polynomial-time estimators to perform better than randomly guessing the labels (the *weak recovery problem*) all the way down to the Kesten–Stigum threshold, [MNS16a, YP24] which give the first polynomial-time algorithm with information-theoretically optimal recovery rate (see also [GP23] for more than two communities), and [GV16, BMR21, DdNS21, DdHS23, DHL+26, MRW24, LM22, MMV12, HWX16, CCAd+23, BJK+24] which aim to design *robust* algorithms to estimate $\boldsymbol{\sigma}^*$, that is, algorithms which can tolerate some adversarial corruption of the observed graph.

Robust estimation of communities with optimal error rate We focus on the latter problem, estimating the community labels $\boldsymbol{\sigma}^*$ given an *adversarial corruption* G' of the SBM graph $\boldsymbol{G}$. We give the first polynomial-time algorithm which recovers the ground-truth labels with information-theoretically optimal accuracy for every d and λ while tolerating a constant fraction of adversarially corrupted edges. More formally, for every d, λ, there is a number $\mathsf{OPT}_{d,\lambda} \in [0, 1)$, defined as

$$\mathsf{OPT}_{d,\lambda} = \lim_{n\to\infty} \sup_{\text{estimators } \widehat{\sigma}} \frac{1}{n} \mathop{\mathbb{E}}_{(\boldsymbol{\sigma}^\star, \boldsymbol{G}) \sim \mathrm{SBM}(n,d,\lambda)} \left| \langle \boldsymbol{\sigma}^\star, \widehat{\sigma}(\boldsymbol{G}) \rangle \right| .$$

For any constant d, λ, $\mathsf{OPT}_{d,\lambda} < 1$ – indeed, with high probability the graph $\boldsymbol{G}$ will have isolated vertices, for which the graph structure gives no useful information about their label, though this is not the only reason for $\mathsf{OPT} < 1$; it is also strictly smaller than the size of the giant component.

Our main result is an algorithm which for every $d > 1$, and $\varepsilon > 0$, and $\lambda \in [-1, 1]$, obtains recovery accuracy $\mathsf{OPT}_{d,\lambda} - \varepsilon$, even in the presence of a small constant fraction $\eta > 0$ of adversarial corruptions.

Theorem 1.1 (Main theorem, follows from Theorems 4.2 and 5.1)**.** *For every $d > 1$, $\varepsilon > 0$, and $\lambda \in [-1, 1]$, there is $\eta > 0$ and a nearly-linear time algorithm which takes a graph G, returns a $\{\pm 1\}$-labeling $\widehat{\sigma}(G)$ of the vertices, and satisfies*

$$\mathop{\mathbb{E}}_{(\sigma^\star, \boldsymbol{G}) \sim \mathrm{SBM}(n,d,\lambda)} \min\left\{ \frac{1}{n} \cdot |\langle \sigma^\star, \widehat{\sigma}(G') \rangle| \text{ s.t. } |E(\boldsymbol{G}) \triangle E(G')| \leqslant \eta n \right\} \geqslant \mathsf{OPT}_{d,\lambda} - \varepsilon - o(1) \,.$$

Here $E(\boldsymbol{G}) \triangle E(G')$ denotes the symmetric difference between the edges of the graphs $\boldsymbol{G}$ and G', capturing that we allow an adversary to add or remove any ηn edges.

Prior efficient algorithms that obtained correlation $\mathsf{OPT}_{d,\lambda} - \varepsilon$ but did not tolerate any adversarial corruption [MNS16a, YP24] were already a major achievement. Prior efficient algorithms which do tolerate adversarial corruption either do not obtain information-theoretically optimal accuracy [DdNS21, DdHS23, MRW24], or obtain information-theoretically optimal accuracy only in the limit of large signal-to-noise ratio $d\lambda^2 \to \infty$ [LM22, DHL+26], rather than for every d, λ such that $d\lambda^2 > 1$—as we will see, the latter requires a substantially different algorithm. Our algorithm runs in nearly-linear time.

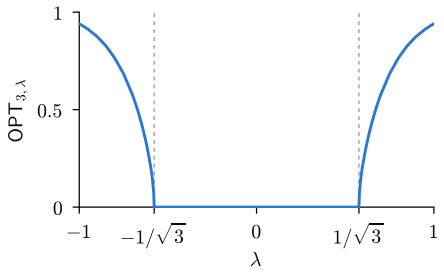


Figure 1: $\mathsf{OPT}_{d,\lambda}$ plotted for $d = 3$. The Kesten-Stigum threshold occurs at $1/\sqrt{3}$. Even at $\lambda = 1$ we have $\mathsf{OPT}_{d,1} < 1$ because of isolated vertices. The accuracy of our algorithm is described precisely by this curve.

Why adversarial robustness? Algorithmic techniques designed for average-case models like the SBM can overfit to their models – they may exploit features of the random input which have little to do with the structure they are meant to find, and fail when the input deviates even slightly from the model. This concern is behind a long tradition of *semirandom* models, in which the input is generated by a combination of random and adversarial choices [BS95, FK01, Fei21]. Aiming for robust algorithms often leads to fundamentally new algorithmic insights – see, for instance, the revolution in high-dimensional algorithmic statistics sparked by the search for robust algorithms [DK23]. Our problem is especially ripe for this kind of development – the only prior efficient algorithm which achieves accuracy $\mathsf{OPT}_{d,\lambda} - \varepsilon$ for every d, λ [MNS16a, YP24] is built on belief propagation, and prior to our work it was not known how to make belief-propagation-based algorithms robust to adversarial corruptions.

Our algorithm: spectral initialization of BP Our algorithm is quite simple; we present an overview in Algorithm 1. The basic idea is to compute the second eigenvector of the *nonbacktracking matrix* of the SBM graph $\boldsymbol{G}$, or approximate this vector via a semidefinite program if given a corrupted graph G', and then use this vector to initialize an algorithm called *belief propagation* (BP). BP aggregates the values that y assigns to edges in a neighborhood of each vertex u into a guess for the label of u. Although our algorithm is simple, the interplay between the spectral initialization

Algorithm 1: BP initialized at the nonbacktracking eigenvector

Input: a graph G' on $[n]$ which differs from an SBM graph $\boldsymbol{G}$ on at most ηn edges and parameter $t \geqslant 1$ depending on the target accuracy.

1. *Compute the second eigenvector.* Let $\vec{E}(\boldsymbol{G})$ be the set of directed edges of $\boldsymbol{G}$, and let $B_{\boldsymbol{G}} \in \{0,1\}^{\vec{E}(\boldsymbol{G}) \times \vec{E}(\boldsymbol{G})}$ be the *nonbacktracking matrix*, which has $(B_{\boldsymbol{G}})_{u\to v, x\to w} = 1$ if $x = v$ and $w \neq u$, and 0 otherwise. If $\eta = 0$, compute a unit eigenvector y of $B_{\boldsymbol{G}}$ for eigenvalue of second-largest modulus. If $\eta > 0$, use a semidefinite program to compute a vector y with correlation $1 - \eta'(\eta)$ to this unit eigenvector.

2. *Round.* Independently for each directed edge e, draw $\tau_e \in \{\pm 1\}$ with $\mathbb{E}\,\tau_e = \tanh\left(\beta\sqrt{|\vec{E}(G')|}\, y_e\right)$. ($\beta$ is a constant depending on d and λ.)

3. *Iterate BP update rule.* Set $m^{(0)}_{u\to v} = \gamma\tau_{v\to u}$ for each directed edge $u \to v$ in G', and for $s = 1, \dots, t$ set $m^{(s)}_{u\to v} = \mathrm{BP}\big(\{m^{(s-1)}_{w\to u} : w \in \partial u \setminus \{v\}\}\big)$, where ∂u is the set of neighbors of u in G' and

$$\mathrm{BP}(\{m_1, \dots, m_k\}) = \frac{\prod_{i\leqslant k}(1+\lambda m_i) - \prod_{i\leqslant k}(1-\lambda m_i)}{\prod_{i\leqslant k}(1+\lambda m_i) + \prod_{i\leqslant k}(1-\lambda m_i)}.$$

   (γ is a constant depending on d, λ, and t.)

4. *Round by signs.* Output $\hat{\sigma}_v = \operatorname{sign}\mathrm{BP}\big(\{m^{(t)}_{w\to v} : w \in \partial v\}\big)$ for each vertex v.

and BP is subtle, and our main technical contribution is an analysis which allows these two tools to be put together.

Belief propagation (BP) is a canonical algorithm for computing marginal probabilities in graphical models, based on a simple dynamic program. Also known as the *sum-product* algorithm, BP is fundamental enough to be commonly taught in undergraduate courses, and yet rigorous correctness analyses for BP on non-tree-like graphical models are very scarce. (BP is easily shown to compute marginal probabilities exactly in tree-structured graphical models.) BP is the only tool we have which can lead to accuracy $\mathsf{OPT}_{d,\lambda}$ in the SBM, but because it is challenging to analyze, prior work [MNS16a] had to build a complicated initialization scheme to make the analysis tractable. (We will discuss this further in Section 2.)

Zdeborová [Zde17] therefore poses the open problem of whether the simple spectral initialization of BP we use here also leads to optimal-error community estimation. We resolve this problem positively in the two-community case. This gives us a nearly-linear-time algorithm and furnishes a rare example of a rigorous analysis of BP on a nontrivial family of non-tree graphs. In the non-corrupted case, our analysis is not too complicated: if $\eta = 0$, the proof that Algorithm 1 gives an optimal estimate of $\sigma^\star$ will be done by the end of Section 4.

1.1 Related work

There is a vast literature on algorithms and information-theoretic limits for the SBM beyond the works we have already discussed – see for instance [CL15, CRV15, ABH16, MNS16b, BMNN16, MMV16, ZZ16, GMZZ17, DAM17, SM19, ABARS20, FC20, ACGP21, LMR+24] and the survey [Abb18]. The Bethe Hessian of [SKZ14] has also been analyzed rigorously: [MRW24] construct its outlier eigenvectors above the Kesten–Stigum threshold, and Stephan and Zhu [SZ25] show that its negative outliers detect the number of communities above the Kesten–Stigum threshold when $d \geqslant 2$. Rigorous analyses of BP on graphs which are not trees are rare (though there are exceptions, e.g. [COMV09]). More broadly, algorithms for statistical estimation in the presence of adversarial contamination have flourished over the last decade; see [DK23]. We highlight a couple of works closely related to ours.

Liu–Moitra robust community detection [LM22] This work studies community detection with a stronger adversary than ours – it can make an unbounded number of *monotone* changes, adding edges inside communities and deleting edges connecting communities (when $\lambda > 0$). The model is also nominally stronger in that ηn nodes can have their adjacency lists modified arbitrarily; however, [DdHS23, Appendix C] show that this node-corruption model reduces to our edge-corruption model.

But, the [LM22] approach requires $d\lambda^2$ to be at least a large constant, and achieves optimal accuracy only in the limit $d\lambda^2 \to \infty$. More quantitatively, in our notation, the accuracy guarantee of [LM22] even in the $\eta = 0$ case is $1 - (1 - \mathsf{OPT}_{d,\lambda})^{1-o_{d\lambda^2\to\infty}(1)}$. By contrast, we obtain overlap $\mathsf{OPT}_{d,\lambda} - \varepsilon$ for every d, λ with $d\lambda^2 > 1$ and every $\varepsilon > 0$. This difference in guarantees corresponds to an important algorithmic difference. The optimal accuracy as $d\lambda^2 \to \infty$ can be obtained by starting with a weak recovery algorithm and greedily flipping vertices whose labels disagree with a majority of their neighbors—this is sort of a "poor man's belief propagation," whereas we need to use actual BP to achieve the $\mathsf{OPT}_{d,\lambda}$ accuracy guarantee.

Robust approximate message passing [IS24, IS25, IS26] A recent line of work by Ivkov and Schramm studies the robustness of approximate message passing (AMP), the dense-matrix analogue of BP, to adversarial corruption of a principal minor of the input matrix. Especially relevant is [IS26], which studies $\mathbb{Z}_2$ synchronization, a dense analogue of the two-community block model. Their algorithm has the same outline as ours: after spectral preprocessing, robustly estimate the top eigenvector to obtain a spectral initialization, then run the message-passing iteration, which turns out to be robust once its initialization is. As is typical when generalizing from dense to sparse random matrix models, however, the analysis in our sparse setting is quite different. Even in the non-corrupted setting, spectral initialization of the message-passing algorithm (in this case AMP) was well understood going back to [MV21]. In the sparse case the analogous result was not known until our work.

Concurrent work In concurrent and independent work, Dong and McKenzie [DM26] show that the outlier eigenvectors of the Bethe Hessian achieve weak recovery down to the Kesten–Stigum threshold, for every $d > 1$. Our analysis of the corrected Bethe Hessian establishes a similar result.

2 Technical overview

We introduce the key players in our algorithm – the nonbacktracking matrix and belief propagation. We then describe how our algorithm and analysis put these components together, and why it leads naturally to an adversarially robust algorithm. The technical sections of our paper divide the implementation of this plan into two parts. In Section 4 we show that BP, initialized at the second eigenvector of the nonbacktracking matrix, succeeds at optimal community estimation. Then in Section 5 we show how to estimate this vector in the presence of adversarial corruptions, with supporting lemmas proved in subsequent sections.

Weak recovery and nonbacktracking walks *Weak recovery* refers to finding $\widehat{\sigma}$ so that

$$\frac{1}{n} \mathop{\mathbb{E}}_{(\sigma^\star, G)\sim \mathrm{SBM}(n,d,\lambda)} |\langle \widehat{\sigma}, \sigma^\star \rangle| \geqslant \Omega(1)\,.$$

This is information-theoretically possible so long as $d\lambda^2 > 1$, the *Kesten–Stigum threshold* [MNS15, Mas14, MNS18].

The algorithm "from the book" for weak recovery in sparse SBM graphs assigns labels based on the second eigenvector of the *nonbacktracking matrix*. For a graph $G = (V, E)$, let $\vec{E}$ be the set of directed edges of G, so $|\vec{E}| = 2|E|$. The nonbacktracking matrix $B \in \mathbb{R}^{\vec{E}\times\vec{E}}$ has entries given by

$$B_{(u,v),(w,z)} = \begin{cases} 1 & \text{if } w = v \text{ and } z \neq u, \\ 0 & \text{otherwise,} \end{cases}$$

The adjacency matrix of a sparse random graph has top eigenvectors dominated by unusually high degree vertices; they are not useful for finding global community structure in the graph. The nonbacktracking matrix fixes this issue. Concretely, [BLM18, Theorem 5] proves that with high probability, taking ξ to be an eigenvector associated to a second-largest-magnitude eigenvalue of B (which w.h.p. has real entries), the algorithm which thresholds the value of $\sum_{u\sim v} \xi_{(u,v)}$ at each vertex v achieves weak recovery. This eigenvector "pokes out" of the spectrum, creating a gap in the eigenvalue spectrum of B. We will return to this second eigenvector ξ later, but for now we move on to the second key ingredient.

Tree reconstruction and BP For our purposes, it is convenient to define BP through the lens of the *broadcast tree reconstruction* problem.

Definition 2.1 (Broadcasting on trees and tree reconstruction)**.** Let $T_{d,t}$ be a d-ary tree with height t.[1] In the broadcast-on-trees process, a uniformly random label $\sigma_r \in \{\pm 1\}$ is assigned to the root vertex r. This label is then propagated down the tree by having each vertex independently choose to copy the label of its parent with probability $(1+\lambda)/2$ and otherwise flip the label, for a parameter $\lambda \in [-1, 1]$. At the end each vertex in the tree has a ± 1 label.

[1] Later we will instead take $T_{d,t}$ to be a Galton-Watson tree, that is, a tree where each vertex has a Poisson-distributed number of children.

The *tree reconstruction problem* is to guess the value σ_r given access to the signs $\sigma_t \in \{\pm 1\}^{d^t}$ of the leaf vertices. In the *noisy tree reconstruction* problem, the goal is to guess the value σ_r given access to σ'_t, which is σ_t with every bit flipped independently with some probability ρ.[2]

The Bayes-optimal algorithm for tree reconstruction is to output $\mathrm{sign}(\mathbb{E}\,\sigma_r \,|\, \sigma_t)$, the sign of the expected value of the root conditioned on the leaves, or $\mathrm{sign}(\mathbb{E}\,\sigma_r \,|\, \sigma'_t)$ in the noisy case. Let $\text{Tree-OPT}_{d,\lambda,\rho} \in [0,1]$ be the limit of the correlation of the root sign with this optimal guess as $t \to \infty$, which captures the optimal probability with which a tree reconstruction algorithm can correctly guess the sign of the root:

$$\text{Tree-OPT}_{d,\lambda,\rho} = \lim_{t\to\infty} \mathbb{E}\Big[\sigma_r \cdot \mathrm{sign}\big(\mathbb{E}[\sigma_r \,|\, \sigma'_t]\big)\Big]$$

where $\sigma'_t = \sigma_t$ when $\rho = 0$.

An important and surprising theorem we will need later is that the optimal probability of correctly guessing the root is actually independent of the noise rate ρ, as long as ρ is not exactly $1/2$ (which would destroy all the signal about σ_r).

Theorem 2.2 ([MNS16a, YP24]). *If $d > 1$, $\lambda \in (-1,1)$, and $\rho \neq 1/2$, then Tree-OPT$_{d,\lambda,\rho}$ = Tree-OPT$_{d,\lambda,0}$.*

BP is a dynamic program to compute the posterior expected value $\mathbb{E}[\sigma_r \,|\, \sigma_t]$ or $\mathbb{E}[\sigma_r \,|\, \sigma'_t]$. For concreteness we have described it explicitly in Algorithm 1, but actually all we will need to use about BP is that it computes this posterior expected value.

Algorithms for estimating $\boldsymbol{\sigma}^\star$ in the SBM with optimal error use BP as a subroutine, feeding it inputs which are not actually derived from a broadcast-on-tree process, but rather computed in some way from an SBM graph. The key challenge is to analyze BP initialized with these "off distribution" inputs, by relating those inputs back to a tree process.

The basic relationship between the SBM and broadcasting on trees, pioneered in [MNS15], stems from taking a "local view" of the SBM. For a vertex u, let $B_t(u)$ be a ball of radius t in $\boldsymbol{G}$ and look at the joint distribution $(\boldsymbol{\sigma}^\star_{B_t(u)}, B_t(u))$, where $\boldsymbol{\sigma}^\star_{B_t(u)}$ is the restriction of $\boldsymbol{\sigma}^\star$ to coordinates in $B_t(u)$. It turns out that this distribution is very close to the broadcast-on-trees distribution.

Lemma 2.3 (Simplified version of Fact 4.7, [BLM18, Proposition 36]). *Let f be a function mapping depth-t trees whose vertices have ± 1 labels to $[-1,1]$, let $d > 0$, and let $T_{d,t}$ be a depth-t Galton-Watson tree (where every vertex has $\mathrm{Poi}(d)$ children). Then*

$$\frac{1}{n}\sum_{u\in[n]} f\big(B_t(u), \boldsymbol{\sigma}^\star_{B_t(u)}\big) \longrightarrow \mathbb{E}\, f(T_{d,t}, \sigma)$$

in probability as $n \to \infty$, where σ is the broadcast labeling of $T_{d,t}$ with parameter λ, and $B_t(u)$ is rooted at u. (With high probability $B_t(u)$ is a tree for all but $o(n)$ vertices u, so it does not matter how f is defined on the others.)

[2] Noisy tree reconstruction is often called *robust* tree reconstruction, but in this section we reserve the word "robust" in this paper to refer to robustness against adversarial corruptions.

A related fact, proved by [MNS16a, Theorem 2.8], is that Tree-OPT$_{d,\lambda} \geqslant$ OPT$_{d,\lambda}$, because even an estimator for $\sigma_u^\star$ which is also given the true labels of all vertices at distance exactly t from u can do no better than tree reconstruction — conditioned on those labels, the rest of the graph carries only vanishing information about $\sigma_u^\star$, and by Lemma 2.3 the ball $B_t(u)$ together with its labels looks like a depth-t broadcast tree.

Idealized model for combining weak recovery and BP We are building towards an algorithm for estimating $\sigma^\star$ which combines BP and nonbacktracking walks, and achieves correlation Tree-OPT$_{d,\lambda}$, the best possible in light of Tree-OPT$_{d,\lambda} \geqslant$ OPT$_{d,\lambda}$. The basic idea is to initialize BP using the entries of the second eigenvector ξ of the nonbacktracking matrix. At this point we have all the ingredients to describe a useful idealized model to explain why this kind of approach should work.

Imagine that in addition to $\boldsymbol{G}$, our algorithm had access to an idealized weak recovery σ', which is $\sigma^\star$ with every entry flipped independently with probability $\rho < 1/2$ (this is the block model with side information studied in [MX16]). The labeling σ' is itself a weak recovery of $\sigma^\star$ because $\mathbb{E}\,|\langle \sigma^\star, \sigma' \rangle| \geqslant \Omega(n)$.

We could then "boost" the correlation of σ' with $\sigma^\star$ by applying BP in a neighborhood of each vertex – that is, for each vertex u, we could take $\widehat{\sigma}_u$ to be the sign of the output of BP initialized with $\sigma'_{B_t(u)}$, the noisy signs of vertices in a depth-t neighborhood (see also [KMS16] on combining graph and node information). Lemma 2.3 and Theorem 2.2 together show that for this choice of $\widehat{\sigma}$, $\frac{1}{n}\,\mathbb{E}\,|\langle \widehat{\sigma}, \sigma^\star \rangle| \to$ OPT$_{d,\lambda}$.

The main challenge: the need for independence in the input to BP Departing from the idealized model, of course in reality we don't get access to σ' which is an independent noisy copy of the ground truth $\sigma^\star$. We would like to simulate such nice σ' to generate inputs to BP as before, but using a weak recovery algorithm we can actually run.

The major difficulty is that independence of the noise added to $\sigma^\star$ to produce σ' in the idealized model is crucial to being able to apply Theorem 2.2. Indeed, no analog of Theorem 2.2 is possible if independent random flips of the leaf labels are replaced by adversarial flips – the output of BP becomes useless in the presence of a vanishing fraction of adversarial bit-flips at the leaves (this is folklore; see, e.g., [HL24]). But the locations where the output σ_{weak} of a weak recovery algorithm differs from $\sigma^\star$, which become the leaf-flip locations when we feed σ_{weak} to BP, will be correlated in an inscrutable way.

The prior work [MNS16a] tackles this issue by wrapping a weak recovery algorithm in a graph-splitting procedure which creates enough of the necessary independence to simulate σ' by "repurposing" some of the independent randomness in the edges of the SBM graph. (This is sometimes called the "cavity method.")

Unlike [MNS16a], we are looking for an algorithm that works in the presence of adversarial corruptions to the edge structure of the graph. A first idea is to use the graph-splitting approach in conjunction with a robust weak recovery algorithm like [DdNS21, MRW24] which still achieves weak recovery even with a small constant fraction of adversarial edge corruptions. The problem is that the adversarial corruptions destroy the independence in the edges of the SBM graph which the graph-splitting approach relies on. Thus, the edge-corrupting adversary might be able to arrange

their corruptions so that the output σ_{weak} still has $|\langle \sigma_{\text{weak}}, \sigma^\star \rangle| \geqslant \Omega(n)$, but for most vertices u, the locations among $B_t(u)$ where σ_{weak} and $\sigma^\star$ disagree have a very non-random structure, which fools BP. We need a different approach.

Key idea: local characterization of the second eigenvector Let us return to the setting without adversarial corruptions to the graph, to design an algorithm for optimal-error community estimation without relying on graph splitting. Our key insight is that a simple rounding scheme applied to the second eigenvector of the nonbacktracking matrix produces a sign vector which acts enough like the idealized σ' that we can use it directly to initialize BP. In particular, we show that the locations of entries where $\sigma^\star$ and the rounded vector differ are "random enough" – this makes it unlike a generic weak recovery algorithm, where we wouldn't have control over the locations of these disagreements, only over their number.

The crucial technical ingredient is a new *local* characterization of the second eigenvector of the nonbacktracking matrix. Here "local" means that we show that the (u, v) entry of the second eigenvector is essentially a function of a constant-radius local neighborhood of (u, v) in $\boldsymbol{G}$, together with the labels $\sigma^\star$ in that local neighborhood.

Lemma 2.4 (Local characterization of the second eigenvector, informal version of Lemma 4.3)**.** *Let $d\lambda^2 > 1$. With high probability, the vector $W^{(\ell)} \in \mathbb{R}^{\vec{E}}$ given by aggregating $\sigma^\star$ over depth-ℓ nonbacktracking walks*

$$W^{(\ell)}_{(u,v)} = \sum_{(u_0,\ldots,u_{\ell+1})} \sigma^\star_{u_{\ell+1}},$$

where the sum is over nonbacktracking walks with $u_0 = u$ and $u_1 = v$, has correlation $1 - o_{\ell\to\infty}(1)$ with the eigenvector of $B_{\boldsymbol{G}}$ with eigenvalue of second-largest modulus.

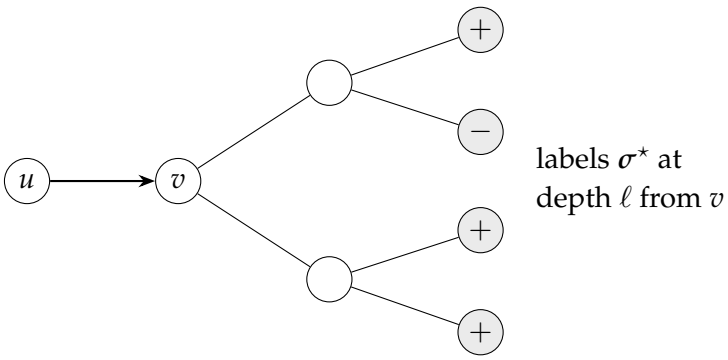


Figure 2: The walk score $W^{(\ell)}_{(u,v)}$, drawn for $\ell = 2$. The nonbacktracking walks starting with the directed edge (u, v) form a tree grown outwards from v, which does not include u, and $W^{(\ell)}_{(u,v)}$ sums the labels $\sigma^\star$ at its leaves.

Intuition for Lemma 2.4 Because we are working with a sparse random graph and the non-Hermitian matrix B, proving Lemma 2.4 requires some care; for instance we cannot rely on an orthonormal basis of eigenvectors. We are able to prove it by combining tools from [BLM18],

especially Lemma 2.3. But the intuition for why it ought to be true is simple. Suppose we pretend that B is a symmetric matrix with an orthonormal basis of eigenvectors such that:

- B has constant size gaps in the spectrum: $|\lambda_1(B)| \geqslant |\lambda_2(B)| + \Omega(1) \cdot |\lambda_1(B)|$ and $|\lambda_2(B)| \geqslant |\lambda_3(B)| + \Omega(1) \cdot |\lambda_1(B)|$, where the eigenvalues are ordered by magnitude, and
- the vector $\text{lift}(\sigma^\star) \in \mathbb{R}^{\vec{E}}$ placing $\sigma^\star_w$ on each directed edge (x, w), has $\Omega(1)$ correlation with the second eigenvector, and is orthogonal to the top eigenvector.

This is roughly the picture established by [BLM18, Theorem 4]. Then if we do $O(1)$ steps of power iteration initialized at $\text{lift}(\sigma^\star)$, applying the spectral theorem we will arrive at a vector with correlation 0.99 to the second eigenvector. The walk score vector $W^{(\ell)}$ is exactly the result of ℓ steps of power iteration initialized at $\text{lift}(\sigma^\star)$.

The heart of the argument – consequences of Lemma 2.4 for spectral initialization of BP The local view of the second eigenvector means that we can pretend that our algorithm has direct access to the walk score vector $W^{(\ell)}$, for large constant ℓ. How can we use this vector to initialize BP in a way which simulates the idealized argument using σ'?

Consider using BP to infer the label of vertex u by initializing BP at each leaf w of $B_t(u)$ with the sign of $W^{(\ell)}_{(\text{parent}(w),w)}$, where $\text{parent}(w)$ is the parent of w in the tree $B_t(u)$. The key idea is: when running BP on this depth-t tree, the input $\{\text{sign}(W^{(\ell)}_{(\text{parent}(w),w)})\}_{w \in \text{leaves}(B_t(u))}$ essentially acts like $\sigma'_{\text{leaves}(B_t(u))}$!

To see why, note that $W^{(\ell)}_{(\text{parent}(w),w)}$ only depends on the labels $\sigma^\star$ at distance ℓ below w – that is, on the depth-ℓ subtree hanging off w, grown away from u (see Figure 2). So everything involved lives in $B_{t+\ell}(u)$, and by Lemma 2.3 we may pretend that this ball is a depth-$(t+\ell)$ broadcast tree. In the broadcast process, once we condition on $\sigma^\star_w$, the labels in the subtree below w are independent of $\sigma^\star_u$ and of the subtrees below all the other leaves, which do not overlap because of nonbacktracking.

All together, conditioned on the labels of the leaves of $B_t(u)$, the signs $\{\text{sign}(W^{(\ell)}_{(\text{parent}(w),w)})\}_{w \in \text{leaves}(B_t(u))}$ are independent, and by symmetry each one equals $\sigma^\star_w$ with the same probability $1 - \rho$. This is exactly the noisy tree reconstruction problem with noise rate ρ. And as long as $\rho < 1/2$, Theorem 2.2 says that as $t \to \infty$, BP from these inputs is just as good as BP from the true labels of the leaves. We just need to analyze the noise rate ρ, which is not so hard because the process which starts at a leaf w and produces $W^{(\ell)}_{(\text{parent}(w),w)}$ is itself a broadcast process on a depth-ℓ tree. We can show that ρ stays bounded away from $1/2$ as $\ell \to \infty$ precisely when $d\lambda^2 > 1$ [KS66, EKPS00, MP03, MNS15]. This completes the non-robust version of our argument. The formal version is captured by Theorem 4.2.

Robustness of BP and the second-eigenvector initialization Now, what if instead of $\boldsymbol{G}$, we get as input G' for which $|E(\boldsymbol{G}) \triangle E(G')| \leqslant \eta n$? Our goal is to estimate $\sigma^\star$ with accuracy $\mathsf{OPT}_{d,\lambda} - \varepsilon$ while allowing $\eta(d, \lambda, \varepsilon) > 0$.

A first observation is that for every t, only around $\eta d^t n$ vertices are distance $\leqslant t$ to an adversarially corrupted edge in G'. So if we initialize depth-t BP with the same vector in $\boldsymbol{G}$ and G', at all but $\eta d^t n$ vertices we will get the same output. Furthermore, if the initializations were to differ at

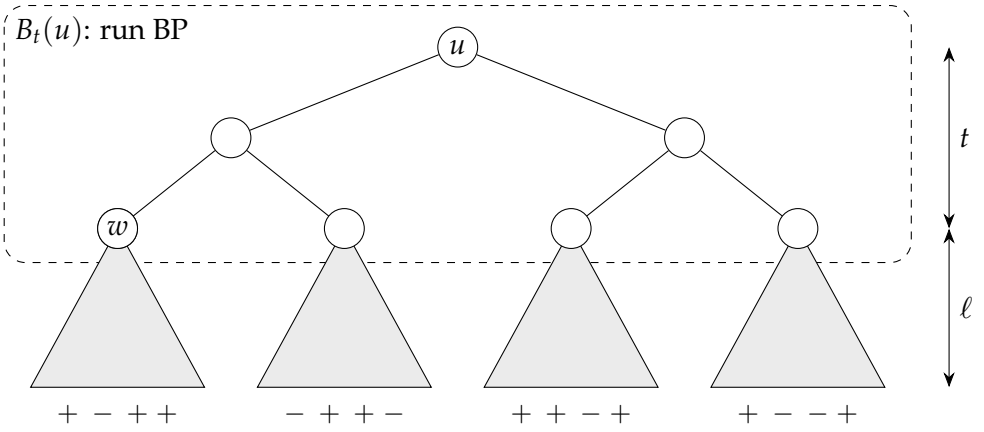


Figure 3: The two-level ball $B_{t+\ell}(u)$, drawn for $t = 2$. BP runs on the inner ball $B_t(u)$ (dashed). Each leaf w of $B_t(u)$ is initialized with $\operatorname{sign}(W^{(\ell)}_{(\text{parent}(w),w)})$, which depends only on the labels $\sigma^\star$ at the bottom of the depth-ℓ subtree hanging off w (shaded). These subtrees are disjoint, so conditioned on the labels of the leaves of $B_t(u)$, the inputs to BP are independent.

some δn locations, it would introduce only an additional $\delta d^t n$ disagreements in the outputs. We can tolerate this so long as $\delta, \eta \ll \varepsilon d^{-t}$, which we can in turn ensure so long as t is constant with respect to n, which it always is in the above arguments.

Thus, to robustify the whole algorithm, all we need to do is show that given G' we can find a vector y which has correlation $1 - \delta^{O(1)}$ with the second eigenvector of B_G (and hence also has correlation close to 1 with the walk score vector $W^{(\ell)}$). This suffices, because our procedure to round the second eigenvector to the signs we use to initialize BP produces sign vectors which differ on only δn locations if given inputs which are $1 - \delta^{O(1)}$ correlated. (Ignoring dependence on d, which is constant anyway.)

As a first attempt, we could try directly using the second eigenvector of $B_{G'}$. But it is well known that a small adversarial corruption of the graph can dramatically shift the eigenvectors of $B_{G'}$ (this argument is typically phrased for the adjacency matrix, but the phenomenon carries over to the nonbacktracking matrix).

Robustly estimating the second eigenvector We design a new algorithm to obtain an estimate of the second eigenvector of B_G, captured by Theorem 5.1. For any $\delta > 0$ there is $\eta > 0$ such that our algorithm can find a vector whose correlation to the second eigenvector of B_G is $1 - \delta$, given a graph G' with ηn edge corruptions.

Before describing our algorithm, we discuss the basic intuition for why enough information is still present after ηn edge corruptions to approximately recover the second eigenvector of B_G. Let's briefly adopt the following simple toy model for how B_G behaves.

Toy model and robust PCA Imagine that $M \in \mathbb{R}^{m \times m}$ is a *symmetric* matrix with a spectral gap, having $\lambda_1(M) = 1$ and all other eigenvalues of M at most $1/2$ in magnitude, and imagine further that the top eigenvector v is delocalized; say $\|v\|_2 = 1$ and the maximum magnitude entry of v satisfies $\|v\|_\infty \leqslant C/\sqrt{m}$. M is our stand-in for B_G and v is our stand-in for the second eigenvector

of B_G. When the adversary modifies ηn edges of the graph $\boldsymbol{G}$, the matrix B_G can change by at most 1 in around ηn locations; in our toy model we can imagine we get access to a matrix $\widetilde{M}$ for which $M - \widetilde{M}$ is small in the entrywise ℓ_1 norm $\|\cdot\|_{\mathrm{ent},1}$.

It is well known that given access to such $\widetilde{M}$, it is possible to find a good approximation to v, and this can be done computationally efficiently [GV16, MS16]; see also the robust PCA literature [CLMW11, CSPW11]. This can be done by solving a semidefinite programming (SDP) relaxation of the optimization problem

$$\max \langle \widetilde{M}, xx^\top \rangle \text{ s.t. } \|x\|_2 = 1\,,\ \|x\|_\infty \leqslant C/\sqrt{m}\,.$$

The SDP is closely related to the cut-norm/Grothendieck SDP. The important feature of the analysis is that for any feasible solution, $\langle M - \widetilde{M}, xx^\top \rangle \leqslant \|x\|_\infty^2 \|M - \widetilde{M}\|_{\mathrm{ent},1}$ by Hölder's inequality, and this is preserved by the SDP relaxation. The analysis relies on the spectral gap of M and the delocalization of the top eigenvector. Similar approaches were used in the context of the SBM in e.g. [GV16].

From the toy model back to the nonbacktracking matrix Our toy model fails to reflect the actual behavior of the nonbacktracking matrix B_G in a few key ways, which our algorithm is designed to work around. First, B_G is not symmetric – we could still try to use the above approach, but it would need a gap in the *singular value* spectrum, which does not happen for B_G. Second, we are interested in the second eigenvector of B_G, not the top one – we will have to do something about the top eigenvalue so we can access the second one. Third, the eigenvector we want to recover doesn't actually have ℓ_∞ norm $O(1/\sqrt{\dim(B_G)})$. We can see why by recalling the local characterization of the second eigenvector (Lemma 2.4), a few unusually dense neighborhoods in the graph will contain a lot of nonbacktracking walks, driving up the magnitude of a few entries. We address each of these issues in turn.

Passing to a symmetric matrix – the corrected Bethe Hessian There is by now an established trick to obtain a symmetric matrix with a spectral gap whose eigenvectors can be transformed into good approximations to the eigenvectors of B_G. This matrix is called the *Bethe Hessian* [SKZ14]. It is actually a family of matrices, indexed by a real-valued parameter t

$$H(t) = I - tA + t^2(D - I)\,,$$

where A is the adjacency matrix and D is the diagonal matrix of degrees. This careful balancing of degree and adjacency matrices, for the right choices of t, regularizes the spectrum of A to get rid of noisy localized eigenvectors, so that the community structure of G can emerge.

More concretely, a technique pioneered by [MRW24] shows that by choosing t appropriately, when the underlying graph is $\boldsymbol{G} \sim \mathrm{SBM}(n, d, \lambda)$, the matrix $H(t)$ has at most two negative eigenvalues, and the associated eigenvectors correspond roughly to the top and second eigenvectors of the nonbacktracking matrix. Furthermore, $H(t)$ has spectral gaps and it cannot change much in entrywise ℓ_1 norm when the adversary edits ηn edges, meaning we can hope to robustly recover those eigenvectors. (Here "correspond" means that if we get access to the $H(t)$ eigenvectors we can in linear time construct good estimates of the corresponding eigenvectors of the nonbacktracking matrix, see Lemma 5.3.)

Because we want to estimate the second eigenvector of B_G to very high accuracy, we need the correspondence between $H(t)$'s eigenvectors and B_G's to be quantitatively stronger than in prior work [MRW24]. To achieve this, we make two technical refinements on top of the ideas from [MRW24]. First, our algorithm computes a careful explicit rank-one correction to $H(t)$ to excise an uninformative eigenvector (the analog of the Perron-Frobenius eigenvector). Second, to even better align the eigenvectors of $H(t)$ with those of B_G we actually work with $A^{(\ell)}H(t)A^{(\ell)}$, where $A^{(\ell)}$ counts nonbacktracking walks of large constant length. Importantly, these changes preserve the robustness to edge corruptions in the graph, if we first preprocess the graph by removing very high degree vertices, another standard trick [FO05]. The details are in Definition 5.5 and Lemmas 5.6 and 5.8.

Tolerating approximate delocalization We turn to the last technical issue making our toy model differ from reality – large entries of the second eigenvector. (The solution would apply to B_G or $H(t)$ or our modified version thereof so we don't distinguish carefully among them now.) It turns out that the eigenvector we want to find still has a good enough delocalization property – its 4-norm is small. Concretely, if we normalize so $\|v\|_2 = 1$, we are able to show that $\|v\|_4^4 \leqslant O(1/n)$, with high probability over $\boldsymbol{G}$. We adapt the standard robust PCA-style arguments to show that this ℓ_4-style delocalization is still enough to approximately recover v using an SDP relaxation of

$$\max\langle -A^{(\ell)}(H(t) - (\text{rank-one correction}))A^{(\ell)}, xx^\top\rangle \text{ s.t. } \|x\|_2 = 1\,,\ \|x\|_4^4 \leqslant O(1/n)\,,$$

even in the presence of adversarial edge corruptions. The argument uses that $A^{(\ell)}(H(t) - \text{rank-one correction})A^{(\ell)}$ is stable under edge corruptions in a slightly stronger norm than entrywise ℓ_1. Finally, we also show that we can solve this SDP in nearly-linear time via matrix multiplicative weights [AK07].

The details are in Lemmas 5.8 and 5.9.

3 Preliminaries

3.1 Stochastic block model and community detection

Fix $d > 0$ and $\lambda \in (-1, 1)$. For all sufficiently large n, in the symmetric two-community stochastic block model $\mathrm{SBM}(n, d, \lambda)$, the hidden labels $\boldsymbol{\sigma}^\star \in \{\pm 1\}^n$ are independent and uniform. Conditional on the labels, each unordered pair of distinct vertices is joined independently with probability

$$\frac{d}{n}\big(1 + \lambda \boldsymbol{\sigma}_u^\star \boldsymbol{\sigma}_v^\star\big).$$

We work above the Kesten–Stigum threshold, namely $d\lambda^2 > 1$. The problem of estimating $\boldsymbol{\sigma}^\star$ from the graph G is called *community detection*. The performance of an estimator $\widehat{\boldsymbol{\sigma}} \in \{\pm 1\}^n$ for community detection is measured by

$$\mathrm{ov}(\widehat{\boldsymbol{\sigma}}, \boldsymbol{\sigma}^\star) = \frac{1}{n}\,|\langle \widehat{\boldsymbol{\sigma}}, \boldsymbol{\sigma}^\star\rangle|\,.$$

Let $\mathsf{OPT}_{d,\lambda}$ denote the information-theoretically optimal asymptotic overlap. Section 3.3 recalls its equivalent characterization by reconstruction on a Poisson broadcast tree.

As a convention, random objects are written in boldface and their realizations are plaintext.

3.2 Nonbacktracking matrix

For a graph G, let ∂v denote the set of neighbors of a vertex v, let $\vec{E}(G)$ be its set of directed edges, containing both orientations $u \to v$ and $v \to u$ of every edge $\{u, v\}$. The *nonbacktracking matrix* [Has89] $B_G \in \{0,1\}^{\vec{E}(G)\times\vec{E}(G)}$ has entries

$$(B_G)_{u\to v,\, x\to w} = \mathbf{1}[x = v]\,\mathbf{1}[w \neq u],$$

so a directed edge $u \to v$ may be followed by any edge leaving v other than its reversal $v \to u$. Equivalently, for $z \in \mathbb{R}^{\vec{E}(G)}$,

$$(B_G z)_{u\to v} = \sum_{w\in\partial v\setminus\{u\}} z_{v\to w}.$$

3.3 Broadcast trees, belief propagation, and optimal recovery

The optimal overlap $\mathsf{OPT}_{d,\lambda}$ achieved by any community-detection algorithm is characterized by *reconstruction on a Poisson broadcast tree* [MNS16a, Theorem 2.8], [YP24, Theorem 38].

Definition 3.1 (Broadcast process on Galton–Watson tree)**.** Fix $d > 0$ and $\lambda \in (-1, 1)$. Let $\boldsymbol{T}_d$ be the rooted Galton–Watson tree [AN72] in which every vertex has an independent $\mathrm{Poi}(d)$ number of children. Give its root o a uniformly random spin $\boldsymbol{x}_o \in \{\pm 1\}$, and let each child independently copy its parent's spin with probability $(1+\lambda)/2$. Write $(\boldsymbol{T}_{d,\lambda}, \boldsymbol{x})$ for the resulting tree together with the vector $\boldsymbol{x}$ of spins.

Let $\boldsymbol{x}_t$ consist of the spins at the depth-t vertices. The depth-t root posterior mean conditioned on the tree itself and the depth-t spins is

$$\boldsymbol{M}_t = \mathbb{E}[\boldsymbol{x}_o \mid (\boldsymbol{T}, \boldsymbol{x}_t)].$$

The problem of computing $\boldsymbol{x}_o$ given the tree $\boldsymbol{T}$ and leaf spins $\boldsymbol{x}_t$ is called *broadcast tree reconstruction*. The canonical algorithm for broadcast tree reconstruction is *belief propagation*.

Definition 3.2 (Belief propagation)**.** The *belief propagation* (BP) algorithm on a depth-t tree with $+1/-1$ labels x_u on its leaves u is defined as follows. Initialize every directed child-to-parent edge $u \to v$ at depth t with a *message* $m_{u\to v} = x_u$. Moving upward, after u receives the messages from its children, it sends its parent v

$$m_{u\to v} = \mathrm{BP}(\{m_{w\to u} : w \in \mathrm{ch}(u)\}) := \frac{\prod_{w\in\mathrm{ch}(u)}(1+\lambda m_{w\to u}) - \prod_{w\in\mathrm{ch}(u)}(1-\lambda m_{w\to u})}{\prod_{w\in\mathrm{ch}(u)}(1+\lambda m_{w\to u}) + \prod_{w\in\mathrm{ch}(u)}(1-\lambda m_{w\to u})}, \tag{1}$$

where $\mathrm{ch}(u)$ is the set of children of u. Apply the same rule at the root to obtain m_o. The denominator is positive because $|\lambda| < 1$ and every message lies in $[-1, 1]$.

The following fact is standard.

Fact 3.3 (Exactness of tree BP)**.** *For every initialization x_t on the vertices at depth t, m_o is the exact posterior mean of the root. Equivalently, $\boldsymbol{m}_o = \boldsymbol{M}_t$.*

Robust tree reconstruction and Bayes-optimal algorithms. As pioneered by Mossel, Neeman, and Sly [MNS16a], the algorithmic question of optimal recovery in block models is closely related to the problem of broadcast tree reconstruction after additional independent noise is applied to the leaves. To be consistent with the literature, we call this the *robust tree reconstruction* problem [JM04]. However, importantly, this problem does not involve *adversarial* corruption of the observations in any way, making it quite distinct from the robust block model problem studied in this paper. But, we will still make extensive use of this random-leaf-noise version of broadcast tree reconstruction.

Definition 3.4 (Robust tree reconstruction)**.** Run the broadcast process through generation t. Independently at every boundary vertex v, retain $\boldsymbol{x}_v$ with probability $\gamma \in (0,1]$ and otherwise replace it by an independent uniform spin; denote the resulting noisy spin by $\boldsymbol{\tau}_v$, and let $\boldsymbol{\tau}_t$ consist of all these noisy spins at the depth-t vertices. The root posterior mean conditioned on the noisy spins is

$$\boldsymbol{M}_t^{(\gamma)} = \mathbb{E}[\boldsymbol{x}_o \mid (\boldsymbol{T}, \boldsymbol{\tau}_t)].$$

Robust reconstruction asks how well the root can be recovered as $t \to \infty$.

Remark 3.5 (BP for robust reconstruction)**.** Belief propagation also computes $\boldsymbol{M}_t^{(\gamma)}$ exactly; only the initialization changes. In Definition 3.2, the message $m_{u\to v}$ that a vertex u sends its parent v is the posterior mean of x_u given the observations in the subtree below u, under a uniform prior on x_u. For a depth-t vertex u, the only observation is τ_u, which equals x_u with probability $(1+\gamma)/2$; hence $\mathbb{E}[x_u \mid \tau_u] = \gamma\tau_u$. So we initialize $m_{u\to v} = \gamma\tau_u$ at every depth-t vertex u, in place of $m_{u\to v} = x_u$, and run the same update (1). The argument of Fact 3.3 then gives $m_o = \boldsymbol{M}_t^{(\gamma)}$.

Given the tree, $\boldsymbol{x}_{t+1}$ is obtained from $\boldsymbol{x}_t$ by one more step of broadcasting, so $\mathbb{E}\,|\boldsymbol{M}_{t+1}| \leqslant \mathbb{E}\,|\boldsymbol{M}_t|$. Define

$$\mathsf{OPT}_{d,\lambda} = \lim_{t\to\infty} \mathbb{E}\,|\boldsymbol{M}_t|.$$

We justify momentarily the collision of notation between the tree reconstruction problem and the previously-defined optimal overlap for the stochastic block model.

Following Mossel, Neeman, and Sly [MNS16a], who proved this when $d\lambda^2$ is sufficiently large, Yu and Polyanskiy showed that the noise in robust reconstruction does not change this limit.[3]

Fact 3.6 (Robust reconstruction [YP24, Theorem 34 and Proposition 48(7)])**.** *For every $d > 0$, $\lambda \in (-1,1)$, and $\gamma \in (0,1]$,*

$$\lim_{t\to\infty} \mathbb{E}\,|\boldsymbol{M}_t^{(\gamma)}| = \mathsf{OPT}_{d,\lambda}. \tag{2}$$

Further, for any $t \geqslant 0$ and $0 \leqslant \gamma' \leqslant \gamma \leqslant 1$, we have $\mathbb{E}\left|\boldsymbol{M}_t^{(\gamma)}\right| \geqslant \mathbb{E}\left|\boldsymbol{M}_t^{(\gamma')}\right|$.

The following theorem relates the (robust) broadcast tree reconstruction problem to the problem of recovering the community labels in the stochastic block model.

[3] Yu and Polyanskiy work with log-likelihood ratios. In their notation, the tree has offspring law $\mathrm{Poi}(d)$ and edge channels $\mathrm{BSC}_{(1-\lambda)/2}$, and the noisy leaves are observed through $\mathrm{BSC}_{(1-\gamma)/2}$, which is nontrivial since $\gamma > 0$. Their Theorem 34 says that the root log-likelihood ratio $\boldsymbol{R}_t$ has the same limiting law as $t \to \infty$ with noisy leaves as with exact leaves. Since $|\boldsymbol{M}_t| = \tanh(|\boldsymbol{R}_t|/2)$ is a bounded continuous function of $\boldsymbol{R}_t$, (2) follows [YP24, Remark 8].

Theorem 3.7 (Tree characterization of optimal recovery [MNS16a, YP24]). *For $(\boldsymbol{G}, \boldsymbol{\sigma}^\star)$ distributed according to the stochastic block model with parameters d, λ, and every estimator $\widehat{\boldsymbol{\sigma}}(\boldsymbol{G})$, and every $\varepsilon > 0$,*

$$\boldsymbol{Pr}\big(\mathrm{ov}(\widehat{\boldsymbol{\sigma}}, \boldsymbol{\sigma}^\star) > \mathsf{OPT}_{d,\lambda} + \varepsilon\big) \longrightarrow 0.$$

Conversely, for every $\varepsilon > 0$ there is an estimator, computable in time $n^{1+o(1)}$, whose overlap is at least $\mathsf{OPT}_{d,\lambda} - \varepsilon$ with high probability.

The fact that no estimator in the block model setting performs better than tree reconstruction is Mossel, Neeman, and Sly [MNS16a, Theorem 2.8]. Yu and Polyanskiy showed there is an estimator for the communities in the block model which obtains overlap $\mathsf{OPT}_{d,\lambda}$, for every choice of d, λ, by improving the analysis of the algorithm of Mossel, Neeman, and Sly [YP24, Theorem 38].

4 Initialization of belief propagation at the second eigenvector of the nonbacktracking matrix

In this section we show that belief propagation, initialized from the second eigenvector of the nonbacktracking matrix, gives an optimal recovery algorithm in the stochastic block model. We also show that this is robust in two senses:

1. It suffices to initialize BP at any vector close to this eigenvector, and
2. Running BP on a corrupted graph (with a small constant fraction of adversarial edge corruptions) also produces a near-optimal solution, with a loss that depends on the fraction of corrupted edges.

We will need to compare vectors indexed by the directed edges of different graphs on $[n]$. To make this meaningful, we view $\mathbb{R}^{\vec{E}(G)}$, for every graph G on $[n]$, as the subspace of $\mathbb{R}^{n^2-n}$ consisting of the vectors indexed by ordered pairs of distinct vertices that vanish off $\vec{E}(G)$. Inner products and distances between vectors on the directed edges of different graphs are taken in $\mathbb{R}^{n^2-n}$, extending vectors in $\mathbb{R}^{\vec{E}(G)}$ by padding with zeros as necessary.

Definition 4.1 (BP from an initialization)**.** Given a graph G, an initialization $Y \in \mathbb{R}^{\vec{E}(G)}$, an integer $t \geqslant 1$, and $\gamma \in (0,1)$, the procedure $\mathsf{BP}_{t,\gamma}(G, Y)$ does the following.

1. Independently for each directed edge e, draw $\boldsymbol{\tau}_e \in \{\pm 1\}$ with $\mathbb{E}\,\boldsymbol{\tau}_e = \tanh Y_e$.
2. Set $m^{(0)}_{u\to v} = \gamma \boldsymbol{\tau}_{v\to u}$ and, for $s = 1, \dots, t$, $m^{(s)}_{u\to v} = \mathrm{BP}\big(\{m^{(s-1)}_{w\to u} : w \in \partial u \setminus \{v\}\}\big)$.
3. Output $\widehat{\boldsymbol{\sigma}}_v = \mathrm{sign}\big(\mathrm{BP}(\{m^{(t)}_{w\to v} : w \in \partial v\})\big)$.

The first, randomized step is a technical convenience that makes it easier to apply the results of [YP24]; we expect that with a more delicate analysis a deterministic initialization procedure mapping Y_e to initial messages would also work.

Our main theorem for this section is:

Theorem 4.2 (BP from the second eigenvector is optimal, robustly). *Let $d\lambda^2 > 1$ and $(\boldsymbol{G}, \boldsymbol{\sigma}^\star) \sim \mathrm{SBM}(n, d, \lambda)$, and let $\boldsymbol{\xi}$ be a unit eigenvector for the eigenvalue of $B_{\boldsymbol{G}}$ of second-largest modulus. For every $\varepsilon > 0$ there are $t \geqslant 1$, $\gamma \in (0,1)$, $\beta > 0$, and $\eta > 0$ such that the following holds with high probability. For every graph G' on $[n]$ with $|E(\boldsymbol{G}) \triangle E(G')| \leqslant \eta n$ and every unit $y \in \mathbb{R}^{\vec{E}(G')}$ such that $\langle y, \boldsymbol{\xi}\rangle^2 \geqslant 1 - \eta$, the output $\widehat{\boldsymbol{\sigma}}$ of $\mathsf{BP}_{t,\gamma}\big(G', \beta\sqrt{|\vec{E}(G')|}\, y\big)$ satisfies*

$$\mathbb{E}\ \mathrm{ov}(\widehat{\boldsymbol{\sigma}}, \boldsymbol{\sigma}^\star) \geqslant \mathsf{OPT}_{d,\lambda} - \varepsilon,$$

where the expectation is over the random rounding in $\mathsf{BP}_{t,\gamma}$, and the procedure uses $O_{d,\lambda,\varepsilon}(n)$ arithmetic operations.

We state the main lemmas from which Theorem 4.2 follows and then put them together to prove Theorem 4.2.

The most important lemma establishes a *local* characterization of the second eigenvector of the nonbacktracking matrix – it shows that the (u, v) entry of the second eigenvector is essentially a function of the ground-truth labels $\boldsymbol{\sigma}^\star$ in a constant-radius neighborhood of the edge (u, v).

Setting up some notation, for a blockmodel graph $\boldsymbol{G}$ with ground-truth labeling $\boldsymbol{\sigma}^\star$ and for $\ell \geqslant 0$, the *walk score* $W^{(\ell)} \in \mathbb{R}^{\vec{E}(\boldsymbol{G})}$ is

$$W^{(\ell)}_{u\to v} = (d\lambda)^{-\ell} \sum_{(u_0,\dots,u_{\ell+1})} \boldsymbol{\sigma}^\star_{u_{\ell+1}},$$

where the sum is over nonbacktracking walks in $\boldsymbol{G}$ with $u_0 = u$, $u_1 = v$, and $u_{i+1} \neq u_{i-1}$ for $1 \leqslant i \leqslant \ell$. The normalization by $(d\lambda)^{-\ell}$ is chosen so that when translated to the broadcast tree, $W^{(\ell)}$ is an unbiased estimator of $\boldsymbol{\sigma}^\star_v$.

Lemma 4.3 (The eigenvector is close to a walk score). *Let $d\lambda^2 > 1$, let $(\boldsymbol{G}, \boldsymbol{\sigma}^\star) \sim \mathrm{SBM}(n, d, \lambda)$, and let $\boldsymbol{\xi}$ be as in Theorem 4.2. For every fixed $\ell \geqslant 0$, $\langle \boldsymbol{\xi}, W^{(\ell)}\rangle^2 / \|W^{(\ell)}\|^2 \to 1 - O\big((d\lambda^2)^{-\ell}\big)$ in probability.*

We prove the above in Section 4.1.

The next lemma shows that these walk scores provide a good initialization for BP on the SBM graph. Some notation is needed:

- For a graph G, a vector $y \in \mathbb{R}^{\vec{E}(G)}$, $t \geqslant 1$, and $\gamma \in (0,1)$, and a vector $\boldsymbol{\sigma}^* \in \{\pm 1\}^n$ which will be clear from context, let $\mathrm{Ov}_{t,\gamma}(G, y)$ be the expected overlap $\mathbb{E}\ \mathrm{ov}(\widehat{\boldsymbol{\sigma}}, \boldsymbol{\sigma}^\star)$ of the output $\widehat{\boldsymbol{\sigma}}$ of $\mathsf{BP}_{t,\gamma}\big(G, \beta\sqrt{|\vec{E}(G)|}\, y\big)$, where the expectation is over the randomized rounding.
- let $\beta = \frac{1}{2}\sqrt{1 - 1/(d\lambda^2)}$, and
- returning to the broadcast tree Definition 3.1, let $\boldsymbol{W}_\ell = (d\lambda)^{-\ell} \sum_u \boldsymbol{x}_u$ be the sum over the spins of the depth-ℓ vertices of the broadcast tree, and let $\gamma_\ell = \mathbb{E}\big[\boldsymbol{x}_o \tanh\big(\beta \boldsymbol{W}_\ell / (\mathbb{E}\, \boldsymbol{W}_\ell^2)^{1/2}\big)\big]$. Equivalently, if $\boldsymbol{\tau} \in \{\pm 1\}$ is drawn with $\mathbb{E}[\boldsymbol{\tau} \mid \boldsymbol{W}_\ell] = \tanh\big(\beta \boldsymbol{W}_\ell / (\mathbb{E}\, \boldsymbol{W}_\ell^2)^{1/2}\big)$, which is how $\mathsf{BP}_{t,\gamma}$ rounds a score, then $\gamma_\ell = \mathbb{E}[\boldsymbol{x}_o \boldsymbol{\tau}]$ is the correlation of the root spin with the rounded, normalized depth-ℓ walk score.

Lemma 4.4 (BP initialized from constant-depth walk score is near-optimal)**.** *Let $d\lambda^2 > 1$ and $(\boldsymbol{G}, \boldsymbol{\sigma}^\star) \sim \mathrm{SBM}(n, d, \lambda)$. For every $\varepsilon > 0$ there is $t = t(d, \lambda, \varepsilon) \geqslant 1$ such that for every $\ell \geqslant 0$, with high probability, $\mathrm{Ov}_{t,\gamma_\ell}(\boldsymbol{G}, W^{(\ell)}/\|W^{(\ell)}\|) \geqslant \mathsf{OPT}_{d,\lambda} - \varepsilon$ and the same holds for $-W^{(\ell)}/\|W^{(\ell)}\|$.*

We prove the above in Section 4.2. Because ℓ is constant, we can prove Lemma 4.4 by comparing each tree-like neighborhood of $\boldsymbol{G}$ to a broadcast tree.

The next two lemmas, proved in Section 4.3, establish the robustness aspects of Theorem 4.2. Both can be established by simple direct analysis of the BP operation.

Lemma 4.5 (BP is stable under changes to the initialization)**.** *Let $d\lambda^2 > 1$ and $(\boldsymbol{G}, \boldsymbol{\sigma}^\star) \sim \mathrm{SBM}(n, d, \lambda)$. For every $t \geqslant 1$ there is $C = C(d, \lambda, t) > 0$ such that with high probability, for all $\gamma \in (0, 1)$ and all $y, y' \in \mathbb{R}^{\vec{E}(\boldsymbol{G})}$,*

$$\left|\mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, y) - \mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, y')\right| \leqslant C\|y - y'\|.$$

Lemma 4.6 (BP is stable under edge edits)**.** *Let $d\lambda^2 > 1$ and $(\boldsymbol{G}, \boldsymbol{\sigma}^\star) \sim \mathrm{SBM}(n, d, \lambda)$. For every $t \geqslant 1$ there is $C = C(d, \lambda, t) > 0$ such that with high probability the following holds for all $\eta, \gamma \in (0, 1)$. Let G' be a graph on $[n]$ with $|E(\boldsymbol{G}) \triangle E(G')| \leqslant \eta n$, let $y \in \mathbb{R}^{\vec{E}(G')}$ have $\|y\| \leqslant 1$, and let $y|_{\vec{E}(\boldsymbol{G})}$ be the restriction of y to $\vec{E}(\boldsymbol{G})$, that is, the vector in $\mathbb{R}^{\vec{E}(\boldsymbol{G})}$ that agrees with y on the directed edges of both graphs and vanishes elsewhere. Then $\left|\mathrm{Ov}_{t,\gamma}(G', y) - \mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, y|_{\vec{E}(\boldsymbol{G})})\right| \leqslant C\sqrt{\eta}$.*

The proofs of Lemmas 4.5 and 4.6 use only that with high probability $\boldsymbol{G}$ has $O_d(n)$ edges and its radius-$(t+1)$ neighborhoods are not too large on average.

We are ready to prove Theorem 4.2.

Proof of Theorem 4.2. We need to show how to choose t, γ and $\eta > 0$ such that there is a high-probability event under $(\boldsymbol{G}, \boldsymbol{\sigma}^\star) \sim \mathrm{SBM}(n, d, \lambda)$ for which $\mathrm{Ov}_{t,\gamma}(G', y) \geqslant \mathsf{OPT}_{d,\lambda} - \varepsilon$, for every G', y allowed by the theorem statement.

First, for any t, γ we eventually choose, by Lemma 4.6 there is a high probability event such that for every G', y allowed by the theorem, we have

$$\mathrm{Ov}_{t,\gamma}(G', y) \geqslant \mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, y|_{\vec{E}(\boldsymbol{G})}) - O_{d,\lambda,t}(\sqrt{\eta}),$$

where $y|_{\vec{E}(\boldsymbol{G})}$ is the restriction of y to $\vec{E}(\boldsymbol{G})$, as in Lemma 4.6.

Next, for any ℓ we eventually choose, write $\widehat{W} = W^{(\ell)}/\|W^{(\ell)}\|$, where $W^{(\ell)}$ is the walk score vector of the SBM graph $\boldsymbol{G}$. By Lemma 4.5, applied to the pairs of vectors $y|_{\vec{E}(\boldsymbol{G})}$ and $\pm\widehat{W}$, there is a high-probability event on which, for every G', y allowed by the theorem,

$$\mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, y|_{\vec{E}(\boldsymbol{G})}) \geqslant \min_{s=\pm 1} \mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, s\widehat{W}) - O_{d,\lambda,t}\left(\min_{s=\pm 1} \|y|_{\vec{E}(\boldsymbol{G})} - s\widehat{W}\|\right).$$

Next, by Lemma 4.3, with high probability $\min_{s=\pm 1} \|\boldsymbol{\xi} - s\widehat{W}\|^2 = 2 - 2|\langle \boldsymbol{\xi}, \widehat{W}\rangle| \leqslant 2(d\lambda^2)^{-(\ell+1)} + o(1)$, where $o(1)$ tends to 0 as $n \to \infty$ with all other parameters fixed. Since $\|y\| \leqslant 1$ and $\langle y, \boldsymbol{\xi}\rangle^2 \geqslant 1 - \eta$, also $\min_{s=\pm 1} \|y - s\boldsymbol{\xi}\|^2 \leqslant 2 - 2\sqrt{1-\eta} \leqslant 2\eta$. As vectors in $\mathbb{R}^{n^2-n}$, $y|_{\vec{E}(\boldsymbol{G})}$ is obtained from y

by setting its entries off $\vec{E}(\boldsymbol{G})$ to zero, and $\boldsymbol{\xi}$ and $\widehat{W}$ vanish off $\vec{E}(\boldsymbol{G})$, so $\|y|_{\vec{E}(\boldsymbol{G})} - s\widehat{W}\| \leqslant \|y - s\widehat{W}\|$ for both signs s. So by the triangle inequality

$$\min_{s=\pm 1} \|y|_{\vec{E}(\boldsymbol{G})} - s\widehat{W}\| \leqslant \min_{s=\pm 1} \|y - s\widehat{W}\| \leqslant \sqrt{2\eta} + \sqrt{2(d\lambda^2)^{-(\ell+1)}} + o(1)\,.$$

Finally, take $t = t(d, \lambda, \varepsilon/3)$ from Lemma 4.4 and $\gamma = \gamma_\ell$ (defined just before Lemma 4.4). Then for every ℓ, with high probability,

$$\min_{s=\pm 1} \mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, s\widehat{W}) \geqslant \mathsf{OPT}_{d,\lambda} - \varepsilon/3\,.$$

Putting these together, with high probability, for every G', y allowed by the theorem,

$$\mathrm{Ov}_{t,\gamma}(G', y) \geqslant \mathsf{OPT}_{d,\lambda} - \varepsilon/3 - O_{d,\lambda,t}\big(\sqrt{\eta} + (d\lambda^2)^{-(\ell+1)/2}\big) - o(1)\,.$$

Since t depends only on d, λ, ε, we can choose ℓ large and then η small, also depending only on d, λ, ε, so that the $O(\cdot)$ term is at most $\varepsilon/3$; for large n the $o(1)$ term is also at most $\varepsilon/3$.

Each round of $\mathsf{BP}_{t,\gamma}$ takes $O(|\vec{E}(G')|) = O(n)$ operations: each vertex u computes $\prod_{v\in\partial u}(1 \pm \lambda m_{v\to u})$ once and divides out one factor per outgoing message. This establishes the running time and completes the proof. ∎

In the sections that follow, we will use the following statement proved in [BLM18], which shows that the local neighborhoods of a graph drawn from the stochastic block model resemble broadcast processes on Galton–Watson trees.

Fact 4.7 (Local law [BLM18, Proposition 36])**.** *Let $q \leqslant L + 1$. Let f and F be functions of a rooted graph whose vertices carry ± 1 labels, depending only on the radius-q neighborhood of the root, with $|f| \leqslant F$ and F nondecreasing under the addition of edges. Write $f(\boldsymbol{G}, v)$ for f evaluated on $\boldsymbol{G}$ rooted at v with the labels $\boldsymbol{\sigma}^\star$, and $f(\boldsymbol{T}_{d,\lambda}, o)$ for f evaluated on the broadcast tree with its spins $\boldsymbol{x}$. Condition on labels $\boldsymbol{\sigma}^\star$ for which both communities have size $n/2 \pm n^{3/4}$. Then*

$$\begin{aligned}&\mathbb{E}\Big|\frac{1}{n}\sum_{v\in[n]} f(\boldsymbol{G}, v) - \mathbb{E}\, f(\boldsymbol{T}_{d,\lambda}, o)\Big| \\ &\qquad \leqslant C\, d^{q/2} n^{-1/8} \log n \, \max\Big\{\big(\mathbb{E}\max_v F(\boldsymbol{G}, v)^4\big)^{1/4},\ \big(\mathbb{E}\, F(\boldsymbol{T}_{d,\lambda}, o)^2\big)^{1/2}\Big\},\end{aligned}$$

where $C = C(d, \lambda)$.

4.1 The eigenvector is close to a constant-depth walk score

In this section, we establish Lemma 4.3, restated for convenience.

Lemma 4.3 (The eigenvector is close to a walk score)**.** *Let $d\lambda^2 > 1$, let $(\boldsymbol{G}, \boldsymbol{\sigma}^\star) \sim \mathrm{SBM}(n, d, \lambda)$, and let $\boldsymbol{\xi}$ be as in Theorem 4.2. For every fixed $\ell \geqslant 0$, $\langle \boldsymbol{\xi}, W^{(\ell)}\rangle^2 / \|W^{(\ell)}\|^2 \to 1 - O\big((d\lambda^2)^{-\ell}\big)$ in probability.*

Throughout, we fix a sufficiently small constant $\kappa > 0$ possibly depending on d and λ, and $L \coloneqq \kappa \log_d n$. We first show that $\boldsymbol{\xi}$ is close to the walk score $W^{(L)}$ of logarithmic depth, and then compare $W^{(L)}$ with $W^{(\ell)}$.

We will need the following two ingredients. The first shows that the second eigenvector of the nonbacktracking matrix is roughly equal to the logarithmic-depth walk scores, and can be found in [SM22, Eq. (5.1) and Section 5.2, Page 226].

Lemma 4.8 (The logarithmic-depth walk score is the eigenvector)**.** *Let $d\lambda^2 > 1$, let $(\boldsymbol{G}, \boldsymbol{\sigma}^\star) \sim \mathrm{SBM}(n, d, \lambda)$, and let $\boldsymbol{\xi}$ be as in Theorem 4.2. Then $\min_{s=\pm 1} \left\| \boldsymbol{\xi} - sW^{(L)} / \|W^{(L)}\| \right\| \to 0$ in probability.*

The second is a bound on the difference between consecutive walk scores from [SM22, Eq. (8.11)].

Lemma 4.9. *There exists a constant $C > 0$ depending on d and λ such that with high probability over $\boldsymbol{G}$, we have for every $\ell \leqslant L$:*

$$\left\| W^{(\ell+1)} - W^{(\ell)} \right\| \leqslant \sqrt{n} \cdot \left(C(\lambda^2 d)^{-\ell/2} + \frac{C^t \log^7 n}{\sqrt{n}} \right).$$

Proof of Lemma 4.3. Without loss of generality, assume $s = +1$ in Lemma 4.8. Then we have:

$$\begin{aligned} \left\| \boldsymbol{\xi} - \frac{W^{(\ell)}}{\|W^{(\ell)}\|} \right\| &\leqslant \left\| \boldsymbol{\xi} - \frac{W^{(L)}}{\|W^{(L)}\|} \right\| + \left\| \frac{W^{(L)}}{\|W^{(L)}\|} - \frac{W^{(\ell)}}{\|W^{(\ell)}\|} \right\| \\ &\leqslant o(1) + 2 \frac{\left\| W^{(L)} - W^{(\ell)} \right\|}{\|W^{(L)}\|} \end{aligned}$$

where we bounded the first term with Lemma 4.8. The numerator of the second term can be bounded by $(C(d\lambda^2)^{-\ell/2} + o(1)) \cdot \sqrt{n}$ by triangle inequality and Lemma 4.9. The denominator is lower bounded by $\Omega(\sqrt{n})$ by [SM22, Proposition 10, Eq. (8.9)], and therefore the bound above translates to $O\big((d\lambda^2)^{-\ell/2}\big) + o(1)$. Rearranging this bound yields the desired statement. ∎

4.2 BP from the walk score is near-optimal

In this section, we establish Lemma 4.4.

Lemma 4.4 (BP initialized from constant-depth walk score is near-optimal)**.** *Let $d\lambda^2 > 1$ and $(\boldsymbol{G}, \boldsymbol{\sigma}^\star) \sim \mathrm{SBM}(n, d, \lambda)$. For every $\varepsilon > 0$ there is $t = t(d, \lambda, \varepsilon) \geqslant 1$ such that for every $\ell \geqslant 0$, with high probability, $\mathrm{Ov}_{t,\gamma_\ell}\big(\boldsymbol{G}, W^{(\ell)} / \|W^{(\ell)}\|\big) \geqslant \mathsf{OPT}_{d,\lambda} - \varepsilon$ and the same holds for $-W^{(\ell)} / \|W^{(\ell)}\|$.*

Proof. We will prove the desired statement by comparing the following pair of experiments we perform after conditioning on $(\boldsymbol{G}, \boldsymbol{\sigma}^\star)$. We will choose t sufficiently large depending on ε.

Experiment 1. Pick a uniformly random vertex $\boldsymbol{o} \sim V(\boldsymbol{G})$. Let $\boldsymbol{T}_{\boldsymbol{o}}$ be the radius-$t + \ell + 1$ ball around $\boldsymbol{o}$, and let $\boldsymbol{x}$ be a labeling of the vertices of $\boldsymbol{T}$ according to its true labeling $\boldsymbol{\sigma}^\star$. Denote by $\mathcal{D}_1$ the law of $(\boldsymbol{T}, \boldsymbol{x})$ arising from this experiment.

Experiment 2. Draw a tree $\boldsymbol{T}$ rooted at a vertex o along with a labeling $\boldsymbol{x}$ of the vertices from the depth $t+\ell+1$ broadcast process tree with parameters d and λ as in Definition 3.1. Denote by $\mathcal{D}_2$ the law of $(\boldsymbol{T}, \boldsymbol{x})$ arising from this experiment.

Let $(\boldsymbol{T}, \boldsymbol{x})$ be a pair arising from Experiment 1 or 2 where $\boldsymbol{T}$ is rooted at a vertex o. Now, consider the following process.

- For each vertex v at depth-t, and each child w of v, define $\boldsymbol{W}^{(\ell,\boldsymbol{T})}$ as the vector

$$\boldsymbol{W}^{(\ell,\boldsymbol{T})}_{v\to w} = (d\lambda)^{-\ell} \sum_{\substack{a \text{ descendant of } w \\ \text{at depth } t+\ell+1}} \boldsymbol{x}_a \,.$$

- For each such $w \to v$, let $\boldsymbol{y}_{w\to v}$ be drawn from $\{\pm\gamma_\ell\}$ such that $\mathbb{E}\,\boldsymbol{y}_{w\to v} = \gamma_\ell \tanh\left(\frac{\beta \boldsymbol{W}^{(\ell,\boldsymbol{T})}_{v\to w}}{\sqrt{\mathbb{E}\,\boldsymbol{W}_\ell^2}}\right)$.

- Let $\widehat{\boldsymbol{x}}_o$ be the outcome of running t rounds of BP with initial messages given by $\boldsymbol{y}$.

Observe that $\mathrm{Ov}_{t,\gamma_\ell}\left(\boldsymbol{G}, \boldsymbol{W}^{(\ell)}/\|\boldsymbol{W}^{(\ell)}\|\right) = \mathbb{E}_{(\boldsymbol{T},\boldsymbol{x})\sim\mathcal{D}_1} \boldsymbol{x}_{\boldsymbol{o}} \cdot \widehat{\boldsymbol{x}}_{\boldsymbol{o}}$.

A consequence of Fact 4.7 is the fact that $d_{\mathrm{TV}}(\mathcal{D}_1, \mathcal{D}_2) = o(1)$ with high probability over $\boldsymbol{G}$. Therefore, we have:

$$\mathop{\mathbb{E}}_{(\boldsymbol{T},\boldsymbol{x})\sim\mathcal{D}_1} \boldsymbol{x}_{\boldsymbol{o}} \cdot \widehat{\boldsymbol{x}}_{\boldsymbol{o}} \geqslant \mathop{\mathbb{E}}_{(\boldsymbol{T},\boldsymbol{x})\sim\mathcal{D}_2} \boldsymbol{x}_{\boldsymbol{o}} \cdot \widehat{\boldsymbol{x}}_{\boldsymbol{o}} - o(1) \geqslant \mathsf{OPT}_{d,\lambda} - o(1)\,,$$

where the first inequality used closeness in total variation distance. The second inequality follows from Fact 3.6; we elaborate on details below. Under Experiment 2, conditional on the labels at depth-t, the labels on the subtrees rooted at each of the depth-t vertices are independent. Thus, $\frac{\boldsymbol{W}^{(\ell,\boldsymbol{T})}}{\sqrt{\mathbb{E}\,\boldsymbol{W}_\ell^2}}$ restricted to the depth-t leaves can be construed as passing each label σ_v through an independent noisy channel. It can also be verified that there is an absolute constant $c > 0$ such that $\gamma_\ell > c$, and therefore the hypothesis of Fact 3.6 applies. In particular, we may choose t independent of ℓ. ■

4.3 Stability of BP

In this subsection we prove Lemmas 4.5 and 4.6, starting with Lemma 4.5. Throughout, for a graph G, a vertex v and $s \geqslant 0$, we write $|B_G(v,s)|$ for the number of vertices of G within distance s of v.

4.3.1 Proof of Lemma 4.5

Lemma 4.5 (BP is stable under changes to the initialization)**.** *Let $d\lambda^2 > 1$ and $(\boldsymbol{G}, \boldsymbol{\sigma}^\star) \sim \mathrm{SBM}(n, d, \lambda)$. For every $t \geqslant 1$ there is $C = C(d,\lambda,t) > 0$ such that with high probability, for all $\gamma \in (0,1)$ and all $y, y' \in \mathbb{R}^{\vec{E}(\boldsymbol{G})}$,*

$$\left|\mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, y) - \mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, y')\right| \leqslant C\|y - y'\|.$$

The idea behind Lemma 4.5 is that after t rounds of BP, the output at a vertex v depends only on the initialization Y at directed edges near v. Furthermore, if we round two initializations Y and Y' using the same randomness, the two outputs can differ at v only if the two roundings differ at some

directed edge e near v, which happens with probability at most $|Y_e - Y'_e|/2$. Summing over v, the expected number of vertices at which the outputs differ is at most $\|Y - Y'\|$ times a measure of the size of a typical neighborhood. We isolate this deterministic part of the argument in the following lemma.

Lemma 4.10 (BP outputs from nearby scores can be coupled)**.** *Let G be a graph on $[n]$, let $t \geqslant 1$ and $\gamma \in (0,1)$, and let $Y, Y' \in \mathbb{R}^{\vec{E}(G)}$. Then the outputs $\widehat{\sigma}$ and $\widehat{\sigma}'$ of $\mathsf{BP}_{t,\gamma}(G, Y)$ and $\mathsf{BP}_{t,\gamma}(G, Y')$ can be coupled so that*

$$\mathbb{E}\,\big|\{v : \widehat{\sigma}_v \neq \widehat{\sigma}'_v\}\big| \leqslant \frac{1}{2}\Big(\sum_{v\in[n]} |B_G(v, t+1)|^3\Big)^{1/2} \|Y - Y'\|. \tag{3}$$

The second ingredient is that neighborhoods in the block model are small on average. This is an immediate corollary of Fact 4.7, and that the moments of $|B_{T_{d,\lambda}}(o, s)|$ and $|B_{\boldsymbol{G}}(v, s)|$ are finite.

Lemma 4.11 (Neighborhoods are small on average)**.** *Let $d\lambda^2 > 1$ and $(\boldsymbol{G}, \boldsymbol{\sigma}^\star) \sim \mathrm{SBM}(n, d, \lambda)$. For every $s \geqslant 1$ there is $C = C(d, \lambda, s)$ such that, with high probability,*

$$\sum_{v\in[n]} |B_{\boldsymbol{G}}(v, s)|^3 \leqslant Cn.$$

We now put these together, and then prove Lemma 4.10.

Proof of Lemma 4.5. Let $C_0 = C(d, \lambda, t+1)$ be the constant of Lemma 4.11, and condition on the high-probability event that $\sum_v |B_{\boldsymbol{G}}(v, t+1)|^3 \leqslant C_0 n$. On this event also $|\vec{E}(\boldsymbol{G})| \leqslant C_0 n$.

Fix $\gamma \in (0,1)$ and $y, y' \in \mathbb{R}^{\vec{E}(\boldsymbol{G})}$. Let $\widehat{\sigma}$ and $\widehat{\sigma}'$ be the outputs of $\mathsf{BP}_{t,\gamma}$ initialized at $Y = \beta\sqrt{|\vec{E}(\boldsymbol{G})|}\, y$ and $Y' = \beta\sqrt{|\vec{E}(\boldsymbol{G})|}\, y'$, respectively, coupled as in Lemma 4.10. Changing the output at one vertex changes the overlap by at most $2/n$. Indeed, since $\mathrm{ov}(\widehat{\sigma}, \boldsymbol{\sigma}^\star) = \frac{1}{n}|\langle\widehat{\sigma}, \boldsymbol{\sigma}^\star\rangle|$ and $\big||x| - |x'|\big| \leqslant |x - x'|$ for real numbers x, x',

$$\begin{aligned}\big|\mathrm{ov}(\widehat{\sigma}, \boldsymbol{\sigma}^\star) - \mathrm{ov}(\widehat{\sigma}', \boldsymbol{\sigma}^\star)\big| &= \frac{1}{n}\Big|\,|\langle\widehat{\sigma}, \boldsymbol{\sigma}^\star\rangle| - |\langle\widehat{\sigma}', \boldsymbol{\sigma}^\star\rangle|\,\Big| \\ &\leqslant \frac{1}{n}|\langle\widehat{\sigma} - \widehat{\sigma}', \boldsymbol{\sigma}^\star\rangle| \leqslant \frac{2}{n}\,\big|\{v : \widehat{\sigma}_v \neq \widehat{\sigma}'_v\}\big|.\end{aligned}$$

The quantities $\mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, y)$ and $\mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, y')$ are the expectations of the two overlaps, whatever the coupling. So taking expectations and applying (3) with $\|Y - Y'\| = \beta\sqrt{|\vec{E}(\boldsymbol{G})|}\,\|y - y'\|$,

$$\begin{aligned}\big|\mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, y) - \mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, y')\big| &\leqslant \frac{2}{n}\,\mathbb{E}\,\big|\{v : \widehat{\sigma}_v \neq \widehat{\sigma}'_v\}\big| \\ &\leqslant \frac{1}{n}\Big(\sum_v |B_{\boldsymbol{G}}(v, t+1)|^3\Big)^{1/2} \beta\sqrt{|\vec{E}(\boldsymbol{G})|}\,\|y - y'\| \\ &\leqslant \frac{1}{n}\sqrt{C_0 n}\cdot\beta\sqrt{C_0 n}\,\|y - y'\| = \beta C_0\,\|y - y'\|.\end{aligned}$$

This proves Lemma 4.5 with $C = \beta C_0$, which depends only on d, λ and t. ∎

Proof of Lemma 4.10. We first couple the random rounding in Definition 4.1. For each directed edge e, let $\boldsymbol{U}_e$ be uniform on $[-1,1]$, independently over e, and set

$$\boldsymbol{\tau}_e = \begin{cases} 1 & \text{if } \boldsymbol{U}_e \leqslant \tanh Y_e, \\ -1 & \text{otherwise,} \end{cases}$$

and define $\boldsymbol{\tau}'_e$ in the same way from Y'_e, with the same $\boldsymbol{U}_e$. Then $\mathbb{E}\,\boldsymbol{\tau}_e = \tanh Y_e$ and $\mathbb{E}\,\boldsymbol{\tau}'_e = \tanh Y'_e$, as required, and $\boldsymbol{\tau}_e \neq \boldsymbol{\tau}'_e$ only if $\boldsymbol{U}_e$ lies between $\tanh Y_e$ and $\tanh Y'_e$, an interval of length $|\tanh Y_e - \tanh Y'_e|$ on which $\boldsymbol{U}_e$ has density $1/2$. Since tanh is 1-Lipschitz,

$$\boldsymbol{Pr}(\boldsymbol{\tau}_e \neq \boldsymbol{\tau}'_e) = \frac{1}{2}\,|\tanh Y_e - \tanh Y'_e| \leqslant \frac{1}{2}\,|Y_e - Y'_e|. \tag{4}$$

Let $\widehat{\boldsymbol{\sigma}}$ and $\widehat{\boldsymbol{\sigma}}'$ be the outputs of the remaining steps of $\mathsf{BP}_{t,\gamma}$ run on $\boldsymbol{\tau}$ and on $\boldsymbol{\tau}'$.

Since t rounds of BP only propagate information a distance t, the output $\widehat{\boldsymbol{\sigma}}_v$ depends only on $\boldsymbol{\tau}_e$ for directed edges e whose head is within distance $t+1$ of v, and likewise for $\widehat{\boldsymbol{\sigma}}'_v$ and $\boldsymbol{\tau}'$.

So $\widehat{\boldsymbol{\sigma}}_v \neq \widehat{\boldsymbol{\sigma}}'_v$ only if $\boldsymbol{\tau}_e \neq \boldsymbol{\tau}'_e$ for some directed edge e whose head is within distance $t+1$ of v. The head v of a directed edge $u \to v$ is within distance $t+1$ of exactly $|B_G(v,t+1)|$ vertices. So a union bound over these edges, followed by (4), gives

$$\mathbb{E}\,\big|\{v : \widehat{\boldsymbol{\sigma}}_v \neq \widehat{\boldsymbol{\sigma}}'_v\}\big| \leqslant \sum_{u\to v} |B_G(v,t+1)|\,\boldsymbol{Pr}(\boldsymbol{\tau}_{u\to v} \neq \boldsymbol{\tau}'_{u\to v}) \leqslant \frac{1}{2}\sum_{u\to v} |B_G(v,t+1)|\,|Y_{u\to v} - Y'_{u\to v}|.$$

By the Cauchy–Schwarz inequality, and since each vertex v is the head of $\deg(v)$ directed edges,

$$\begin{aligned}\sum_{u\to v} |B_G(v,t+1)|\,|Y_{u\to v} - Y'_{u\to v}| &\leqslant \Big(\sum_{u\to v} |B_G(v,t+1)|^2\Big)^{1/2} \|Y - Y'\| \\ &= \Big(\sum_{v\in[n]} \deg(v)\;|B_G(v,t+1)|^2\Big)^{1/2} \|Y - Y'\|.\end{aligned}$$

Finally $\deg(v) \leqslant |B_G(v,t+1)|$, which gives (3). ∎

4.3.2 Proof of Lemma 4.6

Lemma 4.6 (BP is stable under edge edits)**.** *Let $d\lambda^2 > 1$ and $(\boldsymbol{G}, \boldsymbol{\sigma}^\star) \sim \mathrm{SBM}(n,d,\lambda)$. For every $t \geqslant 1$ there is $C = C(d,\lambda,t) > 0$ such that with high probability the following holds for all $\eta, \gamma \in (0,1)$. Let G' be a graph on $[n]$ with $|E(\boldsymbol{G}) \triangle E(G')| \leqslant \eta n$, let $y \in \mathbb{R}^{\vec{E}(G')}$ have $\|y\| \leqslant 1$, and let $y|_{\vec{E}(\boldsymbol{G})}$ be the restriction of y to $\vec{E}(\boldsymbol{G})$, that is, the vector in $\mathbb{R}^{\vec{E}(G)}$ that agrees with y on the directed edges of both graphs and vanishes elsewhere. Then $\big|\mathrm{Ov}_{t,\gamma}(G',y) - \mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, y|_{\vec{E}(\boldsymbol{G})})\big| \leqslant C\sqrt{\eta}$.*

The idea behind the above is again locality: if a vertex v is far from every corrupted edge, then t rounds of BP compute the same output at v on both graphs if initialized at the same vector. By Lemma 4.11, most vertices have no corrupted edges nearby. Apart from this, BP on the corrupted graph and non-corrupted graph differ only in the normalization $\beta\sqrt{|\vec{E}(G)|}$ of the initialization, which changes by a factor $1 + O(\eta)$ and is handled by Lemma 4.5. We isolate the deterministic part of the argument in the following lemma.

Lemma 4.12 (BP only sees nearby edits). *Let G and G' be graphs on $[n]$, let $t \geqslant 1$ and $\gamma \in (0,1)$, and let A be the set of endpoints of the edges in $E(G) \triangle E(G')$. Let $y \in \mathbb{R}^{n^2-n}$. Then the outputs $\widehat{\sigma}$ of $\mathsf{BP}_{t,\gamma}(G, y|_{\vec{E}(G)})$ and $\widehat{\sigma}'$ of $\mathsf{BP}_{t,\gamma}(G', y|_{\vec{E}(G')})$ can be coupled so that with probability 1 over the randomized rounding of the initialization,*

$$\big|\{v : \widehat{\sigma}_v \neq \widehat{\sigma}'_v\}\big| \leqslant \sum_{a \in A} |B_G(a,t)|.$$

We now prove Lemma 4.6, and then Lemma 4.12.

Proof of Lemma 4.6. Let C_0 be the constant of Lemma 4.11 with $s = t+1$, and let $C_1 = C(d,\lambda,t)$ be the constant of Lemma 4.5. We work on the intersection of the high-probability events of these two lemmas and the event $|\vec{E}(\boldsymbol{G})| \geqslant dn/2$, which has high probability because $|\vec{E}(\boldsymbol{G})|/n \to d$ in probability. Fix $\eta, \gamma \in (0,1)$, and G' and y as in the statement, and let

$$c = \Big(\frac{|\vec{E}(G')|}{|\vec{E}(\boldsymbol{G})|}\Big)^{1/2}.$$

We compare $\mathrm{Ov}_{t,\gamma}(G', y)$ and $\mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, y|_{\vec{E}(\boldsymbol{G})})$ with $\mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, c\,y|_{\vec{E}(\boldsymbol{G})})$.

Changing the graph. We apply Lemma 4.12 with $\beta\sqrt{|\vec{E}(G')|}\,y$ in place of y. Its restriction to $\vec{E}(\boldsymbol{G})$ is $\beta\sqrt{|\vec{E}(G')|}\,y|_{\vec{E}(\boldsymbol{G})} = \beta\sqrt{|\vec{E}(\boldsymbol{G})|}\,c\,y|_{\vec{E}(\boldsymbol{G})}$. So the expected overlaps of the two outputs in Lemma 4.12 are $\mathrm{Ov}_{t,\gamma}(G', y)$ and $\mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, c\,y|_{\vec{E}(\boldsymbol{G})})$. As in the proof of Lemma 4.5, changing the output at one vertex changes the overlap by at most $2/n$, so Lemma 4.12 gives

$$\big|\mathrm{Ov}_{t,\gamma}(G', y) - \mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, c\,y|_{\vec{E}(\boldsymbol{G})})\big| \leqslant \frac{2}{n} \sum_{a \in A} |B_{\boldsymbol{G}}(a,t)|,$$

where A is the set of endpoints of the edges in $E(\boldsymbol{G}) \triangle E(G')$, so that $|A| \leqslant 2\eta n$. Then we have

$$\sum_{a \in A} |B_{\boldsymbol{G}}(a,t)| \leqslant |A|^{1/2}\Big(\sum_{v \in [n]} |B_{\boldsymbol{G}}(v,t)|^2\Big)^{1/2} \leqslant |A|^{1/2}\Big(\sum_{v \in [n]} |B_{\boldsymbol{G}}(v,t+1)|^3\Big)^{1/2} \leqslant (2\eta n)^{1/2}(C_0 n)^{1/2} = \sqrt{2C_0\eta}\; n\,,$$

where we used Cauchy–Schwarz, then $1 \leqslant |B_{\boldsymbol{G}}(v,t)| \leqslant |B_{\boldsymbol{G}}(v,t+1)|$, and finally Lemma 4.11.

Changing the normalization. By Lemma 4.5, and since $\|y|_{\vec{E}(\boldsymbol{G})}\| \leqslant \|y\| \leqslant 1$,

$$\big|\mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, c\,y|_{\vec{E}(\boldsymbol{G})}) - \mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, y|_{\vec{E}(\boldsymbol{G})})\big| \leqslant C_1\,|c-1|\,\|y|_{\vec{E}(\boldsymbol{G})}\| \leqslant C_1\,|c-1|.$$

Each edge in $E(\boldsymbol{G}) \triangle E(G')$ accounts for two directed edges, so $\big||\vec{E}(G')| - |\vec{E}(\boldsymbol{G})|\big| \leqslant 2\eta n$. Since $|\sqrt{x}-1| \leqslant |x-1|$ for $x \geqslant 0$ and $|\vec{E}(\boldsymbol{G})| \geqslant dn/2$,

$$|c-1| \leqslant \frac{\big||\vec{E}(G')| - |\vec{E}(\boldsymbol{G})|\big|}{|\vec{E}(\boldsymbol{G})|} \leqslant \frac{2\eta n}{dn/2} = \frac{4\eta}{d}.$$

Putting the two steps together, and using $\eta \leqslant \sqrt{\eta}$,

$$\big|\mathrm{Ov}_{t,\gamma}(G', y) - \mathrm{Ov}_{t,\gamma}(\boldsymbol{G}, y|_{\vec{E}(\boldsymbol{G})})\big| \leqslant 2\sqrt{2C_0\eta} + \frac{4C_1\eta}{d} \leqslant \Big(2\sqrt{2C_0} + \frac{4C_1}{d}\Big)\sqrt{\eta}.$$

This proves Lemma 4.6 with $C = 2\sqrt{2C_0} + 4C_1/d$, which depends only on d, λ and t. ∎

Proof of Lemma 4.12. We couple the random rounding of the initialization in the two runs of BP as in the proof of Lemma 4.10. That is, we use the same uniform variable $\boldsymbol{U}_e$ for each directed edge e that belongs to both graphs, and independent uniform variables for all other directed edges. Since the score of such an edge e is y_e in both runs, the rounded spins agree on every directed edge of both graphs.

Now, if the distance-t neighborhood of a vertex v in G is disjoint from A, then every vertex within distance t of v has the same neighbors in G and in G'. So the vertices within distance $t+1$ of v, and the edges incident to vertices within distance t of v, are the same in the two graphs. The output at v after t rounds of BP is computed from these edges and the rounded spins on them, so $\widehat{\sigma}_v = \widehat{\sigma}'_v$. Every other vertex is within distance t of some $a \in A$ in G, and there are at most $\sum_{a \in A} |B_G(a,t)|$ such vertices. ∎

5 Robustly estimating the second eigenvector of the nonbacktracking matrix

We prove the following main theorem for this section.

Theorem 5.1 (Robust estimation of the informative eigenvector)**.** *For every $d \geqslant 1$, $\lambda \in (-1,1)$ with $d\lambda^2 > 1$, and $\varepsilon > 0$ there is $\eta > 0$ and a nearly-linear time algorithm with the following guarantee. Let $(\boldsymbol{G}, \boldsymbol{\sigma}^\star) \sim \mathrm{SBM}(n,d,\lambda)$ and let $\boldsymbol{\xi}$ be a unit eigenvector for the eigenvalue of $B_{\boldsymbol{G}}$ of second-largest modulus. With high probability over $(\boldsymbol{G}, \boldsymbol{\sigma}^\star)$, for every graph F satisfying $|E(F) \triangle E(\boldsymbol{G})| \leqslant \eta n$, with high probability over its internal randomness the algorithm outputs $\widehat{\xi}(F)$ with*

$$\min_{a \in \{\pm 1\}} \|\widehat{\xi}(F) - a\boldsymbol{\xi}\| \leqslant \varepsilon.$$

We will gather the lemmas we need, capturing several algorithmic subroutines, and then prove Theorem 5.1.

A key challenge in designing and analyzing an algorithm for Theorem 5.1 is the asymmetry of the nonbacktracking matrix—in particular, $\boldsymbol{\xi} \in \mathbb{R}^{\vec{E}(G)}$ does not live in an orthonormal basis of eigenvectors. To design an algorithm which proves Theorem 5.1, we therefore take a somewhat indirect approach, inspired by [MRW24]. We reduce to the problem of recovering, in the presence of corruptions, a certain n-dimensional vector which counts signed nonbacktracking walks, defined below. We will be able to postprocess this vector in a simple (linear time) way to recover $\boldsymbol{\xi}$.

Definition 5.2. For a graph G on $[n]$ and $\ell \in \mathbb{N}$, let $A^{(\ell)} \in \mathbb{R}^{n \times n}$ count length-ℓ nonbacktracking walks in G. We write $A_G^{(\ell)}$ if G is not clear from context, and by convention let $A^{(0)} = I$. Moreover, for $(\boldsymbol{G}, \boldsymbol{\sigma}^*) \sim \mathrm{SBM}$, let $\boldsymbol{\sigma}^{(s)} = (d\lambda)^{-s} A^{(s)} \boldsymbol{\sigma}^*$.

The choice of $(d\lambda)^{-s}$ as a normalization factor will ensure that $\boldsymbol{\sigma}^{(s)}$ converges to a nonzero vector as $s \to \infty$.

For $(\boldsymbol{G}, \boldsymbol{\sigma}^*) \sim \mathrm{SBM}$, the following lemma, proved in Section 9, lets us reduce to the problem of recovering $\boldsymbol{\sigma}^{(s)}$ given an adversarial corruption of $\boldsymbol{G}$.

Lemma 5.3 (Robust postprocessing). *Let $(\boldsymbol{G}, \boldsymbol{\sigma}^\star) \sim \mathrm{SBM}(n, d, \lambda)$ and $\boldsymbol{\xi}$ be a unit eigenvector for the eigenvalue of $B_{\boldsymbol{G}}$ of second-largest modulus. For every $d \geqslant 1$, $\lambda \in (-1, 1)$ with $d\lambda^2 > 1$, and $\varepsilon > 0$, there are $\eta > 0$, $\delta > 0$, $s_0 \in \mathbb{N}$, and a linear-time algorithm with the following guarantee for every $s \geqslant s_0$: with high probability over $(\boldsymbol{G}, \boldsymbol{\sigma}^\star)$, given input $y \in \mathbb{R}^n$ satisfying*

$$\min_{a \in \{\pm 1\}} \left\| y - a \frac{\boldsymbol{\sigma}^{(s)}}{\|\boldsymbol{\sigma}^{(s)}\|} \right\| \leqslant \delta,$$

and an arbitrary graph F satisfying $|E(F) \triangle E(\boldsymbol{G})| \leqslant \eta n$, the algorithm outputs a vector $\widehat{\xi}(F)$ such that

$$\min_{a \in \{\pm 1\}} \|\widehat{\xi}(F) - a\boldsymbol{\xi}\| \leqslant \varepsilon.$$

Our strategy to robustly recover $\boldsymbol{\sigma}^{(s)}$ has two parts. First, we show that if we had access to the non-corrupted graph $\boldsymbol{G}$, a spectral algorithm based on the *Bethe Hessian* matrix (defined below) would recover $\boldsymbol{\sigma}^{(s)}$ with approximation error going to zero as $s \to \infty$. Then we show that even given access to an adversarially corrupted graph, there is a simple preprocessing step together with a semidefinite program which "robustifies" this spectral algorithm and hence still recovers $\boldsymbol{\sigma}^{(s)}$. The ability to robustify this spectral algorithm in a straightforward way is our reason for working with the $n \times n$ symmetric matrices rather than directly with the nonbacktracking matrix.

To state the main lemmas capturing these algorithms we need to define the key matrices involved, which we denote $M_s = M_s(G)$ where G is an underlying graph and $s \in \mathbb{N}$.

Definition 5.4 (Bethe Hessian)**.** Our starting point is the *Bethe Hessian* [SKZ14], a one-parameter family of $n \times n$ matrices associated to a graph G.

$$H(t) = I - tA + t^2(D - I).$$

The *Ihara–Bass formula* [Iha66, Bas92] gives a correspondence between t which make $H(t)$ singular and eigenvalues of the nonbacktracking matrix. We will use something more general, that at some carefully chosen values of t where $H(t)$ is nonsingular, $H(t)$ has a small number of negative eigenvectors and the vector we want, $\boldsymbol{\sigma}^{(s)}$, lies close to their span, after applying a simple linear transformation depending on $\boldsymbol{G}$. This strategy was also followed by [MRW24]. We refine it here by (approximately) isolating $\boldsymbol{\sigma}_{\boldsymbol{G}}^{(s)}$ in the span of these negative eigenvectors by applying a delicate rank-one correction to $H(t)$. This leads to more accurate recovery of $\boldsymbol{\sigma}_{\boldsymbol{G}}^{(s)}$ via the Bethe Hessian than was previously achieved. We package this into an analysis of a family of matrices we call "corrected Bethe Hessian matrices."

Definition 5.5 (Corrected Bethe-Hessian Matrices)**.** For each $s \in \mathbb{N}$ and n-vertex graph G, we define a matrix $M_s(G) \in \mathbb{R}^{n \times n}$. When the block-model parameters d, λ are clear from context, let $t = t(d, \lambda) = \frac{1}{d\lambda}\left(1 + \frac{\sqrt{d}-1}{d}\left(d\lambda^2 - 1\right)\right)$. [4] Since t is always now fixed to the quantity above, we just write $H = H(t)$ for the Bethe Hessian going forward.

[4] Although, a range of choices of t in $\left[\frac{1}{\lambda d}, \frac{1}{\sqrt{d}}\right)$ would work, and specific details of this choice are not so important.

Our choice of t results in 2 eigenvalues of H below zero (when $\lambda > 0$). Roughly, these correspond to the top two eigenvalues (by modulus) of the nonbacktracking matrix. We can remove one of those eigenvalues as follows. Let

$$b = \mathbf{1} + \sum_{j=1}^{L} t^j (A^{(j)} - dA^{(j-1)})\mathbf{1},$$

where $L = L(d, \lambda) \geqslant 2$ will be a sufficiently large constant (see Lemma 6.5(i)).

We will eventually see that, when $\lambda > 0$, the chosen b satisfies $b^\top Hb < 0$ with high probability. To align an eigenvector of H with $\boldsymbol{\sigma}^{(s)}$ we also need to multiply by $A_{\boldsymbol{G}}^{(s)}$. All together, we get the corrected Bethe Hessian:

$$M_s(G) = (d\lambda)^{-2s} \cdot \begin{cases} A^{(s)}\Big(H + \dfrac{2(Hb)(Hb)^\top}{-b^\top Hb}\Big)A^{(s)}, & \lambda > 0, \\ A^{(s)} H A^{(s)}, & \lambda < 0. \end{cases}$$

Now we can state the key properties of the M_s matrices when $\boldsymbol{G} \sim \mathrm{SBM}$. First is that $\boldsymbol{\sigma}^{(s)}$ can be recovered from its (unique) negative eigenvector; for making this recovery algorithm robust it is also important that this vector is delocalized in the sense of having small ℓ_4 norm.

Lemma 5.6 (Informative negative eigenvector)**.** *Let $d\lambda^2 > 1$. There is a constant $g(d, \lambda) > 0$ such that for every $\varepsilon > 0$ there is $s_0(\varepsilon)$ such that for every $s > s_0$, with high probability M_s has exactly one negative eigenvalue, and $\boldsymbol{\sigma}^{(s)}$ is approximately an associated eigenvector, in that*

$$\big\|(M_s + gI)\boldsymbol{\sigma}^{(s)}\big\| \leqslant \varepsilon \big\|\boldsymbol{\sigma}^{(s)}\big\|.$$

Moreover, there is $C = C(d, \lambda)$, not depending on s, such that with high probability, $\boldsymbol{\sigma}^{(s)}$ is delocalized, in that

$$\frac{\|\boldsymbol{\sigma}^{(s)}\|_4^4}{\|\boldsymbol{\sigma}^{(s)}\|^4} \leqslant \frac{C}{n}.$$

We prove the above in Section 6. In light of Lemma 5.6, our goal will be to recover the negative eigenvector of the corrected Bethe Hessian $M_s(\boldsymbol{G})$, given access only to an adversarially corrupted graph F. A first ingredient is to define a norm in which the adversarial corruptions cannot change M_s too much (allowing for some light preprocessing).

Definition 5.7 (Row-sum norms)**.** For $E \in \mathbb{R}^{n \times n}$,

$$\|E\|_{1,2} = \left(\sum_{i \in [n]} \left(\sum_{j \in [n]} |E_{ij}|\right)^2\right)^{1/2} \quad \text{and} \quad \|E\|_{1,\infty} = \max_{i \in [n]} \sum_{j \in [n]} |E_{ij}|.$$

A simple preprocessing routine – removing high degree vertices – can use the adversarially-corrupted graph F and arrive at a matrix $\widetilde{M}$ which is close to $M_s(\boldsymbol{G})$.

Lemma 5.8 (Preprocessing for robustness). *Let $(\boldsymbol{G}, \boldsymbol{\sigma}^\star) \sim \mathrm{SBM}(n, d, \lambda)$ and $d\lambda^2 > 1$. For every $s \in \mathbb{N}$ and $\varepsilon > 0$, there is a linear-time algorithm, $C \geqslant 1$ and $\eta > 0$ which with high probability over $\boldsymbol{G}$, given as input any n-vertex graph F such that $|E(F) \triangle E(\boldsymbol{G})| \leqslant \eta n$, produces a symmetric matrix $\widetilde{M} \in \mathbb{R}^{n \times n}$ such that*

$$\|\widetilde{M} - M_s(\boldsymbol{G})\|_{1,2} \leqslant \varepsilon\sqrt{n}, \qquad \|\widetilde{M}\|_{1,\infty} \leqslant C\,.$$

Moreover, matrix-vector products by $\widetilde{M}$ can be computed in linear time.

We prove the above in Section 7. Finally, we arrive at the key algorithm: a convex program which approximately recovers a delocalized negative eigenvector from a matrix, tolerating perturbations in $\|\cdot\|_{1,2}$ norm. Its analysis is presented in Section 8.

Lemma 5.9 (Robust recovery of a delocalized negative eigenvector). *There is a randomized algorithm with the following properties. Let $\varepsilon, g > 0$ and $M, \widetilde{M} \in \mathbb{R}^{n \times n}$ be symmetric, with M having exactly one negative eigenvalue $-g$. Suppose there exists a unit vector $w \in \mathbb{R}^n$ satisfying*

$$\|w\|_4^4 \leqslant \frac{1}{\varepsilon n} \quad \text{and} \quad w^\top M w \leqslant -(1-\varepsilon)g\,.$$

Suppose also that $\|\widetilde{M} - M\|_{1,2} \leqslant \varepsilon g\sqrt{n}$. Then, given matrix-vector access to $\widetilde{M}$, an upper bound $C \geqslant \|\widetilde{M}\|_{1,\infty} + 1$, and ε (but not g) as inputs, the algorithm returns a unit vector $\hat{\boldsymbol{w}}$ such that with probability at least $1 - n^{-10}$ over the internal randomness of the algorithm,

$$\langle \hat{\boldsymbol{w}}, w\rangle^2 \geqslant 1 - O(\sqrt{\varepsilon}).$$

On the same event, the algorithm performs at most $\left(1 + \frac{C\log n}{\varepsilon g}\right)^{O(1)}$ matrix-vector products with $\widetilde{M}$ and at most n times as many additional operations.

The algorithm applies matrix entropy mirror descent (that is, the matrix multiplicative weights update) to approximately minimize the following convex function F over the set of density matrices $\{X \in \mathbb{R}^{n\times n} : X \succeq 0,\ \mathrm{Tr}\, X = 1\}$

$$F(X) = \langle \widetilde{M}, X\rangle + C \cdot \mathrm{dist}_1(\mathrm{diag}\, X, \mathcal{D}_{1/\varepsilon}),$$

where $\mathcal{D}_{1/\varepsilon} = \{q \in \mathbb{R}^n : \|q\|_2^2 \leqslant 1/(\varepsilon n)\}$.

We are ready to prove Theorem 5.1.

Proof of Theorem 5.1. The algorithm runs the preprocessing algorithm of Lemma 5.8 on F, gives the resulting matrix $\widetilde{M}$ to the algorithm of Lemma 5.9, and runs the postprocessing algorithm of Lemma 5.3 on the resulting unit vector $\hat{w}$ and on F.

We choose the parameters in the following order. Let η_1, δ, and s_0 be given by Lemma 5.3 with accuracy ε. Let g and C be the constants of Lemma 5.6, and let $\varepsilon' > 0$ be small enough that $\varepsilon' \leqslant 1/C$ and the $O(\sqrt{\varepsilon'})$ in Lemma 5.9 is at most $\delta^2/2$. Fix $s \geqslant s_0$ large enough to apply Lemma 5.6 with accuracy $\varepsilon' g/4$. Let η_2 and C' be given by Lemma 5.8 for this s and accuracy $\varepsilon' g/2$, and let $\eta = \min\{\eta_1, \eta_2\}$. The algorithm of Lemma 5.9 is run with accuracy ε' and upper bound $C' + 1$.

With high probability over $(\boldsymbol{G}, \boldsymbol{\sigma}^\star)$, the conclusions of Lemmas 5.3, 5.6 and 5.8 all hold. Fix such $(\boldsymbol{G}, \boldsymbol{\sigma}^\star)$ and a graph F with $|E(F) \triangle E(\boldsymbol{G})| \leqslant \eta n$. Let $M = M_s(\boldsymbol{G})$, let $-g'$ be its unique negative eigenvalue, and let $w = \boldsymbol{\sigma}^{(s)}/\|\boldsymbol{\sigma}^{(s)}\|$, so that $\|(M + gI)w\| \leqslant \varepsilon' g/4$. Expanding w in an eigenbasis of M shows that M has an eigenvalue within $\varepsilon' g/4$ of $-g$. That eigenvalue is negative, so it is $-g'$, and $|g' - g| \leqslant \varepsilon' g/4$; in particular $g' \geqslant g/2$. So the hypotheses of Lemma 5.9 hold for M, $\widetilde{M}$, w, and ε':

$$\|w\|_4^4 \leqslant \frac{C}{n} \leqslant \frac{1}{\varepsilon' n}\,,$$
$$w^\top M w = w^\top (M + gI)w - g \leqslant -g' + \frac{\varepsilon' g}{2} \leqslant -(1 - \varepsilon')g'\,,$$
$$\|\widetilde{M} - M\|_{1,2} \leqslant \frac{\varepsilon' g}{2}\sqrt{n} \leqslant \varepsilon' g' \sqrt{n}\,,$$

and $\|\widetilde{M}\|_{1,\infty} \leqslant C'$. Hence with probability at least $1 - n^{-10}$ over its internal randomness, the algorithm returns $\hat{w}$ with $\langle \hat{w}, w\rangle^2 \geqslant 1 - \delta^2/2$, and then

$$\min_{a \in \{\pm 1\}} \|\hat{w} - aw\|^2 = 2 - 2|\langle \hat{w}, w\rangle| \leqslant 2\big(1 - \langle \hat{w}, w\rangle^2\big) \leqslant \delta^2\,.$$

So Lemma 5.3 gives $\min_{a \in \{\pm 1\}} \|\widehat{\xi}(F) - a\boldsymbol{\xi}\| \leqslant \varepsilon$.

For the running time, preprocessing and postprocessing take linear time. Since C', ε', and $g' \geqslant g/2$ are constants, Lemma 5.9 uses $(\log n)^{O(1)}$ matrix-vector products with $\widetilde{M}$, each taking linear time by Lemma 5.8, and $n(\log n)^{O(1)}$ further operations. ∎

6 Negative eigenvalues of the Bethe Hessian

In this section we prove Lemma 5.6.

Lemma 5.6 (Informative negative eigenvector)**.** *Let $d\lambda^2 > 1$. There is a constant $g(d, \lambda) > 0$ such that for every $\varepsilon > 0$ there is $s_0(\varepsilon)$ such that for every $s > s_0$, with high probability M_s has exactly one negative eigenvalue, and $\boldsymbol{\sigma}^{(s)}$ is approximately an associated eigenvector, in that*

$$\big\|(M_s + gI)\boldsymbol{\sigma}^{(s)}\big\| \leqslant \varepsilon \,\|\boldsymbol{\sigma}^{(s)}\|\,.$$

Moreover, there is $C = C(d, \lambda)$, not depending on s, such that with high probability, $\boldsymbol{\sigma}^{(s)}$ is delocalized, in that

$$\frac{\|\boldsymbol{\sigma}^{(s)}\|_4^4}{\|\boldsymbol{\sigma}^{(s)}\|^4} \leqslant \frac{C}{n}\,.$$

Throughout this section, $d\lambda^2 > 1$, and t, $H = H(t)$, b, and M_s are as in Definition 5.5. We have yet to choose the constant $L(d, \lambda)$ in the definition of b but we will do so in this section.

We split the proof into two lemmas.

Lemma 6.1 (Negative eigenvalue bound for the Bethe Hessian)**.** *With high probability,*

$$n_-(H) \leqslant \begin{cases} 2, & \lambda > 0, \\ 1, & \lambda < 0. \end{cases}$$

where $n_-(H)$ counts the number of strictly negative eigenvalues of H.

Lemma 6.1 is a quick consequence of [BLM18] and [MRW24] so we discharge its proof now.

Proof of Lemma 6.1. Recall that $B = B_G$ is the nonbacktracking matrix. By [MRW24, Lemma 5.4], $n_-(H)$ is at most the number of real eigenvalues θ of B with $t\theta > 1$, counted with multiplicity. By [BLM18, Theorem 4], with high probability all eigenvalues of B other than $d + o(1)$ and $d\lambda + o(1)$ have modulus at most $\sqrt{d} + o(1)$. Since $|t|\sqrt{d} < 1$, none of these other eigenvalues satisfies $t\theta > 1$, so $n_-(H) \leqslant 2$. When $\lambda < 0$, also $t < 0$, so $t\theta > 1$ forces $\theta < 0$, which rules out the eigenvalue near d; hence in this case $n_-(H) \leqslant 1$. ∎

The second main lemma shows that several scalar inequalities related to H, M_s, and $\boldsymbol{\sigma}^{(s)}$ hold with high probability in the SBM. We will prove it in subsequent subsections. The notation is as in Definition 5.5 and Lemma 5.6.

Lemma 6.2 (Estimates for the corrected Bethe Hessian, see Section 6.1)**.**

(a) If $\lambda > 0$, then $\frac{1}{n} b^\top H b$ converges in probability to a negative constant.

(b) There is a constant $g = g(d, \lambda) > 0$ such that for every $\varepsilon > 0$ there is s_0 such that for every $s \geqslant s_0$, with high probability,

$$\left\| (M_s + gI)\boldsymbol{\sigma}^{(s)} \right\| \leqslant \varepsilon \left\| \boldsymbol{\sigma}^{(s)} \right\|.$$

(c) There is $C = C(d, \lambda)$ such that for every s, with high probability

$$\|\boldsymbol{\sigma}^{(s)}\|_4^4 \leqslant \frac{C}{n} \|\boldsymbol{\sigma}^{(s)}\|^4.$$

Now we can prove Lemma 5.6.

Proof of Lemma 5.6. The bound on $\left\| (M_s + gI)\boldsymbol{\sigma}^{(s)} \right\|$ and the delocalization bounds in Lemma 5.6 are Lemma 6.2(b) and (c) respectively. It remains to count negative eigenvalues of M_s. Let g be the constant in Lemma 6.2(b). Fix $\varepsilon > 0$, without loss of generality we can assume $\varepsilon < g/2$. Let s_0 be given by Lemma 6.2(b) applied with $\min\{\varepsilon, g/2\}$ in place of ε. Fix $s > s_0$. Everything below holds on the intersection of the high-probability events in Lemmas 6.1 and 6.2.

First we show that M_s has at most one negative eigenvalue. If $\lambda < 0$ then $M_s = (d\lambda)^{-2s} A^{(s)} H A^{(s)}$. Multiplication on right and left by $A^{(s)}$ cannot increase the number of negative eigenvalues, and by Lemma 6.1 H has at most one negative eigenvalue.

If $\lambda > 0$, then H could have 2 negative eigenvalues. It will be enough to show that the rank-one corrected matrix

$$H' = H + \frac{2(Hb)(Hb)^\top}{-b^\top H b}$$

has at most one negative eigenvalue. This uses only simple linear algebra to show that such a rank-one correction of a matrix with 2 negative eigenvalues leads to one with a single negative eigenvalue. We construct an $(n-1)$-dimensional subspace on which H' is PSD. Let U be the span of the eigenvectors of H with nonnegative eigenvalues, so that $\dim U \geqslant n - 2$ by Lemma 6.1. By

Lemma 6.2(a), $-b^\top Hb > 0$, so $b \notin U$ and $U \oplus \mathrm{span}\{b\}$ has dimension at least $n-1$. For $y = u + \alpha b$ with $u \in U$, expanding and completing the square gives

$$y^\top H' y = u^\top Hu - b^\top Hb \left(\alpha + \frac{b^\top Hu}{b^\top Hb}\right)^2 - \frac{(b^\top Hu)^2}{b^\top Hb} \geqslant 0\,.$$

It remains now to show that M_s also has at least one negative eigenvalue. Let $u = \boldsymbol{\sigma}^{(s)}/\|\boldsymbol{\sigma}^{(s)}\|$. By Lemma 6.2(b), $\|(M_s + gI)u\| \leqslant g/2$, so

$$u^\top M_s u = u^\top (M_s + gI)u - g \leqslant \|(M_s + gI)u\| - g < 0\,. \qquad \blacksquare$$

6.1 Reduction to the broadcast tree

In this section we prove Lemma 6.2 by assembling a number of facts about the broadcasting on trees process. We start with some definitions.

Definition 6.3 (Local statistics)**.** An *R-local statistic* is a real-valued function f of a rooted graph whose vertices carry ± 1 labels such that, for some R, f depends only on the radius-R neighborhood of the root, whose magnitude is always at most $(1+N)^R$, where N is the number of vertices of that neighborhood. For a labeled graph which will be clear from context and a vertex v, we write $f(v)$ for f evaluated on the graph rooted at v.

Local statistics form a vector space – it's easy to check that if f, h are both R-local then $c \cdot f$ is $(R + \log|c|)$-local and $f + h$ is $(R+1)$-local. (This fact depends on using $(1+N)^R$ in the definition rather than the more-natural-looking N^R.) Furthermore, the broadcast process induces an inner product on local statistics. Let $\langle f, h\rangle = \mathbb{E}_{\boldsymbol{T},x}[f(o)h(o)]$ where $(\boldsymbol{T}, x)$ is a labeled broadcast tree with parameters d, λ whose root is o, and let $\|f\|_{L^p} = (\mathbb{E}\,|f(o)|^p)^{1/p}$. We write $\|f\|$ for $\|f\|_{L^2}$.

Mildly abusing notation, for each of the matrices involved in Definition 5.5 we introduce a linear operator on local statistics with the same name. In particular,

$$(A^{(j)}f)(v) = \sum_{(v_0,\ldots,v_j)} f(v_j)\,, \qquad (Df)(v) = \deg(v)\cdot f(v)$$

where the sum is over nonbacktracking walks with $v_0 = v$, and the notation $f(v_j)$ means that f is evaluated on the same underlying labeled graph which contains v, but moving the root to vertex v_j. H is defined analogously. Locality is well behaved under these linear transformations: $A^{(j)}$ maps R-local statistics to $(R+j)$-local statistics, and D maps R-local statistics to $(R+1)$-local statistics.

In particular, continuing to abuse notation by conflating vectors in $\mathbb{R}^n$ and local statistics, $\boldsymbol{\sigma}^{(s)}$ is the local statistic $(d\lambda)^{-s}A^{(s)}x$, and b and Hb are local statistics. Products of local statistics are again local statistics.

We will repeatedly use the following immediate consequence of Fact 4.7, which justifies our abuse of notation – it says that as far as local statistics are concerned, the inner product spaces induced by the SBM graph and by the broadcast tree act the same.

Lemma 6.4 (Transfer to the broadcast tree, corollary of Fact 4.7)**.** *For every $R \in \mathbb{N}$ and every pair of R-local statistics f, h, if $(\boldsymbol{G}, \boldsymbol{\sigma}^*) \sim \mathrm{SBM}(n, d, \lambda)$,*

$$\frac{1}{n} \sum_{v \in [n]} f(v) h(v) \xrightarrow{\boldsymbol{Pr}} \langle f, h \rangle_{\boldsymbol{T}} ,$$

where in the sum v ranges over vertices of $\boldsymbol{G}$.

The following lemma collects the estimates on the broadcast tree that we need; we prove it in the next subsection. Importantly, each of the quantities in the lemma concerns a local statistic.

Lemma 6.5 (Tree estimates, proved in Section 6.2)**.**

(i) If $\lambda > 0$ and L is large enough, then $\langle b, Hb \rangle_{\boldsymbol{T}} < 0$.

(ii) If $\lambda > 0$, then $\langle Hb, A^{(s)} \boldsymbol{\sigma}^{(s)} \rangle_{\boldsymbol{T}} = 0$ for every s.

(iii) For every s, $\|\boldsymbol{\sigma}^{(s)}\|_{L^2} \geqslant 1$, and $\sup_s \|\boldsymbol{\sigma}^{(s)}\|_{L^4} < \infty$.

(iv) There is a constant $g = g(d, \lambda) > 0$ such that, as $s \to \infty$,

$$\left\| (d\lambda)^{-2s} A^{(s)} H A^{(s)} \boldsymbol{\sigma}^{(s)} + g \, \boldsymbol{\sigma}^{(s)} \right\|_{L^2} \to 0 .$$

Proof of Lemma 6.2. Part (a) is immediate from Lemma 6.4 and Lemma 6.5(i). Part (c) is immediate from Lemma 6.4 and Lemma 6.5(iii).

We turn to part (b). Most of the work is to show that the rank-one correction term has negligible interaction with $\boldsymbol{\sigma}^{(s)}$. For notational convenience, let

$$r_s = (d\lambda)^{-2s} A^{(s)} H A^{(s)} \boldsymbol{\sigma}^{(s)} + g \, \boldsymbol{\sigma}^{(s)}$$

be the local statistic in Lemma 6.5(iv). (As usual we sometimes abuse notation and also write r_s for the n-dimensional vector associated to an SBM graph.)

Fix $\varepsilon > 0$. By Lemma 6.5(iv), there is s_0 such that $\|r_s\|_{L^2} \leqslant \varepsilon/4$ for every $s \geqslant s_0$; fix such an s. By Lemma 6.4, $\frac{1}{n}\|r_s\|^2$ converges in probability to a limit at most $\varepsilon^2/16$, so with high probability $\|r_s\| \leqslant \frac{\varepsilon}{3}\sqrt{n} \leqslant \frac{\varepsilon}{2}\|\boldsymbol{\sigma}^{(s)}\|$, using $\|\boldsymbol{\sigma}^{(s)}\|^2 \geqslant n/2$. If $\lambda < 0$, then $M_s = (d\lambda)^{-2s} A^{(s)} H A^{(s)}$, so $(M_s + gI)\boldsymbol{\sigma}^{(s)} = r_s$, so we are done with the $\lambda < 0$ case. If $\lambda > 0$, then

$$(M_s + gI)\boldsymbol{\sigma}^{(s)} = r_s + (d\lambda)^{-2s} A^{(s)} \frac{2(Hb)(Hb)^\top}{-b^\top Hb} A^{(s)} \boldsymbol{\sigma}^{(s)} .$$

The last term is $A^{(s)} Hb$ times the scalar $2(Hb)^\top A^{(s)} \boldsymbol{\sigma}^{(s)} / ((d\lambda)^{2s}(-b^\top Hb))$, so divided by $\sqrt{n}$ it has norm

$$\frac{2\,\|A^{(s)} Hb\|}{(d\lambda)^{2s}\sqrt{n}} \cdot \frac{\left|\frac{1}{n}(Hb)^\top A^{(s)} \boldsymbol{\sigma}^{(s)}\right|}{-\frac{1}{n} b^\top Hb} .$$

By Lemma 6.4, the first factor converges to $2(d\lambda)^{-2s}\|A^{(s)} Hb\|_{L^2} < \infty$, the numerator converges to $\langle Hb, A^{(s)} \boldsymbol{\sigma}^{(s)} \rangle_{\boldsymbol{T}} = 0$ by Lemma 6.5(ii), and the denominator converges to $-\langle b, Hb \rangle_{\boldsymbol{T}} > 0$ by Lemma 6.5(i). So with high probability the last term has norm at most $\frac{\varepsilon}{3}\sqrt{n} \leqslant \frac{\varepsilon}{2}\|\boldsymbol{\sigma}^{(s)}\|$, and the triangle inequality gives (b). ∎

6.2 Proof of the tree estimates

We now prove Lemma 6.5. We prove them roughly in order of difficulty, (ii), (iii), (i), and finally (iv), building up the necessary infrastructure about broadcast trees along the way.

Definition 6.6 (Odd and even local statistics)**.** Call a local statistic *odd* if reversing all labels negates it, and *even* if reversing all labels leaves it unchanged. For example, the root label x is odd, $\mathbf{1}$ is even, and $A^{(j)}$, D, and H preserve parity, since they do not look at labels.

Since the law of the Galton-Watson tree is invariant under flipping the signs of the labels, even and odd local statistics are mutually orthogonal. Now we can discharge the proof of (ii).

Proof of Lemma 6.5(ii). The statistic Hb is even and $A^{(s)}\boldsymbol{\sigma}^{(s)}$ is odd. ∎

Now we assemble a few more tools needed for (iii). For $m \geqslant 0$, let $\mathcal{F}_m$ be the σ-algebra generated by the broadcast tree and its labels up to depth m, and let Z_m be the number of vertices at depth m, so that $\mathbb{E}\, Z_m = d^m$.

Lemma 6.7 (Convergence of $\boldsymbol{\sigma}^{(s)}$)**.** *As $r \to \infty$, $\sup_{s \geqslant r} \|\boldsymbol{\sigma}^{(s)} - \boldsymbol{\sigma}^{(r)}\|_{L^4} \to 0$.*

Proof. Writing $\boldsymbol{\sigma}^{(s)} - \boldsymbol{\sigma}^{(r)} = \sum_{j=r+1}^{s} (\boldsymbol{\sigma}^{(j)} - \boldsymbol{\sigma}^{(j-1)})$ as a telescoping sum, it suffices to show that $\|\boldsymbol{\sigma}^{(j)} - \boldsymbol{\sigma}^{(j-1)}\|_{L^4}$ is summable in j. Given $\mathcal{F}_{j-1}$, each vertex u at depth $j-1$ has independent $\mathrm{Poi}(d(1+\lambda)/2)$ and $\mathrm{Poi}(d(1-\lambda)/2)$ numbers of children with signs $\boldsymbol{x}_u$ and $-\boldsymbol{x}_u$, and expanding $\boldsymbol{\sigma}^{(j)}$ and $\boldsymbol{\sigma}^{(j-1)}$ each evaluated at the root we get

$$(\boldsymbol{\sigma}^{(j)} - \boldsymbol{\sigma}^{(j-1)})(o) = (d\lambda)^{-j} \cdot \sum_{u \text{ at depth } j-1} \Big(\sum_{c \text{ child of } u} \boldsymbol{x}_c - d\lambda \boldsymbol{x}_u \Big).$$

Given $\mathcal{F}_{j-1}$, the terms in the sum $\sum_{u \text{ at depth } j-1}(\cdots)$ are independent and mean zero, and the terms in the sum all have the same fourth moment, which depends only on d, λ and is independent of j. There are Z_{j-1} terms in the sum. So

$$\mathbb{E}[(\boldsymbol{\sigma}^{(j)} - \boldsymbol{\sigma}^{(j-1)})(o)^4 \mid \mathcal{F}_{j-1}] \leqslant (d\lambda)^{-4j} \cdot O_{d,\lambda}(Z_{j-1}^2).$$

Since $\mathbb{E}\, Z_m^2 \leqslant O_d(d^{2m})$, taking expectations,

$$\|\boldsymbol{\sigma}^{(j)} - \boldsymbol{\sigma}^{(j-1)}\|_{L^4}^4 \leqslant (d\lambda)^{-4j} \cdot O_{d,\lambda}(d^{2j}) = O_{d,\lambda}\big((d\lambda^2)^{-2j}\big),$$

so $\|\boldsymbol{\sigma}^{(j)} - \boldsymbol{\sigma}^{(j-1)}\|_{L^4} = O_{d,\lambda}((d\lambda^2)^{-j/2})$, which is summable in j since $d\lambda^2 > 1$. ∎

Now we can prove Lemma 6.5(iii).

Proof of Lemma 6.5(iii). A quick calculation shows $\langle x, \boldsymbol{\sigma}^{(s)} \rangle = 1$, and since $\|x\| = 1$, by Cauchy-Schwarz we must have $\|\boldsymbol{\sigma}^{(s)}\| \geqslant 1$. The fourth moment bound is immediate from Lemma 6.7. ∎

Next we prove Lemma 6.5(i), using a similar telescoping-sum strategy.

The only computation needed is the following truncation of the generating function of nonbacktracking walks. It is an identity of matrices on any graph, and of operators on local statistics.

Fact 6.8 (Nonbacktracking generating function)**.** *For every $m \geqslant 1$,*

$$H \sum_{j=0}^{m} t^j A^{(j)} = (1-t^2)I - t^{m+1}A^{(m+1)} + t^{m+2}(D-I)A^{(m)}\,.$$

Proof. Expand $H = I - tA + t^2(D-I)$ and apply $AA^{(j)} = A^{(j+1)} + (D-I)A^{(j-1)}$ for $j \geqslant 2$ and $AA^{(1)} = A^{(2)} + (D-I) + I$. All terms other than the displayed ones cancel in pairs. ∎

Proof of Lemma 6.5(i). By definition, $b = F_L - dtF_{L-1}$, where $F_m = \sum_{j=0}^{m} t^j A^{(j)}\mathbf{1}$. Writing $\Delta_j = A^{(j)}\mathbf{1} - dA^{(j-1)}\mathbf{1}$, Fact 6.8, applied with $m = L$ and $m = L-1 \geqslant 1$, gives

$$\begin{aligned} Hb &= HF_L - dt\,HF_{L-1} \\ &= (1-t^2)\mathbf{1} - t^{L+1}A^{(L+1)}\mathbf{1} + t^{L+2}(D-I)A^{(L)}\mathbf{1} \\ &\qquad - dt\Big((1-t^2)\mathbf{1} - t^L A^{(L)}\mathbf{1} + t^{L+1}(D-I)A^{(L-1)}\mathbf{1}\Big) \\ &= (1-t^2)(1-dt)\mathbf{1} - t^{L+1}\Delta_{L+1} + t^{L+2}(D-I)\Delta_L\,. \end{aligned}$$

Using the same telescoping expansion as in the proof of (iii) applied to Δ_j shows $\|\Delta_j\| \leqslant O(d^{j/2})$ and $\|(D-I)\Delta_j\| \leqslant O(d^{j/2})$. Hence we have

$$Hb = (1-t^2)(1-dt)\mathbf{1} - O_d((dt^2)^{L/2})\,.$$

Furthermore, $b = \mathbf{1} + \sum_{j=1}^{L} t^j\Delta_j$ has $\langle b, \mathbf{1}\rangle = 1$ and $\|b\|^2 = \sum_{j=0}^{L}(dt^2)^j = O(1)$ where we used $dt^2 < 1$. Hence, $\langle b, Hb\rangle = (1-t^2)(1-dt) + O((dt^2)^{L/2})$, which is negative for L large enough, since $dt^2 < 1 < dt$. ∎

We turn to the proof of (iv), which is a bit more involved. Write $C^{(j)} = (d\lambda)^{-j}A^{(j)}$, so that $\boldsymbol{\sigma}^{(j)} = C^{(j)}x$. The key ingredient for (iv) is the following lemma.

Lemma 6.9 ($\boldsymbol{\sigma}^{(s)}$ is an approximate eigenvector of $C^{(s)}$)**.** *Let $c = d\lambda^2/(d\lambda^2-1)$. As $s \to \infty$, for each $s' \in \{s-1, s, s+1\}$,*

(a) $\|C^{(s)}\boldsymbol{\sigma}^{(s')} - c\,\boldsymbol{\sigma}^{(s)}\| \to 0$,

(b) $\|C^{(s)}(D-I)\boldsymbol{\sigma}^{(s')} - (dc+1)\,\boldsymbol{\sigma}^{(s)}\| \to 0$, *and*

(c) $\|C^{(s)}H(C^{(s)}\boldsymbol{\sigma}^{(s)} - c\,\boldsymbol{\sigma}^{(s)})\| \to 0$.

The basic idea of this lemma is simple: once you are summing up the signs at depth s, there isn't much additional information in signs at depth $2s$. The main difficulty in proving Lemma 6.9 is that as $s \to \infty$ both $C^{(s)}$ and $\boldsymbol{\sigma}^{(s)}$ "see" more of the tree at the same time; if instead we only needed to analyze $C^{(s)}\boldsymbol{\sigma}^{(a)}$ for a fixed a as $s \to \infty$ it would be much easier. We prove Lemma 6.9 in Appendix A, and turn to the proof of Lemma 6.5(iv).

Proof of Lemma 6.5(iv). By Lemma 6.9(c),

$$\big\|C^{(s)}HC^{(s)}\boldsymbol{\sigma}^{(s)} - c\,C^{(s)}H\boldsymbol{\sigma}^{(s)}\big\| \to 0\,,$$

as $s \to \infty$, so it suffices to show that $C^{(s)} H \sigma^{(s)}$ is close to a negative multiple of $\sigma^{(s)}$. For $s \geqslant 2$, the recurrence $A A^{(s)} = A^{(s+1)} + (D - I) A^{(s-1)}$ gives $A\sigma^{(s)} = d\lambda\, \sigma^{(s+1)} + (d\lambda)^{-1}(D - I)\sigma^{(s-1)}$, so

$$C^{(s)} H \sigma^{(s)} = C^{(s)} \sigma^{(s)} - d\lambda t\, C^{(s)} \sigma^{(s+1)} - \frac{t}{d\lambda}\, C^{(s)} (D - I) \sigma^{(s-1)} + t^2\, C^{(s)} (D - I) \sigma^{(s)}\,.$$

By Lemma 6.9(a), the first two terms on the right are close to $c\,\sigma^{(s)}$ and $-d\lambda t\, c\, \sigma^{(s)}$, and by Lemma 6.9(b), the last two are close to $-\frac{t}{d\lambda}(dc+1)\,\sigma^{(s)}$ and $t^2(dc+1)\,\sigma^{(s)}$. Adding these, $\|C^{(s)} H \sigma^{(s)} - q\,\sigma^{(s)}\| \to 0$, where

$$q = (1 - d\lambda t)\, c + \Big(t^2 - \frac{t}{d\lambda}\Big)(dc + 1) = (1 - d\lambda t)\Big(c - \frac{t(dc+1)}{d\lambda}\Big)\,.$$

Hence $\|C^{(s)} H C^{(s)} \sigma^{(s)} + g\,\sigma^{(s)}\| \to 0$ with $g = -c\,q$, and $C^{(s)} H C^{(s)} = (d\lambda)^{-2s} A^{(s)} H A^{(s)}$. Finally, the choice of t in Definition 5.5 ensures that $d\lambda t > 1$ and $t(dc+1)/(d\lambda) < c$, so $q < 0$ and $g > 0$. ■

7 Analysis of degree-cutoff preprocessing

Algorithm (Degree-Cutoff Preprocessing). Choose a sufficiently large fixed integer k. Delete every edge incident to a vertex whose degree in F exceeds k, retaining all vertices. Let the degree truncated graph be F_k and return $\widetilde{M} = M_s(F_k)$.

Since all the statistics of $M_s(\cdot)$ can be computed by fixed-radius neighborhoods, the *Degree-Cutoff Preprocessing* algorithm can be implemented in linear time. Now, we show that the output $\widetilde{M}$ of *Degree-Cutoff Preprocessing* satisfies Lemma 5.8.

Lemma 5.8 (Preprocessing for robustness). *Let $(\boldsymbol{G}, \sigma^\star) \sim \mathrm{SBM}(n, d, \lambda)$ and $d\lambda^2 > 1$. For every $s \in \mathbb{N}$ and $\varepsilon > 0$, there is a linear-time algorithm, $C \geqslant 1$ and $\eta > 0$ which with high probability over $\boldsymbol{G}$, given as input any n-vertex graph F such that $|E(F) \triangle E(\boldsymbol{G})| \leqslant \eta n$, produces a symmetric matrix $\widetilde{M} \in \mathbb{R}^{n \times n}$ such that*

$$\|\widetilde{M} - M_s(\boldsymbol{G})\|_{1,2} \leqslant \varepsilon\sqrt{n}, \qquad \|\widetilde{M}\|_{1,\infty} \leqslant C\,.$$

Moreover, matrix-vector products by $\widetilde{M}$ can be computed in linear time.

Notation. In this section, we use subscripts to denote the graph on which the matrix is computed, e.g., $A^{(s)}_{\boldsymbol{G}}$ and $A^{(s)}_{F_k}$ denote the length-s nonbacktracking adjacency matrix of $\boldsymbol{G}$ and F_k respectively. For notational simplicity, we define, for any graph G,

$$C^{(s)}_G := (d\lambda)^{-s} A^{(s)}_G, \qquad a_G := -b_G^\top H_G b_G, \qquad h_G := C^{(s)}_G H_G b_G.$$

Proof of Lemma 5.8. Let $\boldsymbol{G}_k$ be the degree-truncated graph of $\boldsymbol{G}$. By the triangle inequality,

$$\begin{aligned}\|\widetilde{M} - M_s(\boldsymbol{G})\|_{1,2} \leqslant{}& \|M_s(F_k) - M_s(\boldsymbol{G}_k)\|_{1,2} \\ &+ \|M_s(\boldsymbol{G}_k) - M_s(\boldsymbol{G})\|_{1,2}.\end{aligned} \tag{5}$$

The first term is the edge corruption error and the second is the cutoff error. We bound them separately.

Edge corruption error. Let S be the endpoints of edited edges. Notice that ηn edge edits can change the degrees of at most $2\eta n$ vertices. Only these vertices can change cutoff status between F_k and $\boldsymbol{G}_k$ and each contributes at most k edge differences. Hence,

$$|E(F_k) \triangle E(\boldsymbol{G}_k)| \leqslant (2k+1)\eta n.$$

Since both F_k and $\boldsymbol{G}_k$ have maximum degree k, each vertex starts at most k^s nonbacktracking walks of length s. Moreover, each row of the Bethe Hessian has at most k off-diagonal entries of magnitude $|t|$ and diagonal entries of magnitude at most $1 + t^2(k+1)$. Therefore,

$$\|C_{F_k}^{(s)}\|_{1,\infty},\ \|C_{\boldsymbol{G}_k}^{(s)}\|_{1,\infty} \leqslant |d\lambda|^{-s}k^s, \qquad \|H_{F_k}\|_{1,\infty},\ \|H_{\boldsymbol{G}_k}\|_{1,\infty} \leqslant 1 + |t|k + t^2(k+1).$$

It follows that

$$\|C_{F_k}^{(s)} H_{F_k} C_{F_k}^{(s)}\|_{1,\infty} \leqslant \|C_{F_k}^{(s)}\|_{1,\infty}^2 \|H_{F_k}\|_{1,\infty} = O(1).$$

Observe that the (u,v) entry of the matrix $C_G^{(s)} H_G C_G^{(s)}$ is a normalized weighted sum over pairs of length-s nonbacktracking walks in G, one starting at u and one at v, whose endpoints coincide or are adjacent. Therefore, only rows within distance $2s+1$ of S can differ between $C_{F_k}^{(s)} H_{F_k} C_{F_k}^{(s)}$ and $C_{\boldsymbol{G}_k}^{(s)} H_{\boldsymbol{G}_k} C_{\boldsymbol{G}_k}^{(s)}$. Hence, the row-sum bounds give

$$\|C_{F_k}^{(s)} H_{F_k} C_{F_k}^{(s)} - C_{\boldsymbol{G}_k}^{(s)} H_{\boldsymbol{G}_k} C_{\boldsymbol{G}_k}^{(s)}\|_{1,2} = O(\sqrt{\eta n}).$$

This concludes the bound for $\lambda < 0$. For $\lambda > 0$, we need to bound the error in the rank-one corrections. Recall that, by the definition of $b_{\boldsymbol{G}_k}$, it is an L-neighbourhood local statistic. Since the degrees are bounded by k and L, s are fixed constants, the entries of h_{F_k} and $h_{\boldsymbol{G}_k}$ are bounded. The differences in h between the two graphs are supported on distance $L+s+1$ neighbourhood of vertices in S, and, hence, the support of $h_{F_k} - h_{\boldsymbol{G}_k}$ is of size at most $O((2k)^{L+s+1}\eta n)$. Thus,

$$\|h_{F_k}\|_\infty, \|h_{\boldsymbol{G}_k}\|_\infty = O(1), \qquad \|h_{F_k} - h_{\boldsymbol{G}_k}\|_1 = O(\eta n), \qquad \|h_{F_k} - h_{\boldsymbol{G}_k}\|_2 = O(\sqrt{\eta n}).$$

Hence,

$$\begin{aligned}
\|h_{F_k} h_{F_k}^\top - h_{\boldsymbol{G}_k} h_{\boldsymbol{G}_k}^\top\|_{1,2} &\leqslant \|(h_{F_k} - h_{\boldsymbol{G}_k}) h_{F_k}^\top\|_{1,2} + \|h_{\boldsymbol{G}_k}(h_{F_k} - h_{\boldsymbol{G}_k})^\top\|_{1,2} \\
&\leqslant \|h_{F_k} - h_{\boldsymbol{G}_k}\|_2 \|h_{F_k}\|_1 + \|h_{\boldsymbol{G}_k}\|_2 \|h_{F_k} - h_{\boldsymbol{G}_k}\|_1 \\
&= O\Big((\eta + \sqrt{\eta}) n^{3/2}\Big) = O\Big(\sqrt{\eta} n^{3/2}\Big).
\end{aligned}$$

Following similar local analysis, we can show that

$$|a_{F_k} - a_{\boldsymbol{G}_k}| = O(\eta n).$$

By Lemma 6.2, we have $a_{\boldsymbol{G}} = \Omega(n)$ with high probability. Taking a large enough k and using similar local analysis on $a_{\boldsymbol{G}_k}$ yields the lower bound $a_{\boldsymbol{G}_k} = \Omega(n)$. Therefore, we can get $a_{F_k}, a_{\boldsymbol{G}_k} = \Omega(n)$ with high probability. Now, we can bound the rank-one correction differences as

$$\left\| \frac{2h_{F_k} h_{F_k}^\top}{a_{F_k}} - \frac{2h_{\boldsymbol{G}_k} h_{\boldsymbol{G}_k}^\top}{a_{\boldsymbol{G}_k}} \right\|_{1,2} = \left\| \frac{2h_{F_k} h_{F_k}^\top}{a_{F_k}} - \frac{2h_{\boldsymbol{G}_k} h_{\boldsymbol{G}_k}^\top}{a_{F_k}} + \frac{2h_{\boldsymbol{G}_k} h_{\boldsymbol{G}_k}^\top}{a_{F_k}} - \frac{2h_{\boldsymbol{G}_k} h_{\boldsymbol{G}_k}^\top}{a_{\boldsymbol{G}_k}} \right\|_{1,2}$$

$$\leqslant \frac{2}{a_{F_k}} \left\| h_{F_k} h_{F_k}^\top - h_{G_k} h_{G_k}^\top \right\|_{1,2} + \frac{2|a_{F_k} - a_{G_k}|}{a_{F_k} a_{G_k}} \|h_{G_k}\|_1 \|h_{G_k}\|_2$$
$$= O\big((\eta + \sqrt{\eta})\sqrt{n}\big) = O(\sqrt{\eta n})\,.$$

Thus, by picking η small enough, we can get

$$\begin{aligned} \|\widetilde{M} - M_s(\boldsymbol{G}_k)\|_{1,2} &\leqslant \|C_{F_k}^{(s)} H_{F_k} C_{F_k}^{(s)} - C_{G_k}^{(s)} H_{G_k} C_{G_k}^{(s)}\|_{1,2} \\ &\quad + \left\| \frac{2h_{F_k} h_{F_k}^\top}{a_{F_k}} - \frac{2h_{G_k} h_{G_k}^\top}{a_{G_k}} \right\|_{1,2} \\ &= O(\sqrt{\eta n}) \leqslant \frac{\varepsilon}{2}\sqrt{n}, \end{aligned} \tag{6}$$

and,

$$\|\widetilde{M}\|_{1,\infty} \leqslant \|C_{F_k}^{(s)} H_{F_k} C_{F_k}^{(s)}\|_{1,\infty} + \frac{2\|h_{F_k}\|_\infty \|h_{F_k}\|_1}{a_{F_k}} = O(1). \tag{7}$$

Cutoff error. Let $r_k(i)$ denote the absolute row sum for row i of $C_{G_k}^{(s)} H_{G_k} C_{G_k}^{(s)} - C_{\boldsymbol{G}}^{(s)} H_{\boldsymbol{G}} C_{\boldsymbol{G}}^{(s)}$. Define $r_k(o)$ analogously for the Poisson tree. Since the row sums $r_k(i)$ are local statistics of radius $2s+2$, the walk-counting bound on the Poisson tree gives

$$r_k(o)^2 \leqslant C^2 (1 + |B_{2s+2}(o)|)^{4s+2} \mathbf{1}[\exists v \in B_{2s+1}(o) : \deg(v) > k]\,,$$

where $B_r(o)$ is the radius r neighbourhood of o and C is a constant independent of k. The indicator tends to zero almost surely as $k \to \infty$. Since fixed-radius Poisson-tree neighborhoods have moments of every order, by dominated convergence, we have $\mathbb{E}[r_k(o)^2] \to 0$. Hence, by the local law Fact 4.7, it follows that

$$\frac{1}{n} \left\| C_{G_k}^{(s)} H_{G_k} C_{G_k}^{(s)} - C_{\boldsymbol{G}}^{(s)} H_{\boldsymbol{G}} C_{\boldsymbol{G}}^{(s)} \right\|_{1,2}^2 = \frac{1}{n} \sum_i r_k(i)^2 \xrightarrow{Pr} \mathbb{E}[r_k(o)^2].$$

Consequently, for every $\delta > 0$ and every sufficiently large fixed k, with high probability,

$$\|C_{G_k}^{(s)} H_{G_k} C_{G_k}^{(s)} - C_{\boldsymbol{G}}^{(s)} H_{\boldsymbol{G}} C_{\boldsymbol{G}}^{(s)}\|_{1,2} \leqslant \delta\sqrt{n}.$$

This concludes the cutoff bound for $\lambda < 0$. For $\lambda > 0$, we can apply the same argument to bound the squared coordinate differences of h_{G_k} and $h_{\boldsymbol{G}}$ and $|a_{G_k} - a_{\boldsymbol{G}}|$. Therefore, using the same analysis as in *edge corruption error*, it follows that, for every $\delta > 0$ and every sufficiently large fixed k, with high probability,

$$\left\| \frac{2h_{\boldsymbol{G}} h_{\boldsymbol{G}}^\top}{a_{\boldsymbol{G}}} - \frac{2h_{G_k} h_{G_k}^\top}{a_{G_k}} \right\|_{1,2} \leqslant \delta\sqrt{n}.$$

Combining both parts and choosing δ sufficiently small yields

$$\|M_s(\boldsymbol{G}_k) - M_s(\boldsymbol{G})\|_{1,2} \leqslant \frac{\varepsilon}{2}\sqrt{n}. \tag{8}$$

Conclusion Plugging (6) and (8) into (5), it follows that, with high probability,

$$\|\widetilde{M} - M_s(\boldsymbol{G})\|_{1,2} \leqslant \varepsilon\sqrt{n}.$$

Combining this with the bound for $\|\widetilde{M}\|_{1,\infty}$ in (7) finishes the proof. ∎

8 SDP recovery and implementation

Let n be an integer with $n \geqslant 2$. For $K \geqslant 1$, define

$$\mathcal{S} \coloneqq \{X \in \mathbb{R}^{n\times n} : X \succeq 0,\ \operatorname{Tr} X = 1\}, \qquad \mathcal{D}_K \coloneqq \{q \in \mathbb{R}^n : \|q\|_2^2 \leqslant K/n\}.$$

For matrices E, write $\|E\|_{1,p} \coloneqq (\sum_i (\sum_j |E_{ij}|)^p)^{1/p}$. For matrices $B, C \in \mathbb{R}^{n\times n}$, write $\langle B, C\rangle = \operatorname{Tr}(B^T C)$. For $p \in \mathbb{R}^n$ and $D \subseteq \mathbb{R}^n$, write $\operatorname{dist}_1(p, D) = \inf_{q\in D} \|p - q\|_1$.

The following theorem shows that almost optimal solutions of a particular convex optimization problem over density matrices $\mathcal{S}$ are close to the projector to our target vector's span. Its proof in Section 8.1 combines ideas from earlier recovery algorithms for stochastic block models [GV16] and robust principal component analysis [CLMW11].

Theorem 8.1 (Matrix recovery)**.** *Let $K \geqslant 1$, $g > 0$, and $\varepsilon \geqslant 0$. Suppose $M, E \in \mathbb{R}^{n\times n}$ are symmetric and $v, w \in \mathbb{R}^n$ are unit vectors satisfying*

$$M \succeq -gvv^T, \qquad n\|w\|_4^4 \leqslant K, \qquad w^T M w \leqslant -(1-\varepsilon)\cdot g.$$

Set $A = M + E$ and let $P \geqslant \|A\|_{1,\infty}$. Define $F(X) = \langle A, X\rangle + P \cdot \operatorname{dist}_1(\operatorname{diag} X, \mathcal{D}_K)$. Then

$$\left|\tfrac{1}{g}\min_{\mathcal{S}} F + 1\right| \leqslant \varepsilon + \sqrt{K/n}\|E\|_{1,2}/g,$$

and every $X \in \mathcal{S}$ with $F(X) \leqslant \min_{\mathcal{S}} F + \varepsilon g$ satisfies

$$\|X - ww^T\|_F^2 \leqslant 20\left(\varepsilon + \sqrt{K/n}\|E\|_{1,2}/g\right).$$

The next theorem shows that the convex program from the previous theorem statement admits nearly-linear time algorithms assuming that the input matrix supports nearly-linear time matrix-vector operations. Its proof in Section 8.2 adapts ideas from [AK07].

Theorem 8.2 (Fast optimization)**.** *Given $K \geqslant 1$, $0 < \delta \leqslant P$, a symmetric matrix $A \in \mathbb{R}^{n\times n}$ through a matrix-vector multiplication oracle, and a bound $P \geqslant \|A\|_{1,\infty} + 1$, a randomized algorithm computes $\boldsymbol{Z} \in \mathbb{R}^{n\times r}$ such that*

$$\boldsymbol{Z}\boldsymbol{Z}^T \in \mathcal{S}, \qquad \mathbb{E}F(\boldsymbol{Z}\boldsymbol{Z}^T) \leqslant \min_{\mathcal{S}} F + \delta, \qquad r \leqslant O(1 + P^4 \log n/\delta^4),$$

where $F(X) = \langle A, X\rangle + P \cdot \operatorname{dist}_1(\operatorname{diag} X, \mathcal{D}_K)$. The algorithm uses $O((1 + P/\delta)^5 (\log n)^2)$ matrix-vector multiplications with A and at most n times as many additional operations.

Together the above theorems imply the main algorithmic lemma from Section 5.

Lemma 5.9 (Robust recovery of a delocalized negative eigenvector)**.** *There is a randomized algorithm with the following properties. Let $\varepsilon, g > 0$ and $M, \widetilde{M} \in \mathbb{R}^{n\times n}$ be symmetric, with M having exactly one negative eigenvalue $-g$. Suppose there exists a unit vector $w \in \mathbb{R}^n$ satisfying*

$$\|w\|_4^4 \leqslant \frac{1}{\varepsilon n} \quad \text{and} \quad w^\top M w \leqslant -(1-\varepsilon)g\,.$$

Suppose also that $\|\widetilde{M} - M\|_{1,2} \leqslant \varepsilon g\sqrt{n}$. Then, given matrix-vector access to $\widetilde{M}$, an upper bound $C \geqslant \|\widetilde{M}\|_{1,\infty} + 1$, and ε (but not g) as inputs, the algorithm returns a unit vector $\hat{\boldsymbol{w}}$ such that with probability at least $1 - n^{-10}$ over the internal randomness of the algorithm,

$$\langle \hat{\boldsymbol{w}}, w\rangle^2 \geqslant 1 - O(\sqrt{\varepsilon}).$$

On the same event, the algorithm performs at most $\left(1 + \frac{C\log n}{\varepsilon g}\right)^{O(1)}$ matrix-vector products with $\widetilde{M}$ and at most n times as many additional operations.

Proof. Assume $n \geqslant 2$ and ε is sufficiently small; otherwise the conclusion is trivial. Set $A = \widetilde{M}$, $P = C$, and $K = 1/\varepsilon$. Since $\sqrt{K/n}\|A - M\|_{1,2} \leqslant \sqrt{\varepsilon}\, g$, Theorem 8.1 gives $F^* := \min_{\mathcal{S}} F \in [-2g, -g/2]$, so $g/2 \leqslant |F^*| \leqslant 2g$.

At stage $j \in \{0, 1, \ldots\}$, run the fast solver from Theorem 8.2 with $\delta_j = C2^{-j}$. Retain the best output X_j among $\lceil \log_2(1/\rho_j)\rceil$ independent runs, where $\rho_j = n^{-10}/(10j+10)^2$. Markov's inequality gives, simultaneously at all stages except with probability $\sum_j \rho_j < n^{-10}/4$,

$$0 \leqslant F(X_j) - F^* \leqslant 2\delta_j.$$

Stop when $4\delta_j \leqslant \varepsilon(-F(X_j))$.[5] This condition holds once $\delta_j \leqslant \varepsilon|F^*|/8$, and at stopping

$$F(X_j) - F^* \leqslant 2\delta_j \leqslant \varepsilon|F^*|/2 \leqslant \varepsilon g.$$

Hence the recovery theorem gives $\|X_j - ww^T\|_F^2 = O(\sqrt{\varepsilon})$.

Write $X_j = \lambda uu^T + R$, where u is its leading unit eigenvector. Then $\lambda \geqslant 1/2$, $\|R\|_F^2 = O(\sqrt{\varepsilon})$, and $1 - \langle u, w\rangle^2 = O(\sqrt{\varepsilon})$. For a uniform random sign vector ξ,

$$\boldsymbol{Pr}\left[|\langle u, \xi\rangle| \geqslant 1/\sqrt{2}\right] \geqslant 1/12, \qquad \mathbb{E}\|R\xi\|^2 = \|R\|_F^2.$$

Thus, with constant probability, normalizing $X_j\xi$ gives squared correlation $1 - O(\sqrt{\varepsilon})$ with u. Repeat $O(\log n)$ times with independent randomness and retain the normalized nonzero vector with largest Rayleigh quotient for X_j. In this way, the selected vector has squared correlation $1 - O(\sqrt{\varepsilon})$ with u, except with probability $n^{-10}/2$, and therefore also squared correlation $1 - O(\sqrt{\varepsilon})$ with w as claimed.

The search uses $O(1 + \log(C/(\varepsilon g)))$ stages, all with $\delta_j = \Omega(\varepsilon g)$. The fast-solver invocations, objective evaluations, and the matrix-vector products $X_j\xi = Z_j(Z_j^T\xi)$ therefore are within the desired bounds on the number of matrix-vector products with $\widetilde{M}$ and the number of additional operations. The total failure probability is at most n^{-10}. ∎

[5] In order to check this condition, we need to evaluate $F(X_j)$. The linear part of F corresponds to a small number of matrix-vector products with $\widetilde{M}$. Moreover, we can compute the ℓ_1 distance of a vector to a centered Euclidean ball in time $O(n\log n)$.

8.1 Recovery analysis

To prove Theorem 8.1, we show that an εg-approximate minimizer $X \in \mathcal{S}$ of the penalized objective F is close to ww^T in Frobenius norm. Here w is the unit witness vector from the theorem, and E is the perturbation in $A = M + E$. Recall that $M \succeq -gvv^T$, $w^T M w \leqslant -g + \varepsilon g$, and $F(X) = \langle A, X\rangle + P \operatorname{dist}_1(\operatorname{diag} X, \mathcal{D}_K)$ with $A = M + E$ and $P \geqslant \|A\|_{1,\infty}$.

Proof of Theorem 8.1. Put $r = \sqrt{K/n}$ and $b = r\|E\|_{1,2}$. For every $U \succeq 0$, symmetry and $|U_{ij}| \leqslant (U_{ii} + U_{jj})/2$ give

$$|\langle E, U\rangle| \leqslant \sum_i U_{ii} \sum_j |E_{ij}| \leqslant \|\operatorname{diag} U\|_2 \|E\|_{1,2}.$$

Let $X \in \mathcal{S}$ and write $d = \operatorname{diag} X$. If $\|d\|_2 \leqslant r$, set $q = d$. Otherwise choose $t > 0$ such that $q_i = \min(d_i, t)$ satisfies $\|q\|_2 = r$. Then

$$h := \sum_i (d_i - q_i) = \operatorname{dist}_1(d, \mathcal{D}_K).$$

Indeed, in the latter case $y = q/t$ satisfies $\|y\|_\infty \leqslant 1$, $\langle y, d - q\rangle = h$, and $\langle y, z\rangle \leqslant \langle y, q\rangle$ for every $z \in \mathcal{D}_K$. Hence $\|d - z\|_1 \geqslant h$.

Let $D_{ii} = \sqrt{q_i/d_i}$ when $d_i > 0$, and $D_{ii} = 1$ otherwise, and put $Y = DXD$. Then $Y \succeq 0$, $\operatorname{diag} Y = q \in \mathcal{D}_K$, and $\operatorname{Tr} Y \leqslant 1$. Moreover,

$$|X_{ij} - Y_{ij}| \leqslant \sqrt{d_i d_j} - \sqrt{q_i q_j} \leqslant \frac{d_i - q_i + d_j - q_j}{2}.$$

The last inequality follows from the AM-GM inequality when both coordinates are clipped; when only d_i is clipped, use $\sqrt{d_j} \leqslant \sqrt{t}$ and $2\sqrt{t}(\sqrt{d_i} - \sqrt{t}) \leqslant d_i - t$. By symmetry of A and the entrywise bound,

$$|\langle A, X - Y\rangle| \leqslant \sum_i (d_i - q_i) \sum_j |A_{ij}| \leqslant Ph.$$

Thus the penalty pays for clipping:

$$F(X) = \langle A, X\rangle + Ph \geqslant \langle A, Y\rangle.$$

Since $Y \succeq 0$, $M \succeq -gvv^T$, and $\|\operatorname{diag} Y\|_2 \leqslant r$,

$$\langle A, Y\rangle \geqslant -gv^T Y v - |\langle E, Y\rangle| \geqslant -gv^T Y v - b \geqslant -g - b.$$

where the last inequality uses $v^T Y v \leqslant \operatorname{Tr} Y \leqslant 1$. On the other hand, $\operatorname{diag}(ww^T) \in \mathcal{D}_K$, so

$$\min_{\mathcal{S}} F \leqslant F(ww^T) \leqslant -(1 - \varepsilon)g + b.$$

Together these prove the first assertion.

Now suppose $F(X) \leqslant \min_{\mathcal{S}} F + \varepsilon g$. The preceding bounds imply

$$-gv^T Y v - b \leqslant F(X) \leqslant -g + 2\varepsilon g + b, \qquad 1 - v^T Y v \leqslant 2\varepsilon + 2b/g.$$

It remains to transfer alignment from $Y = DXD$ to X. Put $C = 2D - I$, so $\|C\|_{\mathrm{op}} \leqslant 1$. Since $0 \preceq X \preceq I$,

$$v^T Y v = \tfrac{1}{4}\|X^{1/2}v + X^{1/2}Cv\|_2^2 \leqslant \tfrac{1}{2}v^T X v + \tfrac{1}{2}v^T CXCv \leqslant \tfrac{1}{2}v^T X v + \tfrac{1}{2}.$$

Therefore, using $\mathrm{Tr}(X^2) \leqslant 1$,

$$\|X - vv^T\|_F^2 = \mathrm{Tr}(X^2) + 1 - 2v^T X v \leqslant 4(1 - v^T Y v) \leqslant 8\varepsilon + 8b/g.$$

The witness assumption also gives $(v^T w)^2 \geqslant 1 - \varepsilon$, hence $\|vv^T - ww^T\|_F^2 \leqslant 2\varepsilon$. The squared triangle inequality now yields

$$\|X - ww^T\|_F^2 \leqslant 2\|X - vv^T\|_F^2 + 2\|vv^T - ww^T\|_F^2 \leqslant 20\varepsilon + 16b/g. \qquad \blacksquare$$

8.2 Fast implementation

For a convex function F on all real symmetric matrices and a real symmetric matrix X, let $\partial F(X)$ denote the set of symmetric matrices H satisfying $F(U) \geqslant F(X) + \langle H, U - X\rangle$ for every real symmetric matrix U.

The following theorem is the starting point for our fast implementation. At iteration t, we compute $\widetilde{X}_t \in \mathcal{S}$ and $H_t \in \partial F(\widetilde{X}_t)$. We will work with a low-rank representation of $\widetilde{X}_t$ in order to be able to efficiently compute H_t. The theorem shows that if $\widetilde{X}_t$ approximates in direction H_t a particular matrix X_t, defined as a function of $H_1 + \cdots + H_{t-1}$, then the average of the computed iterates approximately minimizes F over $\mathcal{S}$. Its proof in Appendix B is adapted from [AHK12].

Theorem 8.3 (Nearby-point entropy descent)**.** *Let F be convex on all real symmetric matrices and attain its minimum over $\mathcal{S}$. Let $\eta > 0$, let $T \geqslant 1$ be an integer, and let $H_1, \ldots, H_T \in \mathbb{R}^{n\times n}$ be symmetric matrices. For $1 \leqslant t \leqslant T$, define*

$$S_t = -\eta \sum_{s<t} H_s, \qquad X_t = \frac{e^{S_t}}{\mathrm{Tr}\, e^{S_t}}.$$

Suppose $\widetilde{X}_1, \ldots, \widetilde{X}_T \in \mathcal{S}$ satisfy $H_t \in \partial F(\widetilde{X}_t)$ for $1 \leqslant t \leqslant T$. Then $\bar{X} = T^{-1}\sum_{t=1}^T \widetilde{X}_t$ satisfies

$$F(\bar{X}) - \min_{\mathcal{S}} F \leqslant \frac{\log n}{\eta T} + 2\eta \max_{1\leqslant t\leqslant T} \|H_t\|_{\mathrm{op}}^2 + \frac{1}{T}\sum_{t=1}^T \langle H_t, \widetilde{X}_t - X_t\rangle.$$

Recall that $F(X) = \langle A, X\rangle + P\,\mathrm{dist}_1(\mathrm{diag}\, X, \mathcal{D}_K)$. The linear part of F contributes the constant subgradient A. For the ℓ_1-distance term to $\mathcal{D}_K$, the following fact bounds the entries of subgradients of the distance function.

Fact 8.4 (Distance to a convex set)**.** *Let $C \subseteq \mathbb{R}^n$ be nonempty and convex. The function*

$$f_C(x) = \inf_{q\in C} \|x - q\|_1 \qquad (x \in \mathbb{R}^n)$$

is finite, convex [BV04, Example 3.16, p. 88], and 1-Lipschitz in ℓ_1. At every $x \in \mathbb{R}^n$, its subdifferential satisfies $\varnothing \neq \partial f_C(x) \subseteq [-1, 1]^n$.

The above fact directly gives us the form of the subgradients of our objective function F, recorded in the following lemma.

Lemma 8.5 (Subgradients of the penalized objective)**.** *Let $K \geqslant 1$, $P \geqslant 0$, and $A \in \mathbb{R}^{n\times n}$ be symmetric. Let $F(X) = \langle A, X\rangle + P\,\mathrm{dist}_1(\mathrm{diag}\, X, \mathcal{D}_K)$ on all real symmetric matrices. Every subgradient H of F at a point of $\mathcal{S}$ has the form*

$$H = A + \mathrm{Diag}(z), \qquad z \in \mathbb{R}^n, \quad \|z\|_\infty \leqslant P.$$

In particular,

$$\|H\|_{\mathrm{op}} \leqslant \|A\|_{\mathrm{op}} + P.$$

The following lemma lets us replace the matrix exponential in Theorem 8.3 by a low-degree matrix polynomial. (For completeness, we include its standard proof in Appendix B.) The degree of this polynomial depends linearly on the spectral norm of the exponent S_t. Since the number of iterations of the algorithm is logarithmic in n, we have a bound on this spectral norm that is logarithmic in n. Hence matrix-vector multiplications for Q have the same cost as for S_t up to this logarithmic factor. However, explicitly computing Q may take quadratic time even for sparse matrices S_t.

Lemma 8.6 (Approximate matrix exponential)**.** *Let $S \in \mathbb{R}^{n\times n}$ be symmetric with $\|S\|_{\mathrm{op}} \leqslant L$, where $L \geqslant 0$, and let $0 < \alpha < 1$. Set $m = \lceil 8(L + \log(3/\alpha) + 1)\rceil$ and*

$$Q = \sum_{j=0}^{m-1} \frac{(S/2)^j}{j!}.$$

Then

$$(1-\alpha)e^S \preceq Q^2 \preceq (1+\alpha)e^S.$$

If one product with S costs T_S operations, then, for any $u \in \mathbb{R}^n$, computing Qu costs $O((L + \log(3/\alpha) + 1)(T_S + n))$ operations.

As a final ingredient, we sketch PSD matrices X, given access to a factorization $X = VV^T$, by a low-rank matrix ZZ^T while controlling quadratic forms in expectation in each fixed direction.

Lemma 8.7 (Random-sign quadratic-form estimate)**.** *Let $V \in \mathbb{R}^{n\times n}$ and let $k \geqslant 1$ be an integer. Draw $G \in \{-1,1\}^{n\times k}$ with independent uniform entries and set $Z = k^{-1/2}VG$. For every fixed $u \in \mathbb{R}^n$,*

$$\mathbb{E}\left|\left\langle u, (ZZ^T - VV^T)u\right\rangle\right| \leqslant \sqrt{2/k}\left\langle u, VV^T u\right\rangle.$$

Proof. We bound the variance of the scalar error. Put $a = V^T u$ and let g_j be the columns of G, so

$$u^T(ZZ^T - VV^T)u = k^{-1}\sum_{j=1}^{k}((a^T g_j)^2 - \|a\|_2^2).$$

Independence of the signs gives $\mathbb{E}(a^T g_j)^2 = \|a\|_2^2$ and

$$\mathrm{Var}((a^T g_j)^2) = 4\sum_{i<\ell} a_i^2 a_\ell^2 \leqslant 2\|a\|_2^4.$$

Independence of the columns and Cauchy–Schwarz therefore give $\mathbb{E}|u^T(ZZ^T - VV^T)u| \leqslant \sqrt{2/k}\|a\|_2^2$, the claimed bound since $\|a\|_2^2 = u^T VV^T u$. ∎

The following lemma combines the preceding two lemmas to efficiently construct a low-rank approximation $\widetilde{X}$ of X, with error controlled in the subgradient directions relevant to minimizing F. We defer its proof to Appendix B.

Lemma 8.8 (Exponential matrix density sketch)**.** *There is a randomized algorithm that, given a symmetric matrix $S \in \mathbb{R}^{n\times n}$ through a matrix–vector product oracle, a bound $L \geqslant \|S\|_{\mathrm{op}}$, and $0 < \alpha < 1$, returns $\boldsymbol{Z} \in \mathbb{R}^{n\times k}$ with $\|\boldsymbol{Z}\|_F = 1$ and $k \leqslant O(\alpha^{-2})$, such that $\widetilde{X} = \boldsymbol{Z}\boldsymbol{Z}^T \in \mathcal{S}$ satisfies, for every symmetric $A \in \mathbb{R}^{n\times n}$ and every $P \geqslant 0$,*

$$\mathbb{E} \sup_{\|d\|_\infty \leqslant P} \left| \left\langle A + \mathrm{Diag}(d), \widetilde{X} - X \right\rangle \right| \leqslant \alpha(\|A\|_{\mathrm{op}} + P),$$

where $X = e^S / \operatorname{Tr} e^S$. The algorithm uses $O(\alpha^{-2}(1 + L + \log(1/\alpha)))$ matrix-vector products with S and n times as many additional operations.

In order to efficiently compute subgradients for F at the points $\widetilde{X}_t$, we also need the following lemma.

Lemma 8.9 (Ball distance oracle)**.** *There is a deterministic algorithm that, given $K \geqslant 1$ and $p \in \mathbb{R}^n$, computes $y \in \partial f(p)$, where $f(x) = \mathrm{dist}_1(x, \mathcal{D}_K)$ for $x \in \mathbb{R}^n$. The algorithm uses $O(n \log n)$ arithmetic operations.*

Proof of Theorem 8.2. Set

$$\eta = \frac{\delta}{32P^2}, \qquad T = \left\lceil \frac{128P^2 \log n}{\delta^2} \right\rceil, \qquad \alpha = \frac{\delta}{16P}.$$

Initialize $S_1 = 0$. At iteration t, apply Lemma 8.8 with fresh independent randomness and the bound $L_t = 2\eta(t-1)P$ to obtain $\widetilde{X}_t = \boldsymbol{Z}_t \boldsymbol{Z}_t^T$. Compute $\operatorname{diag} \widetilde{X}_t$ from the squared row norms of $\boldsymbol{Z}_t$. Writing $f(p) = \mathrm{dist}_1(p, \mathcal{D}_K)$, Lemma 8.9 computes $y_t \in \partial f(\operatorname{diag} \widetilde{X}_t)$ in $O(n \log n)$ operations. Thus

$$H_t = A + P \operatorname{Diag}(y_t) \in \partial F(\widetilde{X}_t).$$

By Lemma 8.5, $\|H_t\|_{\mathrm{op}} \leqslant \|A\|_{\mathrm{op}} + P \leqslant 2P$. Set $S_{t+1} = S_t - \eta H_t$. In this way, the iterations maintain the invariant $\|S_t\|_{\mathrm{op}} \leqslant L_t$.

Conditioned on the preceding iterations, S_t is fixed. Although y_t depends on the current sketch, $\|y_t\|_\infty \leqslant 1$, so the supremum over diagonal perturbations in Lemma 8.8 gives

$$\mathbb{E}\langle H_t, \widetilde{X}_t - X_t \rangle \leqslant \alpha(\|A\|_{\mathrm{op}} + P) \leqslant 2\alpha P, \qquad X_t = \frac{e^{S_t}}{\operatorname{Tr} e^{S_t}}.$$

Consequently, Theorem 8.3 yields

$$\mathbb{E} F\left(\frac{1}{T} \sum_{t=1}^{T} \widetilde{X}_t \right) - \min_{\mathcal{S}} F \leqslant \frac{\log n}{\eta T} + 8\eta P^2 + 2\alpha P \leqslant \delta.$$

Return $\boldsymbol{Z} = T^{-1/2}[\boldsymbol{Z}_1 \ \cdots \ \boldsymbol{Z}_T]$. Its Gram matrix is the average above and belongs to $\mathcal{S}$.

It remains to bound the cost. Maintain $b_t = \sum_{s<t} y_s$ incrementally, so

$$S_t x = -\eta\big((t-1)Ax + P\,\mathrm{Diag}(b_t)x\big).$$

Each product with S_t therefore uses one product with A and $O(n)$ additional operations. Since $0 < \delta \leqslant P$,

$$T = O((P/\delta)^2 \log n), \qquad 1 + L_t + \log(1/\alpha) = O((P/\delta)\log n).$$

The sketch lemma thus gives a total of $O((P/\delta)^5 (\log n)^2)$ products with A and

$$r = O(T\alpha^{-2}) = O(P^4 \log n/\delta^4)$$

output columns. Computing the diagonals, forming the output factor, maintaining b_t, and the T ball-oracle calls all fit n times this product bound. ■

9 Analysis of postprocessing by lifting

In this section we prove Lemma 5.3.

Lemma 5.3 (Robust postprocessing)**.** *Let $(\boldsymbol{G}, \boldsymbol{\sigma}^\star) \sim \mathrm{SBM}(n, d, \lambda)$ and $\boldsymbol{\xi}$ be a unit eigenvector for the eigenvalue of $B_{\boldsymbol{G}}$ of second-largest modulus. For every $d \geqslant 1$, $\lambda \in (-1, 1)$ with $d\lambda^2 > 1$, and $\varepsilon > 0$, there are $\eta > 0$, $\delta > 0$, $s_0 \in \mathbb{N}$, and a linear-time algorithm with the following guarantee for every $s \geqslant s_0$: with high probability over $(\boldsymbol{G}, \boldsymbol{\sigma}^\star)$, given input $y \in \mathbb{R}^n$ satisfying*

$$\min_{a \in \{\pm 1\}} \left\| y - a \frac{\boldsymbol{\sigma}^{(s)}}{\|\boldsymbol{\sigma}^{(s)}\|} \right\| \leqslant \delta,$$

and an arbitrary graph F satisfying $|E(F) \triangle E(\boldsymbol{G})| \leqslant \eta n$, the algorithm outputs a vector $\widehat{\xi}(F)$ such that

$$\min_{a \in \{\pm 1\}} \|\widehat{\xi}(F) - a\boldsymbol{\xi}\| \leqslant \varepsilon.$$

We first describe the algorithm. For a graph Q on $[n]$ and $x \in \mathbb{R}^n$, let $R_Q x \in \mathbb{R}^{\vec{E}(Q)}$ lift to a vector on directed edges:

$$(R_Q x)_{u \to v} = x_v - \frac{x_u}{d\lambda}.$$

For $J \geqslant 1$ let Q_J be the graph obtained from Q by deleting every edge with an endpoint of degree more than J in Q. As in Section 4, vectors on the directed edges of different graphs are compared as vectors indexed by all ordered pairs, in $\mathbb{R}^{n^2-n}$. Given y and F, the postprocessing algorithm $\mathcal{A}_{\mathrm{post}}$ outputs $\widehat{\xi}(F) = R_{F_J} y / \|R_{F_J} y\|$, for a constant $J = J(d, \lambda, \varepsilon)$ chosen below; it runs in time $O(n + |E(F)|)$.

We put together a few lemmas to prove Lemma 5.3. First, recall the "walk score vector" from Section 4,

$$W^{(\ell)}_{u \to v} = (d\lambda)^{-\ell} \sum_{\substack{\text{nonbacktracking } (u_0, \ldots, u_{\ell+1}) \\ u_0 = u,\ u_1 = v}} \sigma^\star_{u_{\ell+1}},$$

which is close to the second eigenvector $\boldsymbol{\xi}$ of the nonbacktracking matrix for SBM graphs.

The first lemma shows that the lifting operation R_G lifts the vector $A^{(s)}\boldsymbol{\sigma}^*$ to a vector close to $W^{(s)}$ (up to rescaling). We prove it at the end of this section.

Lemma 9.1 (Lifting the vertex walk score). *Let $(\boldsymbol{G}, \sigma^\star) \sim \mathrm{SBM}(n, d, \lambda)$ with $d\lambda^2 > 1$, and let $W^{(\ell)}$ be the walk scores of Section 4. For every $\varepsilon > 0$ there is $s_0 = s_0(d, \lambda, \varepsilon)$ such that for every fixed $s \geqslant s_0$, with high probability,*

$$\left\| \frac{(d\lambda)^2}{(d\lambda)^2 - 1} \cdot R_G \frac{A_G^{(s)} \sigma^\star}{(d\lambda)^s} - W^{(s)} \right\| \leqslant \varepsilon \sqrt{n}\,.$$

The second lemma captures robustness of the lifting operation – it's not too harmful that the algorithm uses the corrupted graph F to do the lifting. We prove it at the end of this section.

Lemma 9.2 (Lifting using the corrupted graph is ok). *For every $\varepsilon > 0$ there is $J_0(\varepsilon)$ such that for every $J \geqslant J_0$ there is $\eta > 0$ such that for every sufficiently large fixed s, with high probability, every graph F on $[n]$ with $|E(F) \triangle E(\boldsymbol{G})| \leqslant \eta n$ satisfies*

$$\left\| R_{F_J} \frac{A_G^{(s)} \sigma^\star}{(d\lambda)^s} - R_G \frac{A_G^{(s)} \sigma^\star}{(d\lambda)^s} \right\| \leqslant \varepsilon \sqrt{n}\,.$$

Moreover, every graph Q on $[n]$ with maximum degree at most J satisfies $\|R_Q\| \leqslant O_{d,\lambda,J}(1)$.

Finally, we will need some basic bounds on norms of $W^{(s)}$ and $A_G^{(s)} \sigma^*$.

Fact 9.3 (Norms of the walk scores). *Let $(\boldsymbol{G}, \sigma^\star) \sim \mathrm{SBM}(n, d, \lambda)$ with $d\lambda^2 > 1$. There is $C = C(d, \lambda)$, not depending on s, such that for every fixed $s \geqslant 1$, with high probability,*

$$\frac{\sqrt{n}}{C} \leqslant \|W^{(s)}\| \leqslant C\sqrt{n} \quad \text{and} \quad \left\| \frac{A_G^{(s)} \sigma^\star}{(d\lambda)^s} \right\| \leqslant C\sqrt{n}\,.$$

These follow from Fact 4.7 and [SM22, Proposition 6]; see also [SM22, Proposition 10].

Proof of Lemma 5.3. We elide the $\min_{a \in \{\pm 1\}}$ through this proof for simplicity. Fix $\varepsilon > 0$. It will be more convenient to work with a rescaling of y given by $z = \|A_G^{(s)} \sigma^* / (d\lambda)^s\| \cdot y$. With δ to be chosen later, the algorithm gets as input z such that

$$\|z - A_G^{(s)} \sigma^* / (d\lambda)^s\| \leqslant \delta \|A_G^{(s)} \sigma^* / (d\lambda)^s\| \leqslant C\delta\sqrt{n}$$

with high probability, where $C = C(d, \lambda)$ is the constant of Fact 9.3.

We use this to bound the difference between the lifted vector $R_{F_J} z$ and the second eigenvector, with some convenient scalings chosen to create cancelations.

We have

$$\begin{aligned}\left\| \frac{(d\lambda)^2}{(d\lambda)^2 - 1} R_{F_J} z - \|W^{(s)}\| \boldsymbol{\xi} \right\| \leqslant{}& \frac{(d\lambda)^2}{(d\lambda)^2 - 1} \left\| R_{F_J} \left(z - \frac{A_G^{(s)} \sigma^\star}{(d\lambda)^s} \right) \right\| + \frac{(d\lambda)^2}{(d\lambda)^2 - 1} \left\| R_{F_J} \frac{A_G^{(s)} \sigma^\star}{(d\lambda)^s} - R_G \frac{A_G^{(s)} \sigma^\star}{(d\lambda)^s} \right\| \\ &+ \left\| \frac{(d\lambda)^2}{(d\lambda)^2 - 1} R_G \frac{A_G^{(s)} \sigma^\star}{(d\lambda)^s} - W^{(s)} \right\| + \left\| W^{(s)} - \|W^{(s)}\| \boldsymbol{\xi} \right\|.\end{aligned}$$

For the first term, $\|R_{F_J}\| \leqslant O_{d,\lambda,J}(1)$ by Lemma 9.2 and $\left\| z - A_G^{(s)} \sigma^\star / (d\lambda)^s \right\| \leqslant C\delta\sqrt{n}$, so it is at most $O_{d,\lambda,J}(\delta)\sqrt{n}$.

For the second term, Lemma 9.2 bounds it by $\frac{(d\lambda)^2}{(d\lambda)^2-1}\varepsilon\sqrt{n}$, choosing J large enough and then η small enough.

For the third term, Lemma 9.1 bounds it by $\varepsilon\sqrt{n}$ once s is large.

For the fourth term, $\|W^{(s)} - \|W^{(s)}\|\boldsymbol{\xi}\| = \|W^{(s)}\| \cdot \|W^{(s)}/\|W^{(s)}\| - \boldsymbol{\xi}\| \leqslant C\sqrt{n} \cdot o_{s\to\infty}(1)$ by Fact 9.3 and Lemma 4.3.

All together, the left-hand side is at most $\big(O_{d,\lambda,J}(\delta) + O_{d,\lambda}(\varepsilon) + o_{s\to\infty}(1)\big)\sqrt{n}$. Since $\|W^{(s)}\|\boldsymbol{\xi}$ has norm $\|W^{(s)}\| \geqslant \sqrt{n}/C$, the inequality $\|x/\|x\| - x'/\|x'\|\| \leqslant 2\|x-x'\|/\|x'\|$ gives $\|\widehat{\boldsymbol{\xi}}(F) - \boldsymbol{\xi}\| \leqslant 2C\big(O_{d,\lambda,J}(\delta) + O_{d,\lambda}(\varepsilon) + o_{s\to\infty}(1)\big)$, which is small once ε and then δ are small and s is large. ∎

It remains to prove Lemmas 9.1 and 9.2.

Proof of Lemma 9.1. We will show that $R_{\boldsymbol{G}} A_{\boldsymbol{G}}^{(s)} \boldsymbol{\sigma}^*$ is $W^{(s)}_{u\to v}$ up to rescaling and some explicit low-order corrections.

Fix a directed edge $u \to v$ of $\boldsymbol{G}$. $(A_{\boldsymbol{G}}^{(s)}\boldsymbol{\sigma}^*)_v$ is a sum over nonbacktracking walks beginning at v. We can split that sum according to whether the first step from v goes to u or not, to get

$$(A_{\boldsymbol{G}}^{(s)}\boldsymbol{\sigma}^\star)_v = (d\lambda)^s W^{(s)}_{u\to v} + (d\lambda)^{s-1} W^{(s-1)}_{v\to u} \ .$$

Exchanging the roles of u and v,

$$(A_{\boldsymbol{G}}^{(s)}\boldsymbol{\sigma}^\star)_u = (d\lambda)^s W^{(s)}_{v\to u} + (d\lambda)^{s-1} W^{(s-1)}_{u\to v} \, .$$

Substituting both into $(R_{\boldsymbol{G}}x)_{u\to v} = x_v - x_u/(d\lambda)$,

$$\begin{aligned}\Big(R_{\boldsymbol{G}} \frac{A_{\boldsymbol{G}}^{(s)}\boldsymbol{\sigma}^\star}{(d\lambda)^s}\Big)_{u\to v} &= W^{(s)}_{u\to v} - \frac{W^{(s-1)}_{u\to v}}{(d\lambda)^2} - \frac{W^{(s)}_{v\to u} - W^{(s-1)}_{v\to u}}{d\lambda} \\ &= \Big(1 - \frac{1}{(d\lambda)^2}\Big) W^{(s)}_{u\to v} + \frac{W^{(s)}_{u\to v} - W^{(s-1)}_{u\to v}}{(d\lambda)^2} - \frac{W^{(s)}_{v\to u} - W^{(s-1)}_{v\to u}}{d\lambda} \, .\end{aligned}$$

By triangle inequality,

$$\Big\| \frac{(d\lambda)^2}{(d\lambda)^2-1} R_{\boldsymbol{G}} \frac{A_{\boldsymbol{G}}^{(s)}\boldsymbol{\sigma}^\star}{(d\lambda)^s} - W^{(s)} \Big\| \leqslant \frac{1+|d\lambda|}{(d\lambda)^2-1} \big\| W^{(s)} - W^{(s-1)} \big\| \, .$$

By Lemma 4.9, with high probability $\|W^{(s)} - W^{(s-1)}\| \leqslant \big(O_{d,\lambda}((d\lambda^2)^{-s/2}) + o(1)\big)\sqrt{n}$, which is at most $\varepsilon\sqrt{n} \cdot ((d\lambda)^2-1)/(1+|d\lambda|)$ once s is large. ∎

Proof of Lemma 9.2. We first establish the spectral norm bound in the "moreover". Let Q be any graph on $[n]$ with maximum degree J; let $x \in \mathbb{R}^n$. Then

$$\sum_{u\sim v} (R_Q x)^2_{u\to v} \leqslant O_{d,\lambda}(1) \cdot \sum_{u\sim v} (x_u^2 + x_v^2) \leqslant O_{d,\lambda}(J)\|x\|^2 \, .$$

Now we turn to the main statement of the lemma. Henceforth in this proof let $x = A_{\boldsymbol{G}}^{(s)}\boldsymbol{\sigma}^*/(d\lambda)^s$, so our goal is to bound $\|R_{F_J}x - R_{\boldsymbol{G}}x\|$. For any edge u,v in both $\boldsymbol{G}$ and F_J the vector $(R_{F_J}x - R_{\boldsymbol{G}}x)_{u\to v}$ is zero. So

$$\|R_{F_J}x - R_{\boldsymbol{G}}x\|^2 \leqslant O_{d,\lambda}(1) \sum_{\{u,v\}\in E(F_J)\triangle E(\boldsymbol{G})} (x_u^2 + x_v^2) \, .$$

Let $m(v)$ be the number of edges of $E(F_J) \triangle E(\boldsymbol{G})$ at v. By Cauchy–Schwarz,

$$\sum_{\{u,v\} \in E(F_J) \triangle E(\boldsymbol{G})} (x_u^2 + x_v^2) = \sum_{v \in [n]} m(v)\, x_v^2 \leqslant \Big(\sum_{v \in [n]} m(v)^2 \Big)^{1/2} \|x\|_4^2 .$$

The vector x is the local statistic $\boldsymbol{\sigma}^{(s)}$ of Section 6, so by Lemma 6.4 and Lemma 6.5(iii), with high probability $\|x\|_4^4 \leqslant Cn$ for a constant $C = C(d, \lambda)$ not depending on s. So we just need to bound $\sum_{v \in [n]} m(v)^2$ with high probability.

Since $E(\boldsymbol{G}_J) \subseteq E(\boldsymbol{G})$, every edge of $E(F_J) \triangle E(\boldsymbol{G})$ lies in $E(\boldsymbol{G}) \setminus E(\boldsymbol{G}_J)$ or in $E(F_J) \triangle E(\boldsymbol{G}_J)$. So $m(v) \leqslant g_J(v) + m'(v)$, where $g_J(v)$ is the number of edges of $E(\boldsymbol{G}) \setminus E(\boldsymbol{G}_J)$ at v and $m'(v)$ is the number of edges of $E(F_J) \triangle E(\boldsymbol{G}_J)$ at v, and

$$\sum_{v \in [n]} m(v)^2 \leqslant 2 \sum_{v \in [n]} g_J(v)^2 + 2 \sum_{v \in [n]} m'(v)^2 .$$

The first sum is the error from the degree cutoff, and the second is the error from the corruptions.

For the first sum, $g_J(v)$ is the number of edges at v touching a vertex of degree more than J, so g_J^2 is a local statistic and by Lemma 6.4, $\frac{1}{n} \sum_{v \in [n]} g_J(v)^2 \to \|g_J\|_{L^2}^2$. On the broadcast tree, $g_J(o) \leqslant \deg(o)$ and $g_J(o) \to 0$ as $J \to \infty$, so $\|g_J\|_{L^2} \to 0$.

For the second sum, both F_J and $\boldsymbol{G}_J$ have maximum degree at most J, so $m'(v) \leqslant 2J$ and

$$\sum_{v \in [n]} m'(v)^2 \leqslant 2J \sum_{v \in [n]} m'(v) = 4J\, |E(F_J) \triangle E(\boldsymbol{G}_J)| .$$

We claim that $|E(F_J) \triangle E(\boldsymbol{G}_J)| \leqslant O(J\eta n)$. An edge of $E(\boldsymbol{G}_J) \setminus E(F_J)$ was either removed by the adversary, or it is still in F but has an endpoint whose degree the adversary pushed above J. There are at most $2\eta n$ endpoints of corrupted edges, and each has at most J edges in $\boldsymbol{G}_J$, so $|E(\boldsymbol{G}_J) \setminus E(F_J)| \leqslant (2J+1)\eta n$. Symmetrically, $|E(F_J) \setminus E(\boldsymbol{G}_J)| \leqslant (2J+1)\eta n$.

Putting these together, with high probability, simultaneously for every F with $|E(F) \triangle E(\boldsymbol{G})| \leqslant \eta n$,

$$\frac{1}{n} \sum_{v \in [n]} m(v)^2 \leqslant 4 \|g_J\|_{L^2}^2 + O(J^2 \eta) .$$

Choosing J large enough and then η small enough makes this as small as desired, which concludes the proof. ∎

10 AI methodology

In May 2026, the authors proved, without any AI assistance, that initializing belief propagation at the second eigenvector succeeds at optimal recovery for any set of parameters above the KS threshold, along with a robust algorithm based on the Bethe Hessian matrix to estimate the second eigenvector in the presence of adversarial corruptions when the degree $d \to \infty$ sufficiently above the KS threshold.

In July 2026, the authors extensively interacted with GPT-5.6 Sol to upgrade the robust algorithm to estimate the second eigenvector for all $d > 1$ and all SNR above the KS threshold based on a

degree-truncated nonbacktracking matrix power. The analysis was highly technical, and required proving spectral norm estimates on asymmetric random matrices with correlated entries. To abet verification, the authors used GPT-5.6 Sol to produce a Lean certificate.

However, this paper ultimately did not use that proof in favor of a shorter and more human-verifiable proof—in September 2026, the authors fed their earlier research notes to GPT-6 Astra with instructions to upgrade the Bethe Hessian-based algorithm to all parameter regimes, and in a single shot it produced a robust algorithm for second eigenvector estimation based on a suitably "corrected" Bethe Hessian matrix presented here.

The authors interacted extensively with frontier models (GPT-6 Astra, GPT-6.1 Sol, Opus 5.5, and Fable 5.1) to streamline proofs by finding simplifications, and to identify relevant literature containing quantitative estimates used in our proofs. These models were used as writing and editing aids in generating this manuscript.

Acknowledgements

The authors thank Prasad Raghavendra for many conversations about robust community estimation.

References

[ABARS20] Emmanuel Abbe, Enric Boix-Adserà, Peter Ralli, and Colin Sandon. Graph powering and spectral robustness. *SIAM Journal on Mathematics of Data Science*, 2(1):132–157, 2020. 4

[Abb18] Emmanuel Abbe. Community detection and stochastic block models: Recent developments. *Journal of Machine Learning Research*, 18(177):1–86, 2018. 1, 4

[ABH16] Emmanuel Abbe, Afonso S. Bandeira, and Georgina Hall. Exact recovery in the stochastic block model. *IEEE Transactions on Information Theory*, 62(1):471–487, 2016. 4

[ACGP21] Emmanuel Abbe, Elisabetta Cornacchia, Yuzhou Gu, and Yury Polyanskiy. Stochastic block model entropy and broadcasting on trees with survey. In *Proceedings of the 34th Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pages 1–25. PMLR, 2021. 4

[AHK12] Sanjeev Arora, Elad Hazan, and Satyen Kale. The multiplicative weights update method: A meta-algorithm and applications. *Theory of Computing*, 8(6):121–164, 2012. 40, 58

[AK07] Sanjeev Arora and Satyen Kale. A combinatorial, primal-dual approach to semidefinite programs. In *Proceedings of the 39th Annual ACM Symposium on Theory of Computing*, pages 227–236. ACM, 2007. Full version. 12, 37

[AN72] Krishna B. Athreya and Peter E. Ney. *Branching Processes*, volume 196 of *Die Grundlehren der mathematischen Wissenschaften*. Springer-Verlag, Berlin, 1972. 13

[AS18] Emmanuel Abbe and Colin Sandon. Proof of the achievability conjectures for the general stochastic block model. *Communications on Pure and Applied Mathematics*, 71(7):1334–1406, 2018. 1

[Bas92] Hyman Bass. The Ihara–Selberg zeta function of a tree lattice. *International Journal of Mathematics*, 3(6):717–797, 1992. 25

[BBK+21] Afonso S. Bandeira, Jess Banks, Dmitriy Kunisky, Cristopher Moore, and Alexander S. Wein. Spectral planting and the hardness of refuting cuts, colorability, and communities in random graphs. In *Proceedings of the 34th Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pages 410–473. PMLR, 2021. 1

[BCLS87] 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. 1

[BJK+24] Aditya Bhaskara, Agastya Vibhuti Jha, Michael Kapralov, Naren Sarayu Manoj, Davide Mazzali, and Weronika Wrzos-Kaminska. On the robustness of spectral algorithms for semirandom stochastic block models. In *Advances in Neural Information Processing Systems*, volume 37, pages 112731–112776, 2024. 1

[BJR07] Béla Bollobás, Svante Janson, and Oliver Riordan. The phase transition in inhomogeneous random graphs. *Random Structures & Algorithms*, 31(1):3–122, 2007. 1

[BLM18] Charles Bordenave, Marc Lelarge, and Laurent Massoulié. Nonbacktracking spectrum of random graphs: community detection and nonregular Ramanujan graphs. *The Annals of Probability*, 46(1):1–71, 2018. 1, 5, 6, 8, 9, 18, 29, 55, 57

[BMNN16] Jess Banks, Cristopher Moore, Joe Neeman, and Praneeth Netrapalli. Information-theoretic thresholds for community detection in sparse networks. In *Proceedings of the 29th Conference on Learning Theory*, volume 49 of *Proceedings of Machine Learning Research*, pages 383–416. PMLR, 2016. 4

[BMR21] Jess Banks, Sidhanth Mohanty, and Prasad Raghavendra. Local statistics, semidefinite programming, and community detection. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms*, pages 1298–1316. SIAM, 2021. Full version: arXiv:1911.01960. 1

[Bop87] Ravi B. Boppana. Eigenvalues and graph bisection: An average-case analysis. In *Proceedings of the 28th Annual Symposium on Foundations of Computer Science*, pages 280–285. IEEE, 1987. 1

[BS95] Avrim Blum and Joel Spencer. Coloring random and semi-random k-colorable graphs. *Journal of Algorithms*, 19(2):204–234, 1995. 2

[BV04] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004. 40

[CCAd+23] Hongjie Chen, Vincent Cohen-Addad, Tommaso d'Orsi, Alessandro Epasto, Jacob Imola, David Steurer, and Stefan Tiegel. Private estimation algorithms for stochastic block models and mixture models. In *Advances in Neural Information Processing Systems*, volume 36, pages 68134–68183, 2023. 1

[CL15] T. Tony Cai and Xiaodong Li. Robust and computationally feasible community detection in the presence of arbitrary outlier nodes. *The Annals of Statistics*, 43(3):1027–1059, 2015. 4

[CLMW11] Emmanuel J. Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *Journal of the ACM*, 58(3):11:1–11:37, 2011. 11, 37

[CO10] Amin Coja-Oghlan. Graph partitioning via adaptive spectral techniques. *Combinatorics, Probability and Computing*, 19(2):227–284, 2010. 1

[COMV09] Amin Coja-Oghlan, Elchanan Mossel, and Dan Vilenchik. A spectral approach to analysing belief propagation for 3-colouring. *Combinatorics, Probability and Computing*, 18(6):881–912, 2009. 4

[CRV15] Peter Chin, Anup Rao, and Van Vu. Stochastic block model and community detection in sparse graphs: A spectral algorithm with optimal rate of recovery. In *Proceedings of the 28th Conference on Learning Theory*, volume 40 of *Proceedings of Machine Learning Research*, pages 391–423. PMLR, 2015. 4

[CSPW11] Venkat Chandrasekaran, Sujay Sanghavi, Pablo A. Parrilo, and Alan S. Willsky. Rank-sparsity incoherence for matrix decomposition. *SIAM Journal on Optimization*, 21(2):572–596, 2011. 11

[DAM17] Yash Deshpande, Emmanuel Abbe, and Andrea Montanari. Asymptotic mutual information for the balanced binary stochastic block model. *Information and Inference: A Journal of the IMA*, 6(2):125–170, 2017. 4

[DdHS23] Jingqiu Ding, Tommaso d'Orsi, Yiding Hua, and David Steurer. Reaching Kesten-Stigum threshold in the stochastic block model under node corruptions. In *Proceedings of the 36th Conference on Learning Theory*, volume 195 of *Proceedings of Machine Learning Research*, pages 4044–4071. PMLR, 2023. 1, 2, 4

[DdNS21] Jingqiu Ding, Tommaso d'Orsi, Rajai Nasser, and David Steurer. Robust recovery for stochastic block models. In *Proceedings of the 62nd IEEE Annual Symposium on Foundations of Computer Science*, pages 387–394. IEEE, 2021. Full version: arXiv:2111.08568. 1, 2, 7

[DF89] M. E. Dyer and A. M. Frieze. The solution of some random NP-hard problems in polynomial expected time. *Journal of Algorithms*, 10(4):451–489, 1989. 1

[DHL+26] Jingqiu Ding, Yiding Hua, Kasper Lindberg, David Steurer, and Aleksandr Storozhenko. Rate-optimal community detection near the KS threshold via node-robust algorithms. In *Proceedings of the 39th Conference on Learning Theory*, volume 336 of *Proceedings of Machine Learning Research*, pages 1965–2037. PMLR, 2026. 1, 2

[DK23] Ilias Diakonikolas and Daniel M. Kane. *Algorithmic High-Dimensional Robust Statistics*. Cambridge University Press, 2023. 2, 4

[DKMZ11a] Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborová. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Physical Review E*, 84(6):066106, 2011. 1

[DKMZ11b] Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborová. Inference and phase transitions in the detection of modules in sparse networks. *Physical Review Letters*, 107(6):065701, 2011. 1

[DM26] Dingding Dong and Theo McKenzie. Bethe Hessian recovery through the Kesten–Stigum regime, 2026. 4

[EKPS00] William Evans, Claire Kenyon, Yuval Peres, and Leonard J. Schulman. Broadcasting on trees and the Ising model. *The Annals of Applied Probability*, 10(2):410–433, 2000. 9

[FC20] Yingjie Fei and Yudong Chen. Achieving the Bayes error rate in synchronization and block models by SDP, robustly. *IEEE Transactions on Information Theory*, 66(6):3929–3953, 2020. 4

[Fei21] Uriel Feige. Introduction to semirandom models. In Tim Roughgarden, editor, *Beyond the Worst-Case Analysis of Algorithms*, chapter 9, pages 189–211. Cambridge University Press, 2021. 2

[FK01] Uriel Feige and Joe Kilian. Heuristics for semirandom graph problems. *Journal of Computer and System Sciences*, 63(4):639–671, 2001. 2

[FO05] Uriel Feige and Eran Ofek. Spectral techniques applied to sparse random graphs. *Random Structures & Algorithms*, 27(2):251–275, 2005. 12

[GMZZ17] Chao Gao, Zongming Ma, Anderson Y. Zhang, and Harrison H. Zhou. Achieving optimal misclassification proportion in stochastic block models. *Journal of Machine Learning Research*, 18(60):1–45, 2017. 4

[GP23] Yuzhou Gu and Yury Polyanskiy. Uniqueness of BP fixed point for the Potts model and applications to community detection. In *Proceedings of the 36th Conference on Learning Theory*, volume 195 of *Proceedings of Machine Learning Research*, pages 837–884, 2023. 1

[GV16] Olivier Guédon and Roman Vershynin. Community detection in sparse networks via Grothendieck's inequality. *Probability Theory and Related Fields*, 165(3-4):1025–1049, 2016. 1, 11, 37

[Has89] Ki-ichiro Hashimoto. Zeta functions of finite graphs and representations of p-adic groups. In *Automorphic Forms and Geometry of Arithmetic Varieties*, volume 15 of *Advanced Studies in Pure Mathematics*, pages 211–280. Mathematical Society of Japan, 1989. 13

[HL24] Samuel B. Hopkins and Anqi Li. Adversarially-robust inference on trees via belief propagation. In *Proceedings of the 37th Conference on Learning Theory*, volume 247 of *Proceedings of Machine Learning Research*, pages 2389–2417. PMLR, 2024. 7

[HLL83] Paul W. Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels: First steps. *Social Networks*, 5(2):109–137, 1983. 1

[Hop18] Samuel B. Hopkins. *Statistical Inference and the Sum of Squares Method*. PhD thesis, Cornell University, 2018. 1

[HS17] Samuel B. Hopkins and David Steurer. Efficient Bayesian estimation from few samples: Community detection and related problems. In *Proceedings of the 58th IEEE Annual Symposium on Foundations of Computer Science*, pages 379–390. IEEE, 2017. 1

[HWX16] Bruce Hajek, Yihong Wu, and Jiaming Xu. Achieving exact cluster recovery threshold via semidefinite programming. *IEEE Transactions on Information Theory*, 62(5):2788–2797, 2016. 1

[Iha66] Yasutaka Ihara. On discrete subgroups of the two by two projective linear group over $\mathfrak{p}$-adic fields. *Journal of the Mathematical Society of Japan*, 18(3):219–235, 1966. 25

[IS24] Misha Ivkov and Tselil Schramm. Semidefinite programs simulate approximate message passing robustly. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pages 348–357. ACM, 2024. 4

[IS25] Misha Ivkov and Tselil Schramm. Fast, robust approximate message passing. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing*, pages 2237–2248. ACM, 2025. 4

[IS26] Misha Ivkov and Tselil Schramm. Easy, robust approximate message passing for planted spike models, 2026. 4

[JM04] Svante Janson and Elchanan Mossel. Robust reconstruction on trees is determined by the second eigenvalue. *The Annals of Probability*, 32(3B):2630–2649, 2004. 14

[KMM+13] Florent Krzakala, Cristopher Moore, Elchanan Mossel, Joe Neeman, Allan Sly, Lenka Zdeborová, and Pan Zhang. Spectral redemption in clustering sparse networks. *Proceedings of the National Academy of Sciences*, 110(52):20935–20940, 2013. 1

[KMS16] Varun Kanade, Elchanan Mossel, and Tselil Schramm. Global and local information in clustering labeled block models. *IEEE Transactions on Information Theory*, 62(10):5906–5917, 2016. 7

[KS66] H. Kesten and B. P. Stigum. Additional limit theorems for indecomposable multidimensional Galton–Watson processes. *The Annals of Mathematical Statistics*, 37(6):1463–1481, 1966. 9

[KWB22] Dmitriy Kunisky, Alexander S. Wein, and Afonso S. Bandeira. Notes on computational hardness of hypothesis testing: Predictions using the low-degree likelihood ratio. In Paula Cerejeiras and Michael Reissig, editors, *Mathematical Analysis, its Applications and Computation (ISAAC 2019)*, Springer Proceedings in Mathematics & Statistics, pages 1–50. Springer, 2022. 1

[LM22] Allen Liu and Ankur Moitra. Minimax rates for robust community detection. In *Proceedings of the 63rd IEEE Annual Symposium on Foundations of Computer Science*, pages 823–831. IEEE, 2022. Full version: arXiv:2207.11903. 1, 2, 4

[LMR+24] Kuikui Liu, Sidhanth Mohanty, Prasad Raghavendra, Amit Rajaraman, and David X Wu. Locally stationary distributions: A framework for analyzing slow-mixing markov chains. In *2024 IEEE 65th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 203–215. IEEE, 2024. 4

[Mas14] Laurent Massoulié. Community detection thresholds and the weak Ramanujan property. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, pages 694–703. ACM, 2014. Full version: arXiv:1311.3085. 1, 5

[MMV12] Konstantin Makarychev, Yury Makarychev, and Aravindan Vijayaraghavan. Approximation algorithms for semi-random partitioning problems. In *Proceedings of the 44th Annual ACM Symposium on Theory of Computing*, pages 367–384. ACM, 2012. 1

[MMV16] Konstantin Makarychev, Yury Makarychev, and Aravindan Vijayaraghavan. Learning communities in the presence of errors. In *Proceedings of the 29th Conference on Learning Theory*, volume 49 of *Proceedings of Machine Learning Research*, pages 1258–1291. PMLR, 2016. 4

[MNS15] Elchanan Mossel, Joe Neeman, and Allan Sly. Reconstruction and estimation in the planted partition model. *Probability Theory and Related Fields*, 162(3–4):431–461, 2015. 1, 5, 6, 9

[MNS16a] Elchanan Mossel, Joe Neeman, and Allan Sly. Belief propagation, robust reconstruction and optimal recovery of block models. *The Annals of Applied Probability*, 26(4):2211–2256, 2016. 1, 2, 3, 6, 7, 13, 14, 15

[MNS16b] Elchanan Mossel, Joe Neeman, and Allan Sly. Consistency thresholds for the planted bisection model. *Electronic Journal of Probability*, 21(21):1–24, 2016. 4

[MNS18] Elchanan Mossel, Joe Neeman, and Allan Sly. A proof of the block model threshold conjecture. *Combinatorica*, 38(3):665–708, 2018. 1, 5

[MP03] Elchanan Mossel and Yuval Peres. Information flow on trees. *The Annals of Applied Probability*, 13(3):817–844, 2003. 9

[MPW16] Ankur Moitra, William Perry, and Alexander S. Wein. How robust are reconstruction thresholds for community detection? In *Proceedings of the 48th Annual ACM Symposium on Theory of Computing*, pages 828–841. ACM, 2016. Full version: arXiv:1511.01473. 1

[MRW24] Sidhanth Mohanty, Prasad Raghavendra, and David X. Wu. Robust recovery for stochastic block models, simplified and generalized. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pages 367–374, 2024. Full version: arXiv:2402.13921. 1, 2, 4, 7, 11, 12, 24, 25, 29

[MS16] Andrea Montanari and Subhabrata Sen. Semidefinite programs on sparse random graphs and their application to community detection. In *Proceedings of the 48th Annual ACM Symposium on Theory of Computing*, pages 814–827. ACM, 2016. 1, 11

[MV21] Andrea Montanari and Ramji Venkataramanan. Estimation of low-rank matrices via approximate message passing. *The Annals of Statistics*, 49(1):321–345, 2021. 4

[MX16] Elchanan Mossel and Jiaming Xu. Local algorithms for block models with side information. In *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science*, pages 71–80. ACM, 2016. 7

[SKZ14] Alaa Saade, Florent Krzakala, and Lenka Zdeborová. Spectral clustering of graphs with the Bethe Hessian. In *Advances in Neural Information Processing Systems 27*, pages 406–414, 2014. 1, 4, 11, 25

[SM19] Ludovic Stephan and Laurent Massoulié. Robustness of spectral methods for community detection. In *Proceedings of the 32nd Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pages 2831–2860. PMLR, 2019. 4

[SM22] Ludovic Stephan and Laurent Massoulié. Non-backtracking spectra of weighted inhomogeneous random graphs. *Mathematical Statistics and Learning*, 5(3/4):201–271, 2022. 19, 44

[Söd02] Bo Söderberg. General formalism for inhomogeneous random graphs. *Physical Review E*, 66(6):066121, 2002. 1

[SV13] Sushant Sachdeva and Nisheeth K. Vishnoi. Faster algorithms via approximation theory. *Foundations and Trends in Theoretical Computer Science*, 9(2):125–210, 2013. 58

[SZ25] Ludovic Stephan and Yizhe Zhu. Community detection with the Bethe–Hessian. In *Proceedings of the 38th Conference on Learning Theory*, volume 291 of *Proceedings of Machine Learning Research*, pages 5352–5353. PMLR, 2025. Full version: arXiv:2411.02835. 4

[YP24] Qian Yu and Yury Polyanskiy. Ising model on locally tree-like graphs: uniqueness of solutions to cavity equations. *IEEE Transactions on Information Theory*, 70(3):1913–1938, 2024. 1, 2, 6, 13, 14, 15

[Zde17] Lenka Zdeborová. BP with good initialization. Problem 4.2 in *AimPL: Connecting communities via the block model*, 2017. Open problem from the AIM workshop "Connecting communities via the block model", May 22–26, 2017. 3

[ZZ16] Anderson Y. Zhang and Harrison H. Zhou. Minimax rates of community detection in stochastic block models. *The Annals of Statistics*, 44(5):2252–2280, 2016. 4

A Proof of Lemma 6.9

In this section, we establish Lemma 6.9.

Lemma 6.9 ($\sigma^{(s)}$ is an approximate eigenvector of $C^{(s)}$)**.** *Let $c = d\lambda^2/(d\lambda^2-1)$. As $s \to \infty$, for each $s' \in \{s-1, s, s+1\}$,*

(a) $\|C^{(s)}\boldsymbol{\sigma}^{(s')} - c\,\boldsymbol{\sigma}^{(s)}\| \to 0$,

(b) $\|C^{(s)}(D-I)\boldsymbol{\sigma}^{(s')} - (dc+1)\,\boldsymbol{\sigma}^{(s)}\| \to 0$, *and*

(c) $\|C^{(s)}H(C^{(s)}\boldsymbol{\sigma}^{(s)} - c\,\boldsymbol{\sigma}^{(s)})\| \to 0$.

Denote $\rho = d\lambda^2 > 1$. We start by proving (a) and (b).

Proof of (a,b). We begin with $s' = s$. For succinctness, denote $\mu = d\lambda$. For a vertex v, denote by $Z_t(v)$ the number of descendants of v that are t levels lower, and by $S_t(v)$ the sum of spins of descendants of v that are t levels lower. Let $\mathcal{F}_j$ reveal the tree and its spins through depth j.

For simplicity, denote

$$Q_s = ((C^{(s)})^2\boldsymbol{x})(o) \quad \text{and} \quad R_s = (C^{(s)}(D-I)C^{(s)}\boldsymbol{x})(o)\,,$$

Let us denote

$$M_t = (C^{(t)}\boldsymbol{x})(o) = \mu^{-t}S_t(o), \qquad F_s = \sum_{j=0}^{s-1}\rho^{-j}M_{s-j}.$$

We shall show that

$$\mathbb{E}(Q_s - F_s)^2 \lesssim \rho^{-s}, \tag{9}$$
$$\mathbb{E}(R_s - dF_s - M_s)^2 \lesssim \rho^{-s}, \tag{10}$$
$$\mathbb{E}(F_s - cM_s)^2 \lesssim \rho^{-s}. \tag{11}$$

Given this, (a) and (b) for $s' = s$ follow immediately.

The proof of the above bounds is relatively simple—the expression for $((C^{(s)})^2\boldsymbol{x})(o)$ involves a sum over vertices reachable from o by the concatenation of two nonbacktracking paths, each of length s. The depth at which a given concatenation ends depends solely on the size of the intersection of these two paths, that is, the part that is "retraced."

Let us focus on (9). Summing over the length of the retraced part of the path,

$$\mu^{2s}Q_s = S_{2s}(o) + \sum_{j=1}^{s-1}\sum_{|z|=s-j} L_{j-1,s-j-1}(z) + Z_s(o)\boldsymbol{x}_o, \tag{12}$$

where $|z| = \mathsf{dist}(o,z)$.

The martingale argument from the proof of Lemma 6.7, gives, for $b \geqslant a$,

$$\mathbb{E}(M_b - M_a)^2 = \sum_{k=a+1}^{b}\rho^{-k} \leqslant \frac{\rho^{-a}}{\rho-1}, \qquad \sup_t \mathbb{E}\,M_t^2 \leqslant c. \tag{13}$$

This allows us to replace $S_{2s}(o)$ with $\mu^s S_s(o)$.

Simplifying the other terms requires calculations that have essentially already been performed in [BLM18]—let us describe the upshot thereof. Denote

$$L_{p,q}(z) = \sum_{\substack{a,b \text{ children of } z \\ a \neq b}} Z_p(a) S_q(b).$$

The proof of [BLM18, Theorem 25] gives

$$\begin{aligned} \mathbb{E}[L_{p,q}(z) \mid \boldsymbol{x}_z] &= d^{p+1} \mu^{q+1} \boldsymbol{x}_z, \\ \mathbb{E}[L_{p,q}(z)^2 \mid \boldsymbol{x}_z] &\lesssim d^{2p} \mu^{2q}. \end{aligned}$$

It follows that, for $1 \leqslant j < s$,

$$\mathbb{E}\left[\mu^{-2s} \sum_{|z|=s-j} L_{j-1,s-j-1}(z) \,\middle|\, \mathcal{F}_{s-j} \right] = \mu^{-2s} d^j \mu^{s-j} S_{s-j}(o) = \rho^{-j} M_{s-j}.$$

On the other hand, using independence of the subtrees of different vertices at the same depth, we may replace the conditional variance of the sum with the sum of the conditional variances. That is,

$$\begin{aligned} &\mathbb{E}\left(\mu^{-2s} \sum_{|z|=s-j} L_{j-1,s-j-1}(z) - \rho^{-j} M_{s-j} \right)^2 \\ &\quad = \mu^{-4s}\, \mathbb{E}\left[\operatorname{Var}\left(\sum_{|z|=s-j} L_{j-1,s-j-1}(z) \,\middle|\, \mathcal{F}_{s-j} \right) \right] \\ &\quad \lesssim \mu^{-4s} d^{s-j} d^{2j} \mu^{2(s-j)} = \rho^{-(s+j)}. \end{aligned}$$

We also have $\mathbb{E}(\mu^{-2s} Z_s(o))^2 \lesssim \rho^{-2s}$. Taking square roots of the variance bounds and summing over j gives a total error

$$O\left(\sum_{j=1}^{s-1} \rho^{-(s+j)/2} \right) = O(\rho^{-s/2}).$$

Substituting these bounds in (12) allows us to conclude that

$$\mathbb{E}\left(Q_s - \underbrace{\sum_{j=0}^{s-1} \rho^{-j} M_{s-j}}_{F_s} \right)^2 \lesssim \rho^{-s},$$

as desired.

The second bound (10) is very similar. Indeed, observing that $Z_1(u) S_s(u) = S_s(u) + L_{0,s-1}(u)$,

$$\mu^{2s} R_s = \sum_{|u|=s} Z_1(u) S_s(u) + \sum_{j=1}^{s-1} \sum_{|z|=s-j} L_{j,s-j-1}(z) + Z_{s+1}(o) \boldsymbol{x}_o$$

$$= S_{2s}(o) + \sum_{j=0}^{s-1} \sum_{|z|=s-j} L_{j,s-j-1}(z) + Z_{s+1}(o)\boldsymbol{x}_o.$$

We may therefore perform essentially the same manipulations to prove (10). We omit the details.

Finally, let us prove (11). By (13) and the triangle inequality,

$$\mathbb{E}(F_s - cM_s)^2 = \mathbb{E}\left(\sum_{j=0}^{s-1} \rho^{-j}(M_{s-j} - M_s) - \frac{\rho^{-s}}{1-\rho^{-1}} M_s\right)^2$$
$$\lesssim \left(\sum_{j=0}^{s-1} \rho^{-j} \sqrt{\frac{\rho^{-(s-j)}}{\rho - 1}} + \rho^{-s}\right)^2 \lesssim \rho^{-s}.$$

This proves (a) and (b) for $s' = s$.

When $s' \in \{s-1, s+1\}$, by Lemma 6.7, $\|\boldsymbol{\sigma}^{(s')} - \boldsymbol{\sigma}^{(s)}\| \to 0$. Using the Cauchy–Schwarz inequality allows us to transfer this estimate over to conclude that $\mathbb{E}\left\|C^{(s)}\left(\boldsymbol{\sigma}^{(s')} - \boldsymbol{\sigma}^{(s)}\right)\right\|^2 \to 0$ and $\mathbb{E}\left\|C^{(s)}(D-I)\left(\boldsymbol{\sigma}^{(s')} - \boldsymbol{\sigma}^{(s)}\right)\right\|^2 \to 0$. Combining this with the case $s' = s$ proves (a) and (b). ■

It remains to prove (c). This will require us to prove the following lemma, which yields a loose $O(1)$ bound when looking at $P\boldsymbol{x}$ for P being not just $C^{(s)}$, but the products of many such matrices.

Lemma A.1. *Let P be a product of p matrices, each of which is either one of the $\{C^{(j)} : j \geqslant 0\}$ or one of $\{D, D-I\}$. Then, there is a constant $K = K_{p,d,\lambda} < \infty$ such that $\|P\boldsymbol{x}\| \leqslant K$.*

Proof. We are interested in $\mathbb{E}\left[(P\boldsymbol{x})(o)^2\right]$. For starters, we may assume that all the matrices are either $D - \mathrm{Id}$ or one of the $\{C^{(j)}\}$, by replacing any occurrences of D with $(D - \mathrm{Id}) + \mathrm{Id}$. Let us understand what $(P\boldsymbol{x})(o)$ looks like. For example, consider $(C^{(2)} \cdot (D - \mathrm{Id}) \cdot C^{(3)}\boldsymbol{x})(o)$. We can then express this as a summation over concatenations of two non-backtracking paths, the first of length 2 and the second of length 3. For each such concatenation, the summand is the product of the spin at the terminal vertex with the number of children of the intermediate vertex.

More generally, $(P\boldsymbol{x})(o)$ is a summation over concatenations of non-backtracking paths of some prescribed length, with each summand being the products of the terminal spin, with the number of children of some subcollection of intermediate vertex.

When looking at $(P\boldsymbol{x})(o)^2$, we multiply two such terms. Since each degree term appears a constant number of times (with the constant depending on p), and the marginal degree distribution of each vertex is $\mathrm{Poi}(d)$, we can essentially replace the contributions of each of these terms with an $O_p(d)$, resulting in at most a $d^{O(p)}$ term that we may absorb into K—in other words, we may in fact assume that all the matrices in P are one of the $\{C^{(j)}\}$. On the other hand, the expectation of the product of two spins at distinct vertices v, w is given by $\lambda^{\mathsf{dist}(v,w)}$.

Let us carry out this calculation for $P = C^{(s)}$. The two walks involved in the expectation are then length s walks down the tree having some intersection size j (for some parameter j that we will eventually sum over)—in expectation, there are $d^s \cdot d^{s-j}$ such pairs. The normalization of $C^{(s)}$ adds an additional factor of $(d\lambda)^{-2s}$. Finally, the two endpoints being distance $2(s-j)$ apart results

in a λ^{2s-2j} multiplicative term. That is, $\mathbb{E}(P\boldsymbol{x})(o)^2$ is, up to a constant, $\sum_{j\geqslant 0} d^{2s-j}\cdot\lambda^{2s-2j}\cdot(d\lambda)^{-2s} = \sum_{j\geqslant 0}\rho^{-j} < \infty$. Such calculations are very similar to those executed in [BLM18].

The general computation when $P = \prod_{i=1}^{p} C^{(s_i)}$ is very similar. In this case, we can sum over $(r_j)_{1\leqslant j\leqslant 2p}$, where r_i is the number of edges in the ith walk that are contained in the part of the tree that have already been "exposed" so far by the previous walks. The number of such p-tuples of vertices is then $d^{2\sum_{i=1}^{p} s_i - \sum_{j=1}^{2p} r_j}$. The normalization of the $C^{(s_i)}$ yields an additional $(\lambda d)^{-2\sum_{i=1}^{p} s_i}$ term. Finally, the spin correlation can be bounded from above by $\lambda^{2\sum_{i=1}^{p} s_i - 2\sum_{j=1}^{2p} r_j}$. Summing over (r_j) yields

$$\begin{aligned}\mathbb{E}(P\boldsymbol{x})(o)^2 &\leqslant \sum_{(r_j)} d^{2\sum_{i=1}^{p} s_i - \sum_{j=1}^{2p} r_j}\cdot(\lambda d)^{-2\sum_{i=1}^{p} s_i}\cdot\lambda^{2\sum_{i=1}^{p} s_i - 2\sum_{j=1}^{2p} r_j}\\ &= \sum_{(r_j)}\rho^{-\sum_{j=1}^{2p} r_j} = \left(\sum_{r\geqslant 0}\rho^{-r}\right)^{2p} < \infty\,.\end{aligned}$$

We omit the details of the proof when P involves $D - I$ terms—as mentioned earlier, this only results in an additional $d^{O(p)}$ multiplicative term. ∎

Given the above, proving (c) is relatively easy.

Proof of Lemma 6.9(c). Set

$$e_s = C^{(s)}\boldsymbol{\sigma}^{(s)} - c\boldsymbol{\sigma}^{(s)} = \big((C^{(s)})^2 - cC^{(s)}\big)\boldsymbol{x}.$$

By (a), $\|e_s\| \to 0$. By Lemma 6.4, the operators $C^{(j)}$, D, and H are self-adjoint. As a result,

$$\|C^{(s)}He_s\|^2 = \langle e_s, H(C^{(s)})^2He_s\rangle \leqslant \|e_s\|\cdot\|H(C^{(s)})^2He_s\|.$$

Note that the second vector is some bounded linear combinations of vectors Px, where P is a product of finitely many terms of the form $C^{(j)}$, D, or $D - I$, exactly as in Lemma A.1. Therefore, we may bound the second term by $O(1)$, uniformly in s, and conclude that

$$\|C^{(s)}He_s\|^2 \lesssim \|e_s\| \to 0\,.$$ ∎

B Deferred proofs for fast implementation

In this appendix, we prove multiple lemmas used in the SDP implementation: Theorem 8.3 and Lemmas 8.6, 8.8 and 8.9.

Theorem 8.3 (Nearby-point entropy descent)**.** *Let F be convex on all real symmetric matrices and attain its minimum over $\mathcal{S}$. Let $\eta > 0$, let $T \geqslant 1$ be an integer, and let $H_1, \ldots, H_T \in \mathbb{R}^{n\times n}$ be symmetric matrices. For $1 \leqslant t \leqslant T$, define*

$$S_t = -\eta\sum_{s<t} H_s, \qquad X_t = \frac{e^{S_t}}{\operatorname{Tr} e^{S_t}}.$$

Suppose $\widetilde{X}_1, \dots, \widetilde{X}_T \in \mathcal{S}$ satisfy $H_t \in \partial F(\widetilde{X}_t)$ for $1 \leqslant t \leqslant T$. Then $\bar{X} = T^{-1}\sum_{t=1}^{T} \widetilde{X}_t$ satisfies

$$F(\bar{X}) - \min_{\mathcal{S}} F \leqslant \frac{\log n}{\eta T} + 2\eta \max_{1 \leqslant t \leqslant T} \|H_t\|_{\mathrm{op}}^2 + \frac{1}{T}\sum_{t=1}^{T} \langle H_t, \widetilde{X}_t - X_t \rangle.$$

The following proof is adapted from the matrix multiplicative-weights potential argument of Arora, Hazan, and Kale [AHK12, Section 5].

Proof. Put $R = \max_t \|H_t\|_{\mathrm{op}}$, $a_t = \|H_t\|_{\mathrm{op}}$, and $K_t = H_t + a_t I$, so $0 \preceq K_t \preceq 2a_t I$. The scalar inequality $e^{-u} \leqslant 1 - u + u^2/2$ for $u \geqslant 0$ gives $e^{-\eta K_t} \preceq I - \eta K_t + \eta^2 K_t^2/2$. Golden–Thompson and $\log x \leqslant x - 1$ give

$$\begin{aligned} \log \operatorname{Tr} e^{S_t - \eta H_t} - \log \operatorname{Tr} e^{S_t} &\leqslant \eta a_t + \log \operatorname{Tr}(X_t e^{-\eta K_t}) \\ &\leqslant \eta a_t - \eta \operatorname{Tr}(X_t K_t) + \tfrac{\eta^2}{2} \operatorname{Tr}(X_t K_t^2) \leqslant -\eta \langle H_t, X_t \rangle + 2\eta^2 a_t^2. \end{aligned}$$

For every $U \in \mathcal{S}$ and symmetric B, $\log \operatorname{Tr} e^B \geqslant \langle B, U \rangle$. Telescoping from $S_1 = 0$ therefore gives

$$\frac{1}{T}\sum_t \langle H_t, X_t - U \rangle \leqslant \frac{\log n}{\eta T} + 2\eta R^2.$$

The support inequality at $\widetilde{X}_t$ is

$$F(\widetilde{X}_t) - F(U) \leqslant \langle H_t, \widetilde{X}_t - U \rangle = \langle H_t, X_t - U \rangle + \langle H_t, \widetilde{X}_t - X_t \rangle.$$

Sum, apply convexity at $\bar{X}$, and choose a minimizer U. ∎

Lemma 8.6 (Approximate matrix exponential)**.** *Let $S \in \mathbb{R}^{n \times n}$ be symmetric with $\|S\|_{\mathrm{op}} \leqslant L$, where $L \geqslant 0$, and let $0 < \alpha < 1$. Set $m = \lceil 8(L + \log(3/\alpha) + 1) \rceil$ and*

$$Q = \sum_{j=0}^{m-1} \frac{(S/2)^j}{j!}.$$

Then

$$(1 - \alpha) e^S \preceq Q^2 \preceq (1 + \alpha) e^S.$$

If one product with S costs T_S operations, then, for any $u \in \mathbb{R}^n$, computing Qu costs $O((L + \log(3/\alpha) + 1)(T_S + n))$ operations.

For a related Taylor-expansion tail estimate, see [SV13, Proof of Lemma 4.2, p. 151].

Proof. Set $\delta = \alpha/3$ and $p(x) = \sum_{j=0}^{m-1} x^j/j!$. If $L = 0$, then $S = 0$ and $Q = I$, so the sandwich is immediate. For $L > 0$ and $|x| \leqslant L/2$, the scalar Taylor remainder gives

$$\frac{|e^x - p(x)|}{e^x} \leqslant e^L \frac{(L/2)^m}{m!} \leqslant e^L \left(\frac{eL}{2m}\right)^m \leqslant e^{L-m} \leqslant \delta,$$

using $m! \geqslant (m/e)^m$, $m \geqslant 8L$, $e/16 < e^{-1}$, and $m \geqslant L + \log(1/\delta)$. By orthogonal diagonalization of S, the scalar bounds $1 - \delta \leqslant p(x)/e^x \leqslant 1 + \delta$ imply

$$(1 - \delta)^2 e^S \preceq Q^2 \preceq (1 + \delta)^2 e^S.$$

Since $0 < \alpha < 1$, $(1-\delta)^2 \geqslant 1-\alpha$ and $(1+\delta)^2 \leqslant 1+\alpha$, proving the claimed Löwner-order sandwich. Finally, compute $w_0 = u$, $w_j = Sw_{j-1}/(2j)$ for $1 \leqslant j < m$, and sum $Qu = \sum_{j=0}^{m-1} w_j$. This costs $O(m(T_S + n))$ exact-real operations, as claimed. ■

Lemma 8.8 (Exponential matrix density sketch)**.** *There is a randomized algorithm that, given a symmetric matrix $S \in \mathbb{R}^{n\times n}$ through a matrix–vector product oracle, a bound $L \geqslant \|S\|_{\mathrm{op}}$, and $0 < \alpha < 1$, returns $\boldsymbol{Z} \in \mathbb{R}^{n\times k}$ with $\|\boldsymbol{Z}\|_F = 1$ and $k \leqslant O(\alpha^{-2})$, such that $\widetilde{X} = \boldsymbol{Z}\boldsymbol{Z}^T \in \mathcal{S}$ satisfies, for every symmetric $A \in \mathbb{R}^{n\times n}$ and every $P \geqslant 0$,*

$$\mathbb{E} \sup_{\|d\|_\infty \leqslant P} \left| \left\langle A + \mathrm{Diag}(d), \widetilde{X} - X \right\rangle \right| \leqslant \alpha(\|A\|_{\mathrm{op}} + P),$$

where $X = e^S / \operatorname{Tr} e^S$. The algorithm uses $O(\alpha^{-2}(1 + L + \log(1/\alpha)))$ matrix-vector products with S and n times as many additional operations.

Proof. Set $\delta = \alpha/4$ and $k = \lceil 32/\alpha^2 \rceil$. Take Q from Lemma 8.6 at accuracy δ, draw independent uniform signs $G \in \{-1,1\}^{n\times k}$, and return $\boldsymbol{Z} = QG/\|QG\|_F$. Write $B = Q^2$, $a = \operatorname{Tr} B$, $W = QGG^TQ/k$, $b = \operatorname{Tr} W$, $C = B/a$, and $Y = \boldsymbol{Z}\boldsymbol{Z}^T = W/b = \widetilde{X}$. The Löwner-order sandwich makes Q invertible, so $\boldsymbol{Z} \neq 0$ and $b > 0$. For fixed A, P, put $h = \|A\|_{\mathrm{op}} + P$ and

$$N(D) = |\langle A, D\rangle| + P\|\operatorname{diag} D\|_1 = \sup_{\|d\|_\infty \leqslant P} |\langle A + \mathrm{Diag}(d), D\rangle|.$$

The equality holds pointwise, allowing d to depend on G. With $r = \sqrt{2/k}$, Lemma 8.7, applied with $V = Q$ and summed over the coordinate basis, gives $\mathbb{E}\|\operatorname{diag}(W - B)\|_1 \leqslant ra$ and $\mathbb{E}|b - a| \leqslant ra$. Summing the same estimate over a fixed orthonormal eigenbasis of A with weights equal to its absolute eigenvalues gives $\mathbb{E}|\langle A, W - B\rangle| \leqslant ra\|A\|_{\mathrm{op}}$. Since Y is a density matrix, $N(Y) \leqslant h$; hence the identity

$$Y - C = (W - B)/a + (a - b)Y/a$$

yields $\mathbb{E}N(Y - C) \leqslant 2rh$. Put $E = e^S$ and $t = \operatorname{Tr} E$. The sandwich and simultaneous diagonalization of B, E give $\|B - E\|_* \leqslant \delta t$ and $|a - t| \leqslant \delta t$. Thus $C - X = (B - E)/t + (t - a)C/t$ implies $\|C - X\|_* \leqslant 2\delta$, and consequently $N(C - X) \leqslant 2\delta h$. Therefore $\mathbb{E}N(Y - X) \leqslant 2(r + \delta)h \leqslant \alpha h$. Finally, the degree bound in Lemma 8.6 at accuracy δ gives the claimed product and additional-operation counts for the k columns, with $O(nk)$ further operations to compute $\|QG\|_F$ and scale QG; neither a nor t is computed. ■

Lemma 8.9 (Ball distance oracle)**.** *There is a deterministic algorithm that, given $K \geqslant 1$ and $p \in \mathbb{R}^n$, computes $y \in \partial f(p)$, where $f(x) = \mathrm{dist}_1(x, \mathcal{D}_K)$ for $x \in \mathbb{R}^n$. The algorithm uses $O(n \log n)$ arithmetic operations.*

Proof. If $p \in \mathcal{D}_K$, return $y = 0$. Otherwise find $t > 0$ satisfying that, if we truncate coordinates of p to magnitude at most t, the resulting vector p^* has squared ℓ_2 norm exactly K/n, i.e.,

$$p_i^* = \operatorname{sgn}(p_i)\min\{|p_i|, t\}, \qquad \sum_i (p_i^*)^2 = K/n.$$

Let $y = p^*/t$, we will show that $y \in \partial f(p)$.

Consider arbitrary $z \in \mathbb{R}^n$ and its closest point $z^* \in \mathcal{D}_K$. By the definition of y, we have $y^\top p^* \geqslant y^\top z^*$ and $\|y\|_\infty \leqslant 1$. Therefore, it follows that

$$y^\top (z - p^*) \leqslant y^\top (z - z^*) \leqslant \|z - z^*\|_1 = f(z).$$

Since $y_i = \operatorname{sgn}(p_i)$ for $|p_i| > t$ and $p - p^*$ is supported on those coordinates, we have $y^\top (p - p^*) = \|p - p^*\|_1$. Hence,

$$f(z) \geqslant y^\top (z - p^*) = y^\top (z - p) + y^\top (p - p^*) \geqslant y^\top (z - p) + \|p - p^*\|_1 \geqslant y^\top (z - p) + f(p).$$

To find t, we sort the coordinate magnitudes of p and scan the sorted list to locate the clipping threshold. Sorting takes $O(n \log n)$ time, and the scan takes $O(n)$ time. ■