Exact Community Recovery in the Geometric SBM

Julia Gaudio* Xiaochun Niu† Ermin Wei‡

Abstract

We study the problem of exact community recovery in the Geometric Stochastic Block Model (GSBM), where each vertex has an unknown community label as well as a known position, generated according to a Poisson point process in $\mathbb{R}^d$. Edges are formed independently conditioned on the community labels and positions, where vertices may only be connected by an edge if they are within a prescribed distance of each other. The GSBM thus favors the formation of dense local subgraphs, which commonly occur in real-world networks, a property that makes the GSBM qualitatively very different from the standard Stochastic Block Model (SBM). We propose a linear-time algorithm for exact community recovery, which succeeds down to the information-theoretic threshold, confirming a conjecture of Abbe, Baccelli, and Sankararaman. The algorithm involves two phases. The first phase exploits the density of local subgraphs to propagate estimated community labels among sufficiently occupied subregions, and produces an almost-exact vertex labeling. The second phase then refines the initial labels using a Poisson testing procedure. Thus, the GSBM enjoys *local to global amplification* just as the SBM, with the advantage of admitting an information-theoretically optimal, linear-time algorithm.

1 Introduction

Community detection is the problem of identifying latent community structure in a network. In 1983, Holland, Laskey, and Leinhardt [20] introduced the *Stochastic Block Model* (SBM), a probabilistic model which generates graphs with community structure, where edges are generated independently conditioned on community labels. Since then, the SBM has been intensively studied in the probability, statistics, machine learning, and information theory communities. Many community recovery problems are now well-understood; for example, the fundamental limits of the exact recovery problem are known, and there is a corresponding efficient algorithm that achieves those limits [5]. For an overview of theoretical developments and open questions, please see the survey of Abbe [1].

While the SBM is a powerful model, its simplicity fails to capture certain properties that occur in real-world networks. In particular, social networks typically contain many triangles; a given pair of people are more likely to be friends if they already have a friend in common [27]. The SBM by its very nature does not capture this transitive behavior, since edges are formed independently, conditioned on the community assignments. To address this shortcoming, Baccelli and Sankararaman [29] introduced a spatial random graph model, which we refer to as the Geometric Stochastic Block Model (GSBM). In the GSBM, vertices are generated according to a Poisson point process in a bounded region of $\mathbb{R}^d$. Each vertex is randomly assigned one of two community labels, with equal probability. A given pair of vertices (u, v) is connected by an edge with a probability that depends on both the community labels of u and v as well as their distance. Edges are formed independently, conditioned on the community assignments and locations. The geometric embedding thus governs the transitive edge behavior. The goal is to determine the communities of the vertices, observing the edges and the locations. In a follow-up work, Abbe, Sankararaman, and Baccelli [2] studied both partial recovery in sparse graphs, as well as exact recovery in logarithmic-degree graphs. Their work established a phase transition for both partial and exact recovery, in terms of the Poisson intensity parameter λ. The critical value of λ was identified in some special cases of the sparse model, but a precise characterization of the information-theoretic threshold for exact recovery in the logarithmic regime was left open.

Our work resolves this gap, by identifying the information-theoretic threshold for exact recovery in the logarithmic degree regime (and confirming a conjecture of Abbe et al [2]). Additionally, we propose a polynomial-time algorithm achieving the information-theoretic threshold. The algorithm consists of two phases: the first

*(julia.gaudio@northwestern.edu) Department of Industrial Engineering and Management Sciences, Northwestern University
†(xiaochunniu2024@u.northwestern.edu) Department of Industrial Engineering and Management Sciences, Northwestern University
‡(ermin.wei@northwestern.edu) Department of Electrical and Computer Engineering and Department of Industrial Engineering and Management Sciences, Northwestern University

phase produces a preliminary almost-exact labeling through a local label propagation scheme, while the second phase refines the initial labels to achieve exact recovery. At a high level, the algorithm bears some similarity to prior works on the SBM using a two-phase approach [5, 25]. Our work therefore shows that just like the SBM, the GSBM exhibits the so-called *local to global amplification* phenomenon [1], meaning that exact recovery is achievable whenever the probability of misclassifying an individual vertex, given the labels of the remaining $n-1$ vertices, is $o(1/n)$. However, the GSBM is qualitatively very different from the SBM, and it is not apparent at the outset that it should exhibit local to global amplification. In particular, the GSBM is not a low-rank model, suggesting that approaches such as spectral methods [2] and semidefinite programming [17], which exploit the low-rank structure of the SBM, may fail in the GSBM. In order to achieve almost exact recovery in the GSBM, we instead use the density of local subgraphs to propagate labels. Our propagation scheme allows us to achieve almost exact recovery, and also ensures that no local region has too many misclassified vertices. The dispersion of errors is crucial to showing that labels can be correctly refined in the second phase.

Notably, our algorithm runs in linear time (where the input size is the number of edges). This is in contrast with the SBM, for which no statistically optimal linear-time algorithm for exact recovery has been proposed. To our knowledge, the best-known runtime for the SBM in the logarithmic degree regime is achieved by the spectral algorithm of Abbe et al [4], which runs in $O(n\log^2 n)$ time, while the number of edges is $\Theta(n\log n)$. More recent work of Cohen–Addad et al [11] proposed a linear-time algorithm for the SBM, but the algorithm was not shown to achieve the information-theoretic threshold for exact recovery. Intuitively, the strong local interactions in the GSBM enable more efficient algorithms than what seems to be possible in the SBM.

Notation and organization. We write $[n] = \{1, \cdots, n\}$. We use Bachmann–Landau notation with respect to the parameter n; i.e. $o(1)$ means $o_n(1)$. Bin denotes the binomial distribution. For $\mu \in \mathbb{R}^m$, Poisson(μ) denotes the m-type Poisson distribution.

The rest of the paper is organized as follows. Section 2 describes the exact recovery problem as well as our main result (Theorem 2.2). The exact recovery algorithm is given in Section 3, along with an outline of the proof of exact recovery. Sections 4 and 5 include the proofs of the two phases of the algorithm. Section 6 contains the proof of impossibility (Theorem 2.3) (a slight generalization of [2, Theorem 3.7] to cover the disassortative case). Section 7 includes additional related work. We conclude with future directions in Section 8.

2 Model and main results

We now describe the GSBM in the logarithmic degree regime, where edges are formed only between sufficiently close vertices, as proposed in [2, 29].

DEFINITION 2.1. *Let $\lambda > 0$, $a, b \in [0,1]$, and $a \neq b$ be constants, and let $d \in \mathbb{N}$. A graph G is sampled from GSBM(λ, n, a, b, d) according to the following steps:*

1. *The locations of vertices are determined according to a homogeneous Poisson point process[1] with intensity λ in the region $\mathcal{S}_{d,n} := [-n^{1/d}/2, n^{1/d}/2]^d \subset \mathbb{R}^d$. Let $V \subset \mathcal{S}_{d,n}$ denote the vertex set.*
2. *Community labels are generated independently. The ground truth label of vertex $u \in V$ is given by $\sigma_0(u) \in \{-1, 1\}$, with $\mathbb{P}(\sigma_0(u) = 1) = \mathbb{P}(\sigma_0(u) = -1) = 1/2$.*
3. *Conditioned on the locations and community labels, edges are formed independently. Letting E denote the edge set, for $u, v \in V$ and $u \neq v$, we have*

$$\mathbb{P}(\{u,v\} \in E) = \begin{cases} a & \text{if } \sigma_0(u) = \sigma_0(v),\ \|u-v\| \le (\log n)^{1/d} \\ b & \text{if } \sigma_0(u) \neq \sigma_0(v),\ \|u-v\| \le (\log n)^{1/d} \\ 0 & \text{if } \|u-v\| > (\log n)^{1/d}. \end{cases}$$

The graph does not contain self-loops. Here $\|u-v\|$ denotes the toroidal metric:

$$\|u-v\| = \left\| \min\left\{|u_i - v_i|, n^{1/d} - |u_i - v_i|\right\}, \ldots, \min\left\{|u_d - v_d|, n^{1/d} - |u_d - v_d|\right\} \right\|_2,$$

where $\|\cdot\|_2$ is the standard Euclidean metric.

[1] The definition and construction of a homogeneous Poisson point process are provided in Definition 4.1.

In other words, a given pair of vertices can only be connected by an edge if they are within a distance of $(\log n)^{1/d}$; in that case, we say they are *mutually visible*. When a pair of vertices are mutually visible, the probability of being connected by an edge depends on their community labels, as in the standard SBM. Observe that any unit volume region has Poisson(λ) vertices (and hence λ vertices in expectation). In particular, the expected number of vertices in the region $\mathcal{S}_{d,n}$ is λn.

Given an estimator $\widetilde{\sigma} = \widetilde{\sigma}_n$, we define $A(\widetilde{\sigma}, \sigma_0) = \max_{s\in\{\pm 1\}}(\sum_{u\in V} \mathbb{1}_{\widetilde{\sigma}(u)=s\sigma_0(u)})/|V|$ as the agreement of $\widetilde{\sigma}$ and σ_0. We define some recovery requirements including *exact recovery* as follows.

- *Exact recovery:* $\lim_{n\to\infty} \mathbb{P}(A(\widetilde{\sigma}, \sigma_0) = 1) = 1$,
- *Almost exact recovery:* $\lim_{n\to\infty} \mathbb{P}(A(\widetilde{\sigma}, \sigma_0) \geq 1 - \epsilon) = 1$, for all $\epsilon > 0$,
- *Partial recovery:* $\lim_{n\to\infty} \mathbb{P}(A(\widetilde{\sigma}, \sigma_0) \geq \alpha) = 1$, for some $\alpha > 1/2$.

In other words, an exact recovery estimator must recover all labels (up to a global sign flip), with probability tending to 1 as the graph size goes to infinity. Abbe et al [2] identified an impossibility regime for the exact recovery problem. Here, ν_d is the volume of a unit Euclidean ball in d dimensions.

THEOREM 2.1. (THEOREM 3.7 IN [2]) *Let $\lambda > 0$, $d \in \mathbb{N}$, and $0 \leq b < a \leq 1$ satisfy*

$$\lambda\nu_d(1 - \sqrt{ab} - \sqrt{(1-a)(1-b)}) < 1, \tag{2.1}$$

and let $G \sim GSBM(\lambda, n, a, b, d)$. Then any estimator $\widetilde{\sigma}$ fails to achieve exact recovery.

Abbe et al [2] conjectured that the above result is tight, but only established that exact recovery is achievable for $\lambda > \lambda(a, b, d)$ sufficiently large [2, Theorem 3.9]. In this regime, [2] provided a polynomial-time algorithm based on the observation that the relative community labels of two nearby vertices can be determined with high accuracy by counting their common neighbors. By taking $\lambda > 0$ large enough to drive up the density of points, the failure probability of pairwise classification can be taken to be an arbitrarily small inverse polynomial in n.

Our main result is a positive resolution to [2, Conjecture 3.8] (with a slight modification for the case $d = 1$, noting that $\nu_1 = 2$).

THEOREM 2.2. (ACHIEVABILITY) *There exists a polynomial-time algorithm achieving exact recovery in $G \sim GSBM(\lambda, n, a, b, d)$ whenever*

1. *$d = 1$, $\lambda > 1$, $a, b \in [0, 1]$, and $2\lambda(1 - \sqrt{ab} - \sqrt{(1-a)(1-b)}) > 1$; or*
2. *$d \geq 2$, $a, b \in [0, 1]$, and $\lambda\nu_d(1 - \sqrt{ab} - \sqrt{(1-a)(1-b)}) > 1$.*

We drop the requirement that $a > b$ in Theorem 2.1, thus covering the disassortative case. We additionally expand the impossible regime for $d = 1$, compared to Theorem 2.1.

THEOREM 2.3. (IMPOSSIBILITY) *Let $\lambda > 0$, $d \in \mathbb{N}$, and $a, b \in [0, 1]$ satisfy (2.1) and let $G \sim GSBM(\lambda, n, a, b, d)$. Then any estimator $\widetilde{\sigma}$ fails to achieve exact recovery. Additionally, if $d = 1$ and $\lambda < 1$, then any estimator $\widetilde{\sigma}$ fails to achieve exact recovery.*

Putting Theorems 2.2 and 2.3 together establishes the information-theoretic threshold for exact recovery in the GSBM, and shows that recovery is efficiently achievable above the threshold. We remark that the condition $\lambda\nu_d(1 - \sqrt{ab} - \sqrt{(1-a)(1-b)}) > 1$ in Theorem 2.2 is equivalent to $D_+(x\|y) > 1$, where $D_+(x\|y)$ is the Chernoff–Hellinger (CH) divergence [5] between the vectors $x = \lambda\nu_d[a, 1-a, b, 1-b]/2$ and $y = \lambda\nu_d[b, 1-b, a, 1-a]/2$. As we will show, the exact recovery problem can be reduced to a multitype Poisson hypothesis testing problem; the CH-divergence condition characterizes the parameters for which the hypothesis test is successful.

Abbe et al [2] suggested that the threshold given by Theorem 2.1 might be achieved by a two-round procedure reminiscent of the exact recovery algorithm for the SBM developed by Abbe and Sandon [5]. Indeed, our algorithm is a two-round procedure, but the details of the first phase (achieving almost exact recovery) are qualitatively very different from the strategy employed in the standard SBM. At a high level, our algorithm spreads vertex label information locally by exploiting the density of local subgraphs. The information is spread by iteratively labeling

"blocks", labeling a given block by using a previously labeled block as a reference. To ensure that the algorithm spreads label information to all (sufficiently dense) blocks, we establish a connectivity property of the dense blocks that holds with high probability whenever $\lambda\nu_d > 1$ ($\lambda > 1$ if $d = 1$). This is in contrast to the Sphere Comparison algorithm [5] for the SBM, where the relative labels of a pair of vertices u, v are determined by comparing their neighborhoods.

The algorithm in Phase I in fact achieves almost exact recovery for a wider range of parameters than what is required to achieve exact recovery.

THEOREM 2.4. *There is a polynomial-time algorithm achieving almost exact recovery in $G \sim GSBM(\lambda, n, a, b, d)$ whenever*

1. *$d = 1$, $\lambda > 1$, and $a, b \in [0, 1]$ with $a \neq b$; or*
2. *$d \geq 2$, $\lambda\nu_d > 1$, and $a, b \in [0, 1]$ with $a \neq b$.*

3 Exact recovery algorithm

This section presents our algorithm, which consists of two phases. In Phase I, our goal is to estimate an almost-exact labeling $\widehat{\sigma}\colon V \to \{-1, 0, 1\}$, where the label 0 indicates uncertainty. Phase I is based on the following observation: for any $\delta > 0$, if we know the true labels of some $\delta \log n$ vertices visible to a given vertex v, then by computing edge statistics, we can determine the label of v with probability $1 - n^{-c}$, for some $c(\delta) > 0$. In Phase I, we partition the region into hypercubes of volume $\Theta(\log n)$ (called *blocks*), and show how to produce an almost exact labeling of all blocks that contain at least $\delta \log n$ vertices (called *occupied blocks*), by an iterative label propagation scheme. Next, Phase II refines the labeling $\widehat{\sigma}$ to $\widetilde{\sigma}$ using Poisson testing. Phase II builds upon a well-established approach in the SBM literature [5, 25], to refine an almost-exact labeling with dispersed errors into an exact labeling. The main novelty of our algorithm therefore lies in Phase I.

Before describing the algorithm, we introduce the notion of a *degree profile*.

DEFINITION 3.1. (DEGREE PROFILE) *Given $G \sim GSBM(\lambda, n, a, b, d)$, the* degree profile *of a vertex $u \in V$ with respect to a reference set $S \subset V$ and a labeling $\sigma\colon S \to \{-1, 1\}$ is given by the 4-tuple,*

$$d(u, \sigma, S) = \left[d_1^+(u, \sigma, S), d_1^-(u, \sigma, S), d_{-1}^+(u, \sigma, S), d_{-1}^-(u, \sigma, S)\right],$$

where

$$\begin{aligned}
d_1^+(u, \sigma, S) &= |\{v \in S\colon \sigma(v) = 1, (u, v) \in E\}|,\\
d_1^-(u, \sigma, S) &= |\{v \in S\colon \sigma(v) = 1, (u, v) \notin E, \|u - v\| \leq (\log n)^{1/d}\}|,\\
d_{-1}^+(u, \sigma, S) &= |\{v \in S\colon \sigma(v) = -1, (u, v) \in E\}|,\\
d_{-1}^-(u, \sigma, S) &= |\{v \in S\colon \sigma(v) = -1, (u, v) \notin E, \|u - v\| \leq (\log n)^{1/d}\}|.
\end{aligned}$$

Note that we only consider v such that $\|u - v\| \leq (\log n)^{1/d}$, since we only want to count non-edges to vertices that are visible to u. For convenience, when V serves as the reference set, we write $d(u, \sigma) := d(u, \sigma, V)$ and $d(u, \sigma) := [d_1^+(u, \sigma), d_1^-(u, \sigma), d_{-1}^+(u, \sigma), d_{-1}^-(u, \sigma)]$.

3.1 Exact recovery for $d = 1$. We first describe the algorithm specialized to the case $d = 1$. Several additional ideas are required to move to the $d \geq 2$ case, to ensure uninterrupted propagation of label estimates over all occupied blocks. We first describe the simplest case where $d = 1$, $\lambda > 2$, and $a, b \in [0, 1]$ with $a \neq b$.

Algorithm for $\lambda > 2$. The algorithm is presented in Algorithm 1. In Phase I, we first partition the interval into blocks of length $\log n/2$ and define V_i as the set of vertices in the ith block for $i \in [2n/\log n]$. In this way, any pair of vertices in adjacent blocks are within a distance of $\log n$. The density $\lambda > 2$ ensures a high probability that all blocks have $\Omega(\log n)$ vertices, as we later show in (4.3). Next, we use the `Pairwise Classify` subroutine to label the first block (Line 3). Here, we select an arbitrary vertex $u_0 \in V_1$ and set $\widehat{\sigma}(u_0) = 1$. The labels of other vertices $u \in V_1$ are labeled by counting common neighbors with u_0, among the vertices in V_1. Next, the labeling of V_1 is propagated to other blocks V_i for $i \geq 2$ utilizing the edges between V_{i-1} and V_i and the estimated labeling on V_{i-1}, by thresholding degree profiles with respect to V_{i-1} according to Algorithm 3 (Lines 4-5). The reference set

S in Algorithm 3 plays the role of V_{i-1} and S' plays the role of V_i. Intuitively, if $a > b$, a vertex tends to exhibit more edges and fewer non-edges within its own community while having fewer edges and more non-edges with the other community. Conversely, if $a < b$, the opposite observation holds. In order to classify the vertices in V_i, we use edges from V_i to the larger set of $\{u \in V_{i-1} : \widehat{\sigma}(u) = 1\}$ and $\{u \in V_{i-1} : \widehat{\sigma}(u) = -1\}$, rather than using all edges between V_i and V_{i-1}, which simplifies the analysis. In Theorem 4.1, we will demonstrate that Phase I achieves almost-exact recovery on G under the conditions in Theorem 2.4.

Algorithm 1 Exact recovery for the GSBM ($d = 1$ and $\lambda > 2$)

Input: $G \sim \mathrm{GSBM}(\lambda, n, a, b, 1)$ where $\lambda > 2$.
Output: An estimated community labeling $\widetilde{\sigma} : V \to \{-1, 1\}$.

Phase I:
Partition the interval $[-n/2, n/2]$ into $2n/\log n$ blocks[2] of volume $\log n/2$ each. Let B_i be the ith block and V_i be the set of vertices in B_i for $i \in [2n/\log n]$.
Apply `Pairwise Classify` (Algorithm 2) on input G, V_1, a, b to obtain a labeling $\widehat{\sigma}$ of V_1.
for $i = 2, \cdots, 2n/\log n$ **do**
    Apply `Propagate` (Algorithm 3) on input G, V_{i-1}, V_i to determine the labeling $\widehat{\sigma}$ on V_i.
Phase II:
for $u \in V$ **do**
    Apply `Refine` (Algorithm 4) on input $G, \widehat{\sigma}, u$ to obtain $\widetilde{\sigma}(u)$.

Algorithm 2 `Pairwise Classify`

Input: Graph $G = (V, E)$, vertex set $S \subset V$, parameters $a, b \in [0, 1]$ with $a \neq b$.
Choose an arbitrary vertex $u_0 \in S$, and set $\widehat{\sigma}(u_0) = 1$.
for $u \in S \setminus \{u_0\}$ **do**
    if $|\{v \in S \setminus \{u, u_0\} : \{u_0, v\}, \{u, v\} \in E\}| > (a + b)^2(|S| - 2)/4$ **then**
        Set $\widehat{\sigma}(u) = 1$.
    else
        Set $\widehat{\sigma}(u) = -1$.

Algorithm 3 `Propagate`

Input: Graph $G = (V, E)$, mutually visible sets of vertices $S, S' \subset V$ with $S \cap S' = \emptyset$, where S is labeled according to $\widehat{\sigma}$.
if $|\{v \in S \colon \widehat{\sigma}(v) = 1\}| \geq |\{v \in S \colon \widehat{\sigma}(v) = -1\}|$ **then**
    for $u \in S'$ **do**
        if $a > b$ and $d_1^+(u, \widehat{\sigma}, S) \geq (a + b) \cdot |\{v \in S \colon \widehat{\sigma}(v) = 1\}|/2$ **then**
            Set $\widehat{\sigma}(u) = 1$.
        else if $a < b$ and $d_1^+(u, \widehat{\sigma}, S) < (a + b) \cdot |\{v \in S \colon \widehat{\sigma}(v) = 1\}|/2$ **then**
            Set $\widehat{\sigma}(u) = 1$.
        else
            Set $\widehat{\sigma}(u) = -1$.
else
    for $u \in S'$ **do**
        if $a > b$ and $d_{-1}^+(u, \widehat{\sigma}, S) \geq (a + b) \cdot |\{v \in S \colon \widehat{\sigma}(v) = -1\}|/2$ **then**
            Set $\widehat{\sigma}(u) = -1$.
        else if $a < b$ and $d_{-1}^+(u, \widehat{\sigma}, S) < (a + b) \cdot |\{v \in S \colon \widehat{\sigma}(v) = -1\}|/2$ **then**
            Set $\widehat{\sigma}(u) = -1$.
        else
            Set $\widehat{\sigma}(u) = 1$.

[2]The number of blocks is $\lceil 2n/\log n \rceil$ if $2n/\log n$ is not an integer.

Algorithm 4 `Refine`

Input: Graph $G \sim \mathrm{GSBM}(\lambda, n, a, b, d)$, vertex $u \in V$, labeling $\widehat{\sigma} : V \to \{-1, 0, 1\}$.
Output: An estimated labeling $\widetilde{\sigma}(u) \in \{-1, 1\}$.

1: Set $\widetilde{\sigma}(u) = \mathrm{sign}\Big[\log\big(\frac{a}{b}\big)\big(d_1^+(u,\widehat{\sigma}) - d_{-1}^+(u,\widehat{\sigma})\big) + \log\big(\frac{1-a}{1-b}\big)\big(d_1^-(u,\widehat{\sigma}) - d_{-1}^-(u,\widehat{\sigma})\big)\Big]$.

In Phase II, we refine the almost-exact labeling $\widehat{\sigma}$ obtained from Phase I. Our refinement procedure mimics the so-called *genie-aided* estimator [1], which labels a vertex u knowing the labels of all other vertices (i.e., $\{\sigma_0(v) \colon v \in V \setminus \{u\}\}$). The degree profile relative to the ground-truth labeling, $d(u, \sigma_0)$, is random and depends on realizations of node locations and edges in G and community assignment σ_0. We use $D \in \mathbb{R}^4$ to denote the vector representing the four random variables in $d(u, \sigma_0)$. Then D is characterized by a multi-type Poisson distribution such that conditioned on $\{\sigma_0(u) = 1\}$, $D \sim \mathrm{Poisson}(\lambda\nu_d \log n[a, 1-a, b, 1-b]/2)$ and conditioned on $\{\sigma_0(u) = -1\}$, $D \sim \mathrm{Poisson}(\lambda\nu_d \log n[b, 1-b, a, 1-a]/2)$. Given a realization $D = d(u, \sigma_0)$, we pick the most likely hypothesis to minimize the error probability; that is,

$$\begin{aligned}\sigma_{\mathsf{genie}}(u) &= \operatorname*{argmax}_{s\in\{1,-1\}} \mathbb{P}(D = d(u,\sigma_0) \,|\, \sigma_0(u) = s) \\ &= \mathrm{sign}\Big[\log\big(\tfrac{a}{b}\big)\big(d_1^+(u,\sigma_0) - d_{-1}^+(u,\sigma_0)\big) + \log\big(\tfrac{1-a}{1-b}\big)\big(d_1^-(u,\sigma_0) - d_{-1}^-(u,\sigma_0)\big)\Big]. \end{aligned} \tag{3.1}$$

For convenience, let

$$\tau(u,\sigma) = \log\big(\tfrac{a}{b}\big)\big(d_1^+(u,\sigma) - d_{-1}^+(u,\sigma)\big) + \log\big(\tfrac{1-a}{1-b}\big)\big(d_1^-(u,\sigma) - d_{-1}^-(u,\sigma)\big). \tag{3.2}$$

In short, we have $\sigma_{\mathsf{genie}}(u) = \mathrm{sign}(\tau(u, \sigma_0))$. The genie-aided estimator motivates the `Refine` subroutine (Algorithm 4) in Phase II that assigns $\widetilde{\sigma}(u) = \mathrm{sign}(\tau(u, \widehat{\sigma}))$ for any $u \in V$. Since $\widehat{\sigma}$ makes few errors compared with σ_0, for any $u \in V$, its degree profile $d(u, \widehat{\sigma})$ is close to $d(u, \sigma_0)$. Thus, $d(u, \widehat{\sigma})$ is well-approximated by the aforementioned multi-type Poisson distribution.

Modified algorithm for general $\lambda > 1$. If $1 < \lambda < 2$, partitioning the interval into blocks of length $\log n/2$, as done in Line 2 of Algorithm 1, fails. This is because each of the $2n/\log n$ blocks is independently empty with probability $e^{-\lambda \log n/2} = n^{-\lambda/2}$ and $-\lambda/2 > -1$, leading to a high probability of encountering empty blocks, and thus a failure of the propagation scheme. To address this, we instead adopt smaller blocks of length $\chi \log n$, where $\chi < (1 - 1/\lambda)/2$, for any $\lambda > 1$. We only attempt to label blocks with sufficiently many vertices, according to the following definition. For the rest of the paper, let $V(B) \subset V$ denote the set of vertices in a subregion $B \subset \mathcal{S}_{d,n}$.

Definition 3.2. (Occupied block) *Given any $\delta > 0$, a block $B \subset \mathcal{S}_{d,n}$ is δ-occupied if $|V(B)| > \delta \log n$. Otherwise, B is δ-unoccupied.*

We will show that for sufficiently small $\delta > 0$, all but a negligible fraction of blocks are δ-occupied. As a result, achieving almost-exact recovery in Phase I only requires labeling the vertices within the occupied blocks. To ensure successful propagation, we introduce a notion of visibility. Two blocks $B_i, B_j \in \mathcal{S}_{d,n}$ are *mutually visible*, defined as $B_i \sim B_j$, if

$$\sup_{x\in B_i, y\in B_j} \|x - y\| \le (\log n)^{1/d}.$$

Thus, if $B_i \sim B_j$, then any pair of vertices $u \in B_i$ and $v \in B_j$ are at a distance at most $(\log n)^{1/d}$ of each other. In particular, if B_j is labeled and $B_i \sim B_j$, then we can propagate labels to B_i.

Similar to the case of $\lambda > 2$, we propagate labels from left to right. Despite the presence of unoccupied blocks, we establish that if $\lambda > 1$ and χ is chosen as above, each block B_i following the initial B_1 has a corresponding block B_j $(j < i)$ to its left that is occupied and satisfies $B_i \sim B_j$. We thus modify Lines 4-5 so that a given block B_i is labeled by one of the visible, occupied blocks to its left (Figure 1). The modification is formalized in the general algorithm (Algorithm 5) given below.

3.2 Exact recovery for general d. The propagation scheme becomes more intricate for $d \geq 2$. For general d, we divide the region $\mathcal{S}_{d,n}$ into hypercubes[3] with volume parametrized as $\chi \log n$. The underlying intuition for successful propagation stems from the condition $\lambda \nu_d > 1$. This condition ensures that the graph formed by connecting all pairs of mutually visible vertices is connected with high probability, a necessary condition for exact recovery. Moreover, the condition ensures that every vertex has $\Omega(\log n)$ vertices within its visibility radius of $(\log n)^{1/d}$. It turns out that the condition $\lambda \nu_d > 1$ also ensures that blocks of volume $\chi \log n$ for $\chi > 0$ sufficiently small satisfy the same connectivity properties.

To propagate the labels, we need a schedule to visit all occupied blocks. However, the existence of unoccupied blocks precludes the use of a predefined schedule, such as a lexicographic order scan. Instead, we employ a data-dependent schedule. The schedule is determined by the set of occupied blocks, which in turn is determined in Step 1 of Definition 2.1. Crucially, the schedule is thus independent of the community labels and edges, conditioned on the number of vertices in each block. We first introduce an auxiliary graph $H = (V^\dagger, E^\dagger)$, which records the connectivity relation among occupied blocks.

DEFINITION 3.3. (VISIBILITY GRAPH) *Consider a Poisson point process $V \subset \mathcal{S}_{d,n}$, the $(\chi \log n)$-block partition of $\mathcal{S}_{d,n}$, $\{B_i\}_{i=1}^{n/(\chi \log n)}$, corresponding vertex sets $\{V_i\}_{i=1}^{n/(\chi \log n)}$, and a constant $\delta > 0$. The (χ, δ)-visibility graph is denoted by $H = (V^\dagger, E^\dagger)$, where the vertex set $V^\dagger = \{i \in [n/(\chi \log n)] : |V_i| \geq \delta \log n\}$ consists of all δ-occupied blocks and the edge set is given by $E^\dagger = \{\{i, j\} : i, j \in V^\dagger, B_i \sim B_j\}$.*

We adopt the standard connectivity definition on the visibility graph. Lemma 4.2 shows that the visibility graph of the Poisson point process underlying the GSBM is connected with high probability. Based on this connectivity property, we establish a propagation schedule as follows. We construct a spanning tree of the visibility graph and designate a root block as the initial block. We specify an ordering of $V^\dagger = \{i_1, i_2, \dots\}$ according to a tree traversal (e.g., breadth-first search). Labels are propagated according to this ordering, thus labeling vertex sets $V_{i_1}, V_{i_2}, \cdots$ (see Figure 1). Letting $p(i)$ denote the parent of vertex $i \in V^\dagger$ according to the rooted tree, we label V_{i_j} using $V_{p(i_j)}$ as reference. Importantly, the visibility graph and thus the propagation schedule is determined only by the locations of vertices, independent of the labels and edges between mutually visible blocks.

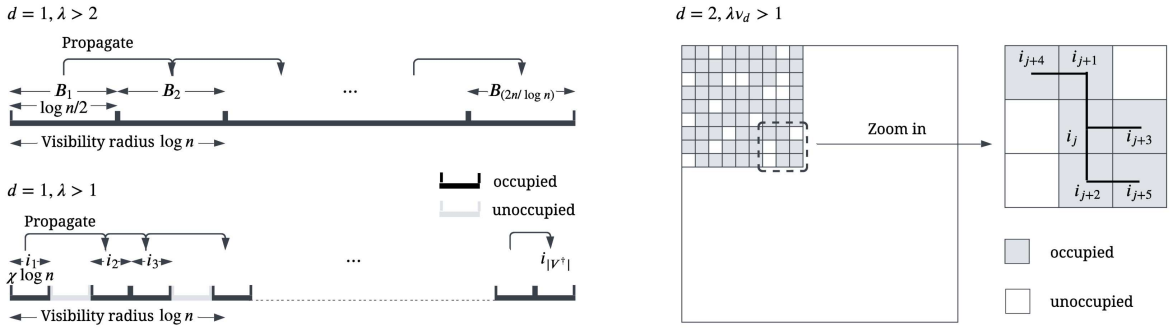

Figure 1: Propagation schedule for $d = 1$ and $d = 2$.

Algorithm 5 Exact recovery for the GSBM

Input: $G \sim \text{GSBM}(\lambda, n, a, b, d)$.
Output: An estimated community labeling $\widetilde{\sigma} : V \to \{-1, 1\}$.

1: **Phase I:**
2: Take small enough $\chi, \delta > 0$, satisfying the conditions to be specified in (4.1) and (4.2) respectively.
3: Partition the region $\mathcal{S}_{d,n}$ into $n/(\chi \log n)$ blocks of volume $\chi \log n$ each. Let B_i be the ith block and V_i be the set of vertices in B_i for $i \in [n/(\chi \log n)]$.
4: Form the associated visibility graph $H = (V^\dagger, E^\dagger)$.
5: **if** H is disconnected **then**

[3]For $d = 1, 2, 3$, the hypercubes represent line segments, squares and cubes respectively.

6: Return FAIL.
7: Find a rooted spanning tree of H, ordering $V^\dagger = \{i_1, i_2, \cdots\}$ in breadth-first order.
8: Apply `Pairwise Classify` (Algorithm 2) on input G, V_{i_1}, a, b to obtain a labeling $\widehat{\sigma}$ of V_{i_1}.
9: **for** $j = 2, \cdots, |V^\dagger|$ **do**
10: Apply `Propagate` (Algorithm 3) on input $G, V_{p(i_j)}, V_{i_j}$ to determine the labeling $\widehat{\sigma}$ on V_i.
11: **for** $u \in V \setminus (\cup_{i \in V^\dagger} V_i)$ **do**
12: Set $\widehat{\sigma}(u) = 0$.

13: **Phase II:**
14: **for** $u \in V$ **do**
15: Apply `Refine` (Algorithm 4) on input $G, \widehat{\sigma}, u$ to determine $\widetilde{\sigma}(u)$.

Algorithm 5 presents our algorithm for the general case. We partition the region $\mathcal{S}_{d,n}$ into blocks with volume $\chi \log n$, for a suitably chosen $\chi > 0$. A threshold level of occupancy $\delta > 0$ is specified. The value of χ is carefully chosen to ensure that the visibility graph H is connected with high probability in Line 5. In Line 8, we label an initial δ-occupied block, corresponding to the root of H, using the `Pairwise Classify` subroutine. In Lines 9-10, we label the occupied blocks in the tree order determined in Line 7, using the `Propagate` subroutine. Those vertices appearing in unoccupied blocks are assigned a label of 0. At the end of Phase I, we obtain a first-stage labeling $\widehat{\sigma}\colon V \to \{-1, 0, 1\}$, such that with high probability, all occupied blocks are labeled with few mistakes. Finally, Phase II refines the almost-exact labeling $\widehat{\sigma}$ to an exact one $\widetilde{\sigma}$ according to Algorithm 4.

To analyze the runtime, note that the number of edges (input size) is $\Theta(n \log n)$ with high probability. The visibility graph $H = (V^\dagger, E^\dagger)$ can be formed in $O(n/\log n)$ time, since $|V^\dagger| = O(n/\log n)$ and each vertex has at most $\Theta(1)$ possible neighbors. If H is connected, a spanning tree can be found in $O(|E^\dagger| \log(|E^\dagger|))$ time using Kruskal's algorithm, and $|E^\dagger| = O(n/\log n)$. The subsequent `Pairwise Classify` subroutine goes over all edges of the vertices in V_1 to count the common neighbors, with a runtime of $O(\log^2 n)$. Next, the `Propagation` subroutine requires counting edges and non-edges from any given vertex in an occupied block to the vertices in its reference block, yielding a runtime of $O(n \log n)$. Finally, `Refine` runs in $O(n \log n)$ time, since each visible neighborhood contains $O(\log n)$ vertices. We conclude that Algorithm 5 runs in $O(n \log n)$ time, which is linear in the number of edges.

3.3 Proof outline. We outline the analysis of Algorithm 5. We begin with Phase I. Our goal is to show that in addition to achieving almost exact recovery stated in Theorem 2.4, Phase I also satisfies an error dispersion property. Let $\mathcal{N}(u) = \{v \in V, \|u - v\| \le (\log n)^{1/d}\}$ for a vertex u. Namely, for any $\eta > 0$, we can take suitable $\chi, \delta > 0$ so that with high probability, every vertex has at most $\eta \log n$ incorrectly classified vertices in its local neighborhood $\mathcal{N}(u)$. Theorem 4.1 will present the formal results.

Phase I: Connectivity of the visibility graph. We first establish that the block division specified in Algorithm 5 ensures that the resulting visibility graph $H = (V^\dagger, E^\dagger)$ is connected. Elementary analysis shows that any fixed subregion of $\mathbb{R}^d$ with volume $\nu \log n$ contains $\Omega(\log n)$ vertices with probability $1 - o(n^{-1})$, whenever $\nu > 1/\lambda$. A union bound over all vertices then implies that all vertices' neighborhoods have $\Omega(\log n)$ vertices. In the special case of $d = 1$, the left neighborhood of a given vertex has volume $\log n$. The observation with $\nu = 1$ implies that when $\lambda > 1$, the left neighborhood of every vertex has $\Omega(\log n)$ points. In fact, we can make a stronger claim: if the block lengths are chosen to be sufficiently small (according to (4.1)), then we can ensure that for a given vertex $v \in V_i$, there are $\Omega(\log n)$ vertices among $\{V_j : B_j \sim B_i, j \neq i\}$. In turn, by an appropriate choice of δ (according to (4.2)), for a given block B_i, there is at least one δ-occupied, visible block to its left. Hence, the visibility graph is connected, as shown in Proposition 4.1.

However, the analysis becomes more intricate when $d \ge 2$. In particular, while a lexicographic order propagation schedule succeeds for $d = 1$, it fails for $d \ge 2$. For example, when $d = 2$, we cannot say that every vertex has $\Omega(\log n)$ vertices in the top left quadrant of its neighborhood, since the volume of the quadrant is only $\nu_d \log n/4$. We therefore establish connectivity of H using the fact that if H is disconnected, then H must contain an isolated connected component. The key idea is that if there is an isolated connected component in H, then the corresponding occupied blocks in $\mathbb{R}^d$ must be surrounded by sufficiently many unoccupied blocks. However, as Lemma 4.6 shows, there cannot be too many adjacent unoccupied blocks, which prevents the existence of isolated connected components. As a result, the visibility graph is connected, as shown in Lemma 4.2.

Phase I: Labeling the initial block. We show that the `Pairwise Classify` (Line 8) subroutine ensures the successful labeling for V_{i_1}. Since we only need to determine community labels up to a global flip, we are free to set $\widehat{\sigma}(u_0) = 1$ for an arbitrary $u_0 \in V_{i_1}$. For any $u \in V_{i_1} \setminus \{u_0\}$, where $|V_{i_1}| = m_1$, Lemma 4.7 shows that the number of common neighbors of u and u_0 follows a binomial distribution; in particular, $\mathrm{Bin}(m_1 - 2, (a^2 + b^2)/2)$ if $\sigma_0(u) = \sigma_0(u_0)$ and $\mathrm{Bin}(m_1 - 2, ab)$ otherwise. We thus threshold the number of common neighbors in order to classify u relative to u_0. Lemma 4.8 bounds the probability of misclassifying a given vertex $u \in V_{i_1} \setminus \{u_0\}$, using Hoeffding's inequality. A union bound then implies that all vertices are correctly classified with high probability.

Phase I: Propagating labels among occupied blocks. We show that the `Propagate` subroutine ensures that $\widehat{\sigma}$ makes at most M mistakes in each occupied block, where M is a suitable constant. Our analysis reduces to bounding the probability that for a given $i \in V^{\dagger}$, the estimator $\widehat{\sigma}$ makes more than M mistakes on V_i, conditioned on making no more than M mistakes on $V_{p(i)}$. In order to analyze the probability that a given vertex $v \in V_i$ is misclassified, we condition on the *label configuration* of $V_{p(i)}$, meaning the number of vertices labeled s according to $\sigma_0(\cdot)$ and t according to $\sigma_0(u_0)\widehat{\sigma}(\cdot)$, for $s, t \in \{-1, +1\}$. We find a uniform upper bound on the probability of misclassifying an individual vertex $v \in V_i$ when applying the thresholding test given in Algorithm 3, over all label configurations of $V_{p(i)}$ with at most M mistakes. To bound the total number of mistakes in V_i, observe that the labels of all vertices in V_i are decided based on disjoint subsets of edges between V_i and $V_{p(i)}$. Therefore, conditioned on the label configuration of $V_{p(i)}$, the number of mistakes in V_i can be stochastically dominated by a binomial random variable. It follows by elementary analysis that the number of mistakes in V_i is at most M with probability $1 - o(n^{-1})$, as long as M is a suitably large constant.

Phase II: Refining the labels. Our final step is to refine the initial labeling $\widehat{\sigma}$ from Phase I into a final labeling $\widetilde{\sigma}$. Unfortunately, conditioning on a successful labeling $\widehat{\sigma}$ destroys the independence of edges, making it difficult to bound the error probability of $\widetilde{\sigma}$. This issue can be remedied using a technique called *graph splitting*, used in the two-round procedure of [5]. Graph splitting is a procedure to form two graphs, G_1 and G_2, from the original input graph. A given edge in G is independently assigned to G_1 with probability p, and G_2 with probability $1 - p$, for p chosen so that almost exact recovery can be achieved on G_1, while exact recovery can be achieved on G_2. Since the two graphs are nearly independent, conditioning on the success of almost exact recovery in G_1 essentially maintains the independence of edges in G_2.

While we believe that our Phase I algorithm, along with graph splitting, would achieve the information-theoretic threshold in the GSBM, we instead directly analyze the robustness of Poisson testing. Specifically, we bound the error probability of labeling a given vertex $v \in V$ with respect to the worst-case labeling over all labelings that differ from σ_0 on at most $\eta \log n$ vertices in the neighborhood of v. Since $\widehat{\sigma}$ makes at most $\eta \log n$ errors with probability $1 - o(1/n)$ (Theorem 4.1), we immediately obtain a bound on the error probability of $\widetilde{\sigma}(v)$.

The proof in Section 5 bounds the worst-case error probability. We define $x = \lambda\nu_d[a, 1-a, b, 1-b]/2$ and $y = \lambda\nu_d[b, 1-b, a, 1-a]/2$, so that $D \,|\, \{\sigma_0(u) = 1\} \sim \mathrm{Poisson}(x)$ and $D \,|\, \{\sigma_0(u) = -1\} \sim \mathrm{Poisson}(y)$. The condition $\lambda\nu_d(1 - \sqrt{ab} - \sqrt{(1-a)(1-b)}) > 1$ in Theorem 2.2 is equivalent to $D_+(x\|y) > 1$, where $D_+(x\|y)$ is the Chernoff–Hellinger divergence of x and y [5]. To provide intuition for bounding the error probability at a given vertex $u \in V$, consider the genie-aided estimator $\sigma_{\mathsf{genie}}(u)$, and assume $\sigma_0(u) = 1$ without loss of generality. Recalling the definition of τ (3.2), the estimator $\sigma_{\mathsf{genie}}(u)$ makes a mistake when $\tau(u, \sigma_0) \le 0$. It can be shown that this occurs with probability at most $n^{-D_+(x\|y)}$. Viewing the worst-case labeling σ differing from σ_0 on at most $\eta \log n$ vertices as a perturbation of σ_0, we show that $\tau(u, \sigma) \le 0$ implies $\tau(u, \sigma_0) \le \rho\eta \log n$ for a certain constant ρ. Similarly, the probability of such a mistake is at most $n^{-D_+(x\|y)+\rho\eta/2}$. Thus, for small $\eta > 0$, the condition $D_+(x\|y) > 1$ and a union bound over all vertices yields an error probability of $o(1)$.

4 Phase I: Proof of almost exact recovery

In this section, we prove Theorem 2.4. We begin by defining sufficiently small constants χ and δ used in Algorithm 5. We define χ to satisfy the following condition, relying on λ and d:

$$\nu_d\big(1 - 3\sqrt{d}\chi^{1/d}/2\big)^d \ge (\nu_d + 1/\lambda)/2 \text{ and } 0 < \chi < [(\mathbb{1}_{d=1} + \nu_d \cdot \mathbb{1}_{d\ge 2}) - 1/\lambda]/2. \tag{4.1}$$

The first condition is satisfiable since $\lim_{\chi\to 0} \nu_d\big(1 - 3\sqrt{d}\chi^{1/d}/2\big)^d = \nu_d$ and we have $\nu_d > (\nu_d + 1/\lambda)/2$ when $\lambda\nu_d > 1$. The second one is also satisfiable since $\mathbb{1}_{d=1} + \nu_d \cdot \mathbb{1}_{d\ge 2} = 1 > 1/\lambda$ if $d = 1$ and otherwise $\mathbb{1}_{d=1} + \nu_d \cdot \mathbb{1}_{d\ge 2} = \nu_d > 1/\lambda$, under the conditions of Theorems 2.2 and 2.4. Associated with the choice of

χ, there is a constant δ'(or $\widetilde{\delta}$ for $d \geq 2$) > 0 such that for any block B_i, its visible blocks $\bigcup_{j\in V}\{V_j : B_j \sim B_i\}$ contain at least $\delta' \log n$ (or $\widetilde{\delta}\log n$) vertices with probability $1 - o(n^{-1})$. We define $R_d = 1 - \sqrt{d}\chi^{1/d}/2$. The first condition in (4.1) implies that $\sqrt{d}\chi^{1/d}/2 < 1/3$ and thus $R_d > 0$. With specific values of δ' and $\widetilde{\delta}$ to be determined in Proposition 4.1 and Lemma 4.5, respectively, we define δ such that

$$0 < \delta < (\delta'\chi) \cdot \mathbb{1}_{d=1} + [\widetilde{\delta}\chi/(\nu_d R^d)] \cdot \mathbb{1}_{d\geq 2}. \tag{4.2}$$

Propositions 4.1 and 4.2 will present the connectivity properties of δ-occupied blocks of volume $\chi \log n$, for χ and δ satisfying the conditions in (4.1) and (4.2), respectively.

We now record some preliminaries (see [9]).

LEMMA 4.1. (CHERNOFF BOUND, POISSON) *Let $X \sim Poisson(\mu)$ with $\mu > 0$. For any $t > 0$,*

$$\mathbb{P}(X \geq \mu + t) \leq \exp\Big(-\frac{t^2}{2(\mu + t)}\Big).$$

For any $0 < t < \mu$, we have

$$\mathbb{P}(X \leq \mu - t) \leq \exp\Big(-(\mu - t)\log\Big(1 - \frac{t}{\mu}\Big) - t\Big).$$

LEMMA 4.2. (HOEFFDING'S INEQUALITY) *Let $X_1, \cdots, X_n$ be independent bounded random variables with values $X_i \in [0, 1]$ for all $1 \leq i \leq n$. Let $X = \sum_{i=1}^n X_i$ and $\mu = \mathbb{E}[X]$. Then for any $t \geq 0$, it holds that*

$$\mathbb{P}(X \geq \mu + t) \leq \exp(-2t^2/n), \quad \mathbb{P}(X \leq \mu - t) \leq \exp(-2t^2/n).$$

LEMMA 4.3. (CHERNOFF UPPER BOUND) *Let $X_1, \cdots, X_n$ be independent Bernoulli random variables. Let $X = \sum_{i=1}^n X_i$ and $\mu = \mathbb{E}(X)$. Then for any $t > 0$, we have*

$$\mathbb{P}(X \geq (1 + t)\mu) \leq \Big(\frac{e^t}{(1 + t)^{(1+t)}}\Big)^{\mu}.$$

We also define a homogeneous Poisson point process used to generate locations as described in Definition 2.1.

DEFINITION 4.1. ([23]) *A homogeneous Poisson point process with intensity λ on $S \subseteq \mathbb{R}^d$ is a random countable set $\Phi := \{v_1, v_2, \cdots\} \subset S$ such that*

1. *For any bounded Borel set $B \subset \mathbb{R}^d$, the count $N_\Phi(B) := |\Phi \cap B| = |\{i \in \mathbb{N} : v_i \in B\}|$ has a Poisson distribution with mean $\lambda vol(B)$, where $vol(B)$ is the measure (volume) of B.*
2. *For any $k \in \mathbb{N}$ and any disjoint Borel sets $B_1, \cdots, B_k \subset \mathbb{R}^d$, the random variables $N_\Phi(B_1), \cdots, N_\Phi(B_k)$ are mutually independent.*

In the GSBM, the set of locations $V = \{v_1, v_2, \cdots\}$ are generated by a homogeneous Poisson point process with intensity λ on $\mathcal{S}_{n,d}$. The established properties guarantee that $|V|$ follows Poisson(λn). Moreover, conditioned on $|V|$, the locations $\{v_i\}_{i\in[|V|]}$ are independently and uniformly distributed in $\mathcal{S}_{n,d}$. This gives a simple construction of a Poisson point process as follows:

1. Sample $N_V \sim$ Poisson(λn);
2. Sample $v_1, \cdots, v_{N_V}$ independently and uniformly in the region $\mathcal{S}_{n,d}$.

This procedure ensures that the resulting set $\{v_1, \cdots, v_{N_V}\}$ constitutes a Poisson point process as desired.

4.1 Connectivity of the visibility graph. In this subsection, we establish the connectivity of the visibility graph $H = (V^\dagger, E^\dagger)$ from Line 4 of Algorithm 5. The following lemma shows that regions of appropriate volume have $\Omega(\log n)$ vertices with high probability.

LEMMA 4.4. *For any fixed subset $B \subset \mathcal{S}_{d,n}$ with a volume $\nu \log n$ such that $\lambda\nu > 1$, there exist constants $0 < \gamma < \lambda\nu$ and $\epsilon > 0$ such that*

$$\mathbb{P}(|V(B)| > \gamma \log n) \geq 1 - n^{-1-\epsilon}.$$

Proof. For a subset B with $\mathrm{vol}(B) = \nu \log n$, we have $|V(B)| \sim \mathrm{Poisson}(\lambda\nu \log n)$. To show the lower bound, we define a function $g : (0, \lambda\nu] \to \mathbb{R}$ as $g(x) = x(\log x - \log(\lambda\nu)) + \lambda\nu - x$. It is easy to check that g is continuous and decreases on $(0, \lambda\nu]$ with $\lim_{x\to 0} g(x) = \lambda\nu$ and $g(\lambda\nu) = 0$. When $\lambda\nu > 1$, it holds that $\lim_{x\to 0} g(x) = \lambda\nu > (1 + \lambda\nu)/2$ and thus there exists a constant $\gamma \in (0, \lambda\nu)$ such that $g(\gamma) > (1 + \lambda\nu)/2$. Thus, the Chernoff bound in Lemma 4.1 yields that

$$\mathbb{P}(|V(B)| \leq \gamma \log n) \leq \exp\Big(- [\gamma(\log \gamma - \log(\lambda\nu)) + \lambda\nu - \gamma] \log n\Big) = n^{-g(\gamma)} \leq n^{-(1+\lambda\nu)/2}.$$

Taking $\epsilon = (\lambda\nu - 1)/2 > 0$ concludes the proof. ∎

4.1.1 The simple case when $d = 1$ and $\lambda > 1$.

We start with the simple case when $d = 1$.

An example when $\lambda > 2$. We first study an example when $d = 1$ and $\lambda > 2$. If $\lambda > 2$ and $\mathrm{vol}(B_i) = \log n/2$, we have $\lambda \mathrm{vol}(B_i)/\log n > 1$, and thus Lemma 4.4 ensures the existence of positive constants γ and ϵ such that $\mathbb{P}(|V_i| > \gamma \log n) \geq 1 - n^{-1-\epsilon}$ for all $i \in [2n/\log n]$. Thus, the union bound gives that

$$\mathbb{P}\Big(\bigcap_{i=1}^{2n/\log n} \big\{|V_i| > \gamma \log n\big\}\Big) = 1 - \mathbb{P}\Big(\bigcup_{i=1}^{2n/\log n} \big\{|V_i| \leq \gamma \log n\big\}\Big) \geq 1 - \frac{2n}{\log n} \cdot n^{-1-\epsilon} = 1 - o(1). \tag{4.3}$$

Since all blocks are γ-occupied, the $(1/2, \gamma)$-visibility graph $H = (V^\dagger, E^\dagger)$ is trivially connected.

General case when $\lambda > 1$. For small density λ, we partition the interval into small blocks and establish the existence of visible occupied blocks on the left side of each block.

PROPOSITION 4.1. *If $d = 1$ and $\lambda > 1$, with $0 < \chi < (1 - 1/\lambda)/2$, we consider the blocks $\{B_i\}_{i=1}^{n/(\chi \log n)}$ obtained from Line 3 in Algorithm 5. Then there exists a constant $\delta' > 0$ such that for any $0 < \delta < \delta'\chi$, it holds that*

$$\mathbb{P}\Big(\bigcap_{i=1}^{n/(\chi \log n)} \big\{\exists j \colon j < i, B_j \sim B_i, \text{ and } B_j \text{ is } \delta\text{-occupied}\big\}\Big) = 1 - o(1).$$

It follows that the (χ, δ)-visibility graph is connected with high probability.

Proof. For any $i \in [n/(\chi \log n)]$, we define $U_i = \bigcup_{j \colon j<i, B_j \sim B_i} B_j$ as the union of visible blocks on the left-hand side of B_i. We have $\mathrm{vol}(U_i) = (\lfloor 1/\chi \rfloor - 1)\chi \log n \geq (1 - 2\chi) \log n$ and $\lambda \mathrm{vol}(U_i)/\log n \geq \lambda(1 - 2\chi) > 1$ when $\lambda > 1$ and $\chi < (1 - 1/\lambda)/2$. Thus, Lemma 4.4 ensures the existence of positive constants δ' and ϵ such that $\mathbb{P}(|\bigcup_{j \colon j<i, B_i \sim B_j} V_j| \leq \delta' \log n) \leq n^{-1-\epsilon}$. We note that $|\{j \colon j < i, B_j \sim B_i\}| \leq (\lceil 1/\chi \rceil - 1) \leq 1/\chi$. Thus, we take $0 < \delta < \delta'\chi$ and obtain that

$$\begin{aligned}\mathbb{P}\Big(\bigcap_{j \colon j<i, B_j \sim B_i} \big\{|V_j| \leq \delta \log n\big\}\Big) &\leq \mathbb{P}\Big(\Big| \bigcup_{j \colon j<i, B_j \sim B_i} V_j\Big| \leq \delta \log n/\chi\Big) \\ &\leq \mathbb{P}\Big(\Big| \bigcup_{j \colon j<i, B_j \sim B_i} V_j\Big| \leq \delta' \log n\Big) \leq n^{-1-\epsilon}.\end{aligned}$$

Therefore, the union bound over all $i \in [n/(\chi \log n)]$ gives

$$\begin{aligned}&\mathbb{P}\Big(\bigcap_{i=1}^{n/(\chi \log n)} \big\{\exists j \colon j < i, B_j \sim B_i, \text{ and } B_j \text{ is } \delta\text{-occupied}\big\}\Big) \\ &\quad = 1 - \mathbb{P}\Big(\bigcup_{i=1}^{n/(\chi \log n)} \bigcap_{j \colon j<i, B_j \sim B_i} \big\{|V_j| \leq \delta \log n\big\}\Big)\end{aligned}$$

$$\geq 1 - \frac{n}{\chi \log n} \cdot n^{-1-\epsilon} = 1 - o(1).$$

□

4.1.2 General case when $d \geq 2$ and $\lambda\nu_d > 1$. We now study general cases. We first show that for any block B, the set of surrounding visible blocks $\{B' \colon B \sim B', B' \neq B\}$ contains $\Omega(\log n)$ vertices. For any block $B_i \subset \mathcal{S}_{d,n}$ with $\mathrm{vol}(B_i) = \chi \log n$, the length of its longest diagonal is given by $\sqrt{d}(\chi \log n)^{1/d}$. Recall the definition of $R_d = 1 - \sqrt{d}\chi^{1/d}/2$, and let C_i be the ball of radius $R_d(\log n)^{1/d}$ centered at the center of B_i. Observe that

$$\sup_{x \in B_i, y \in C_i} \|x - y\| = \frac{1}{2}\sqrt{d}(\chi \log n)^{1/d} + R_d(\log n)^{1/d} = (\log n)^{1/d}.$$

It follows that if $B_j \subseteq C_i$, then $B_i \sim B_j$. Also, C_i contains all blocks $B_j \sim B_i$ (see Figure 2). We define

$$U_i = \bigcup_{j \colon j \neq i, B_j \sim B_i} B_j = \bigcup_{j \neq i \colon B_j \subset C_i} B_j$$

as the union of all visible blocks to B_i, excluding B_i itself. Observe that as $\chi \to 0$, the volume of the blue region approaches the volume of C_i. The following lemma quantifies this observation, showing that our conditions on χ guarantee that U_i (and any set with the same volume as U_i) will contain sufficiently many vertices.

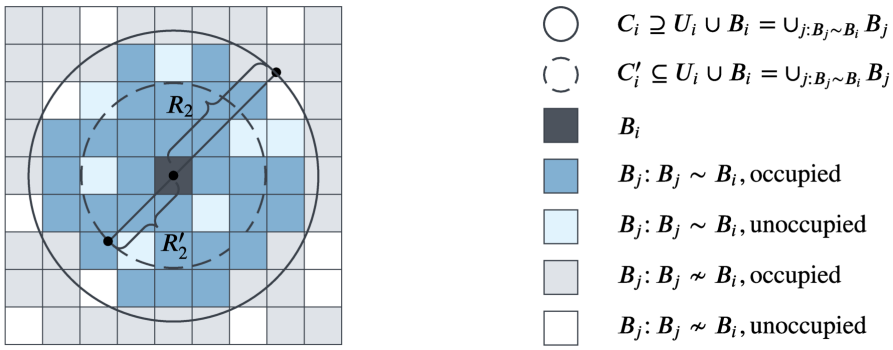

Figure 2: Geometry around block B_i, showing a portion of the region $\mathcal{S}_{2,n}$. The set U_i is comprised of dark and light blue blocks.

LEMMA 4.5. *If χ satisfies the condition in (4.1) and $\lambda\nu_d > 1$, there exist positive constants $\widetilde{\delta}$ and ϵ, depending on λ and d, such that for any subset $S \in \mathcal{S}_{d,n}$ with $vol(S) = vol(U_i)$, we have*

$$\mathbb{P}\big(|V(S)| > \widetilde{\delta} \log n\big) \geq 1 - n^{-1-\epsilon}.$$

Proof. We first evaluate the volume of $U_i \subset C_i$.[4] We define $R_d' = R_d - \sqrt{d}\chi^{1/d}$ and C_i' as the ball centered at the center of B_i with a radius $R_d'(\log n)^{1/d}$. The condition in (4.1) implies that $3\sqrt{d}\chi^{1/d}/2 < 1$ and thus $R_d' > 0$. Based on geometric observations, we note that $C_i' \subset U_i \cup B_i \subset C_i$. It follows that $\mathrm{vol}(U_i \cup B_i) \geq \mathrm{vol}(C_i') = \nu_d(R_d')^d \log n$, and thus $\mathrm{vol}(U_i) \geq (\nu_d(R_d')^d - \chi)\log n$.

We now show that when $\lambda\nu_d > 1$, the conditions in (4.1) imply $\lambda(\nu_d(R_d')^d - \chi) > 1$ by observing the following relations:

$$\begin{aligned}\nu_d(R_d')^d - \chi &= \nu_d\big(1 - 3\sqrt{d}\chi^{1/d}/2\big)^d - \chi \\ &\geq (\nu_d + 1/\lambda)/2 - \chi \\ &\geq 1/\lambda.\end{aligned}$$

[4] This is similar to the Gauss circle problem [21].

In summary, we have shown that $\mathrm{vol}(S) = \mathrm{vol}(U_i) \geq (\nu_d(R_d')^d - \chi)\log n$ and $\lambda(\nu_d(R_d')^d - \chi) > 1$. Thus, Lemma 4.4 ensures the existence of positive constants $\widetilde{\delta}$ and ϵ such that $\mathbb{P}(|V(S)| > \widetilde{\delta}\log n) > 1 - n^{-1-\epsilon}$. ∎

Henceforth, we use the term "occupied block" to refer to δ-occupied blocks, as well as "unoccupied block", with the constant threshold $\delta = \delta(\lambda, d)$ defined in (4.2) in the rest of the section. We define $K = |\{j\colon B_j \subset U_i\}|$ as the number of blocks in U_i, a constant relying on λ and d. We note that $K \leq \nu_d(R_d)^d/\chi - 1 < \widetilde{\delta}/\delta$ since $U_i \cup B_i \subset C_i$. The key observation in establishing connectivity is that there cannot be a large *cluster* of unoccupied blocks.

Definition 4.2. (Cluster of blocks) *Two blocks are adjacent if they share an edge or a corner. We say that a set of blocks $\mathcal{B}$ is a cluster if for every $B, B' \in \mathcal{B}$, there is a path of blocks of the form $(B = B_{j_1}, B_{j_2}, \ldots, B_{j_m} = B')$, where $B_{j_k} \in \mathcal{B}$ for $k \in [m]$ and $B_{j_k}, B_{j_{k+1}}$ are adjacent.*

The following lemma shows that all clusters of unoccupied blocks have fewer than K blocks, with high probability. This also implies that U_i contains at least one occupied block for each i.

Lemma 4.6. *Suppose $d \geq 2$ and $\lambda\nu_d > 1$. Let Y be the size of the largest cluster of unoccupied blocks produced in Line 3 in Algorithm 5. Then $\mathbb{P}(Y < K) = 1 - o(1)$.*

Proof. We first bound the probability that all K blocks in any given set are unoccupied. For any set of K blocks $\{B_{j_k}\}_{k=1}^K$, we have

$$\begin{aligned}\mathbb{P}\Big(\bigcap_{k=1}^K \{|V_{j_k}| \leq \delta\log n\}\Big) &\leq \mathbb{P}\Big(\Big|\bigcup_{k=1}^K V_{j_k}\Big| \leq \delta K\log n\Big)\\ &\leq \mathbb{P}\Big(\Big|\bigcup_{k=1}^K V_{j_k}\Big| < \widetilde{\delta}\log n\Big)\\ &\leq n^{-1-\epsilon}, \qquad (4.4)\end{aligned}$$

where the second inequality holds due to $K < \widetilde{\delta}/\delta$ and the last inequality follows from Lemma 4.5 and the fact that $\mathrm{vol}(\bigcup_{k=1}^K B_{j_k}) = \mathrm{vol}(U_i)$.

Let Z be the number of unoccupied block clusters with a size of K. Then we have $\mathbb{P}(Y \geq K) = \mathbb{P}(Z \geq 1)$. Let $\mathfrak{S}$ be the set of all possible shapes of clusters of blocks with a size of K. Clearly, $|\mathfrak{S}|$ is a constant depending on K and d. For any $s \in \mathfrak{S}$, $i \in [n/(\chi\log n)]$, and $j \in [K]$, we define $\mathcal{Z}_{s,i,j}$ as the event that there is a cluster of unoccupied blocks, characterized by shape s with block B_i occupying the jth position. Due to (4.4), we have $\mathbb{P}(\mathcal{Z}_{s,i,j}) \leq n^{-1-\epsilon}$. Thus, the union bound gives

$$\begin{aligned}\mathbb{P}(Y \geq K) = \mathbb{P}(Z \geq 1) &= \mathbb{P}\Big(\bigcup_{s\in\mathfrak{S}, i\in[n/(\chi\log n)], j\in[K]} \mathcal{Z}_{s,i,j}\Big)\\ &\leq |\mathfrak{S}| \cdot \frac{n}{\chi\log n} \cdot K \cdot n^{-1-\epsilon} = o(1).\end{aligned}$$

∎

Finally, we establish the connectivity of the visibility graph.

Proposition 4.2. *Suppose that $d \geq 2$ and $\lambda\nu_d > 1$. Let $V \subset \mathcal{S}_{d,n}$ be a Poisson point process on $\mathcal{S}_{d,n}$ with intensity λ. Then for χ and δ given in (4.1) and (4.2), respectively, the (χ,δ)-visibility graph H on V is connected with probability $1 - o(1)$.*

Proof. For a visibility graph $H = (V^\dagger, E^\dagger)$, we say that $S \subset V^\dagger$ is a *connected component* if the subgraph of H induced on S is connected. Let $\mathcal{E}$ be the event that H contains an isolated connected component. Formally, $\mathcal{E}$ is the event that there exists $S \subset V^\dagger$ [5] such that (1) $S \neq \emptyset$ and $S \neq V^\dagger$; (2) S is a connected component; (3) for all $i \in S, j \notin S$ we have $\{i,j\} \notin E^\dagger$. Observe that $\{H \text{ is disconnected}\} = \mathcal{E}$.

[5] The notation $\subset$ denotes a strict subset.

For any $S \neq \emptyset$ and $S \subset V^\dagger$ to be an isolated connected component, it must be completely surrounded by a cluster of unoccupied blocks. In other words, all blocks in the cluster $(\bigcup_{i\in S} U_i) \setminus (\bigcup_{i\in S} B_i)$ must be unoccupied. We next show that for any isolated, connected component S, we have $|\{j\colon B_j \subset (\bigcup_{i\in S} U_i) \setminus (\bigcup_{i\in S} B_i)\}| \geq K$; that is, the number of unoccupied blocks visible to an isolated connected component is at least K.

We prove the claim by induction on $|S|$. In fact, we prove it for S that is isolated, but not necessarily connected. The claim holds true whenever $|S| = 1$ by the definition of K. Suppose that the claim holds for every isolated component with k blocks. Consider an isolated component S, with $|S| = k+1$. Let $F = (\bigcup_{i\in S} B_i) \bigcup (\bigcup_{i\in S} U_i)$ be the collective "footprint" of all elements of S along with the surrounding unoccupied blocks. For each $j \in S$, let $F_j = (\bigcup_{i\in S, i\neq j} B_i) \bigcup (\bigcup_{i\in S, i\neq j} U_i)$ be the footprint of all blocks in S excluding j. Let G_j be the graph formed from G by removing all vertices from V_j, thus rendering V_j unoccupied. Observe that there must exist some $j^\star \in S$ such that $F_{j^\star} \neq F$ and $F_{j^\star} \subset F$, as the regions $\{B_i \cup U_i\}_{i\in S}$ are translations of each other. Since $S \setminus \{j^\star\}$ is an isolated component in $G_{j^\star}$, the inductive hypothesis implies that $S \setminus \{j^\star\}$ has at least K surrounding unoccupied blocks in $G_{j^\star}$. Comparing $G_{j^\star}$ to G, there are two cases (see Figure 3 for examples in $\mathcal{S}_{2,n}$). *Case I.* In the first case, $F \setminus F_{j^\star}$ contains at least one unoccupied block. In that case, the inclusion of $V_{j^\star}$ changes one block from unoccupied to occupied, and increases the number of surrounding unoccupied blocks by at least one. Thus, S contains at least K surrounding unoccupied blocks. *Case II.* In the second case, $F \setminus F_{j^\star}$ contains only occupied blocks. Since there are $k+1$ total occupied blocks in F and k of them are in $F_{j^\star}$, we have $F \setminus F_{j^\star} = B_{j^\star}$, so that $B_{j^\star} \cap F_{j^\star} = \emptyset$. In this case, the set of K surrounding unoccupied blocks in $F_{j^\star}$ remains unoccupied in F.

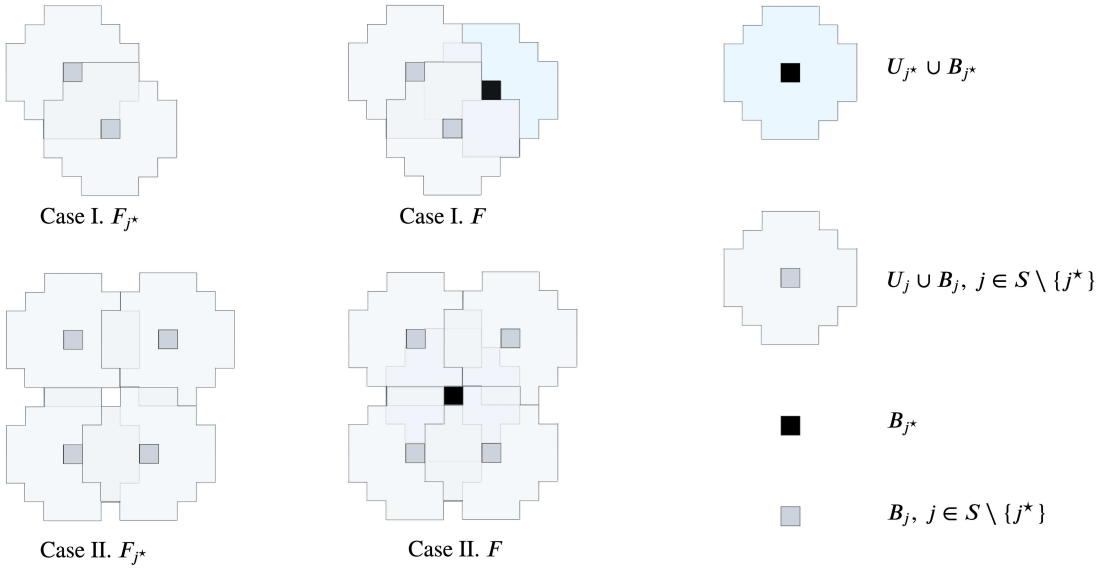

Figure 3: Possible isolated components in $\mathcal{S}_{2,n}$ for Proposition 4.2.

Thus, $\mathcal{E}$ implies $\{Y \geq K\}$. The result follows from Lemma 4.6. ∎

In summary, Propositions 4.1 and 4.2 establish the connectivity of visibility graphs for cases when $d = 1$ and $\lambda > 1$, or $d \geq 2$ and $\lambda\nu_d > 1$, ensuring successful label propagation in the algorithm. For convenience, let $\mathcal{H} = \{H \text{ is connected}\}$. We conclude that $\mathbb{P}(\mathcal{H}) = 1 - o(1)$.

4.2 Labeling the initial block. We now prove that the `Pairwise Classify` subroutine (Line 8 of Algorithm 5) ensures, with high probability, the correct labeling of all vertices in the initial block V_{i_1}. Let $N_{u_0,u} = |\{v \in V_{i_1}\colon \{v, u_0\} \in E, \{v, u\} \in E\}|$ be the number of common neighbors of u_0 and u within V_{i_1}.

LEMMA 4.7. *For any vertex $u \in V_{i_1} \setminus \{u_0\}$, it holds that*

1. *Conditioned on $\sigma_0(u) = \sigma_0(u_0)$ and $|V_{i_1}| = m_{i_1}$, we have $N_{u_0,u} \sim \mathrm{Bin}(m_{i_1} - 2, (a^2 + b^2)/2)$.*
2. *Conditioned on $\sigma_0(u) \neq \sigma_0(u_0)$ and $|V_{i_1}| = m_{i_1}$, we have $N_{u_0,u} \sim \mathrm{Bin}(m_{i_1} - 2, ab)$.*

Proof. We first consider the case when $\sigma_0(u) = \sigma_0(u_0)$. For any vertex $v \in V_{i_1} \setminus \{u, u_0\}$, we have

$$\begin{aligned}
&\mathbb{P}\big((v,u) \in E, (v,u_0) \in E \,|\, \sigma_0(u) = \sigma_0(u_0)\big) \\
&\quad = \mathbb{P}\big((v,u) \in E, (v,u_0) \in E \,|\, \sigma_0(v) = \sigma_0(u), \sigma_0(u) = \sigma_0(u_0)\big)\mathbb{P}\big(\sigma_0(v) = \sigma_0(u)\big) \\
&\qquad + \mathbb{P}\big((v,u) \in E, (v,u_0) \in E \,|\, \sigma_0(v) \neq \sigma_0(u), \sigma_0(u) = \sigma_0(u_0)\big)\mathbb{P}\big(\sigma_0(v) \neq \sigma_0(u)\big) \\
&\quad = (a^2 + b^2)/2.
\end{aligned}$$

The first statement follows from mutual independence of the events $\{(v,u), (v,u_0) \in E\}$ over $v \in V_{i_1} \setminus \{u, u_0\}$, conditioned on $|V_{i_1}| = m_{i_1}$.

Similarly, if $\sigma_0(u) \neq \sigma_0(u_0)$, for any $v \in V_{i_1} \setminus \{u, u_0\}$, we have

$$\begin{aligned}
&\mathbb{P}\big((v,u) \in E, (v,u_0) \in E \,|\, \sigma_0(u) \neq \sigma_0(u_0)\big) \\
&\quad = \mathbb{P}\big((v,u) \in E, (v,u_0) \in E \,|\, \sigma_0(v) = \sigma_0(u), \sigma_0(u) \neq \sigma_0(u_0)\big)\mathbb{P}\big(\sigma_0(v) = \sigma_0(u)\big) \\
&\qquad + \mathbb{P}\big((v,u) \in E, (v,u_0) \in E \,|\, \sigma_0(v) \neq \sigma_0(u), \sigma_0(u) \neq \sigma_0(u_0)\big)\mathbb{P}\big(\sigma_0(v) \neq \sigma_0(u)\big) \\
&\quad = ab,
\end{aligned}$$

implying the second statement. ∎

The following lemma will be used to bound the misclassification probability of $u \in V_{i_1} \setminus \{u_0\}$ using the thresholding rule given in Algorithm 2, Line 3. Let $\mathcal{T}_{u_0,u} = \{N_{u_0,u} > (a+b)^2(|V_{i_1}| - 2)/4\}$. We define constants $\eta_1 = \exp[(a-b)^4/4]$ and $c_1 = \delta(a-b)^4/8$.

LEMMA 4.8. *For any vertex $u \in V_{i_1} \setminus \{u_0\}$ and any $m_{i_1} \geq \delta \log n$, we have*

$$\max\Big\{\mathbb{P}\big(\mathcal{T}^c_{u_0,u} \,\big|\, \sigma_0(u) = \sigma_0(u_0), |V_{i_1}| = m_{i_1}\big),\ \mathbb{P}\big(\mathcal{T}_{u_0,u} \,\big|\, \sigma_0(u) \neq \sigma_0(u_0), |V_{i_1}| = m_{i_1}\big)\Big\} \leq \eta_1 n^{-c_1}.$$

Proof. Fix $m_{i_1} \geq \delta \log n$. Lemma 4.7 along with Hoeffding's inequality (Lemma 4.2) gives that

$$\begin{aligned}
&\mathbb{P}\big(\mathcal{T}^c_{u_0,u} \,\big|\, \sigma_0(u) = \sigma_0(u_0), |V_{i_1}| = m_{i_1}\big) \\
&= \mathbb{P}\big(N_{u_0,u} - (a^2+b^2)(m_{i_1} - 2)/2 \leq -(a-b)^2(m_{i_1} - 2)/4 \,\big|\, \sigma_0(u) = \sigma_0(u_0), |V_{i_1}| = m_{i_1}\big) \\
&\leq \exp\big(-(a-b)^4(m_{i_1} - 2)/8\big) \\
&\leq \exp(-(a-b)^4(\delta \log n - 2)/8) = \eta_1 n^{-c_1}.
\end{aligned}$$

Similarly,

$$\begin{aligned}
&\mathbb{P}\big(\mathcal{T}_{u_0,u} \,\big|\, \sigma_0(u) \neq \sigma_0(u_0), |V_{i_1}| = m_{i_1}\big) \\
&\quad = \mathbb{P}\big(N_{u_0,u} - ab(m_{i_1} - 2) > (a-b)^2(m_{i_1} - 2)/4 \,\big|\, \sigma_0(u) \neq \sigma_0(u_0), |V_{i_1}| = m_{i_1}\big) \\
&\quad \leq \exp\big(-(a-b)^4(m_{i_1} - 2)/8\big) \leq \eta_1 n^{-c_1}.
\end{aligned}$$

∎

The following proposition ensures the high probability of correct labeling for all vertices in V_{i_1}.

PROPOSITION 4.3. *Suppose that $a, b \in [0,1]$ with $a \neq b$. Then Line 8 of Algorithm 5 ensures that for any $\Delta > \delta$,*

$$\mathbb{P}\Big(\bigcap_{u \in V_{i_1}} \big\{\widehat{\sigma}(u) = \sigma_0(u_0)\sigma_0(u)\big\} \,\Big|\, \delta \log n \leq |V_{i_1}| \leq \Delta \log n\Big) \geq 1 - \eta_1 \Delta n^{-c_1} \log n.$$

Proof. For any $u \in V_{i_1} \setminus \{u_0\}$, when $m_{i_1} \geq \delta \log n$, Lemma 4.8 implies

$$\begin{aligned}
&\mathbb{P}\big(\widehat{\sigma}(u) \neq \sigma_0(u_0)\sigma_0(u) \,\big|\, |V_{i_1}| = m_{i_1}\big) \\
&\quad = \mathbb{P}\big(\widehat{\sigma}(u) = -1 \,\big|\, \sigma_0(u) = \sigma_0(u_0), |V_{i_1}| = m_{i_1}\big)\mathbb{P}\big(\sigma_0(u) = \sigma_0(u_0)\big)
\end{aligned}$$

$$
\begin{aligned}
&+ \mathbb{P}\big(\widehat{\sigma}(u) = 1 \,\big|\, \sigma_0(u_0) \neq \sigma_0(u), |V_{i_1}| = m_{i_1}\big)\mathbb{P}\big(\sigma_0(u_0) \neq \sigma_0(u)\big) \\
&= \mathbb{P}\big(\mathcal{T}^c_{u,u_0} \,\big|\, \sigma_0(u) = \sigma_0(u_0), |V_{i_1}| = m_{i_1}\big)/2 + \mathbb{P}\big(\mathcal{T}_{u,u_0} \,\big|\, \sigma_0(u) \neq \sigma_0(v), |V_{i_1}| = m_{i_1}\big)/2 \\
&\leq \eta_1 n^{-c_1}.
\end{aligned}
$$

Thus, for any $\delta \log n \leq m_{i_1} \leq \Delta \log n$, the union bound yields that

$$
\begin{aligned}
\mathbb{P}\Big(\bigcap_{u\in V_1} \big\{\widehat{\sigma}(u) = \sigma_0(u_0)\sigma_0(u)\big\} \,\Big|\, |V_{i_1}| = m_{i_1}\Big) &= 1 - \mathbb{P}\Big(\bigcup_{u\in B_1} \big\{\widehat{\sigma}(u) \neq \sigma_0(u_0)\sigma_0(u)\big\} \,\Big|\, |V_{i_1}| = m_{i_1}\Big) \\
&\geq 1 - m_{i_1}\eta_1 n^{-c_1} \geq 1 - \eta_1 \Delta n^{-c_1} \log n.
\end{aligned}
$$

It follows that

$$
\mathbb{P}\Big(\bigcap_{u\in V_{i_1}} \big\{\widehat{\sigma}(u) = \sigma_0(u_0)\sigma_0(u)\big\} \,\Big|\, \delta \log n \leq |V_{i_1}| \leq \Delta \log n\Big) \geq 1 - \eta_1 \Delta n^{-c_1} \log n.
$$

□

4.3 Propagating labels among occupied blocks. We now demonstrate that the `Propagate` subroutine (Lines 9-10 of Algorithm 5) ensures that all occupied blocks are classified with at most M mistakes, for a suitable constant M.

We introduce a vector $m = (m_1, \cdots, m_{(n/(\chi \log n))}) \in \mathbb{Z}_+^{(n/(\chi \log n))}$ and define the event

$$
\mathcal{V}(m) = \{|V_i| = m_i \text{ for } i \in [n/(\chi \log n)]\}.
$$

Each $\mathcal{V}(m)$ corresponds to a specific (χ, δ)-visibility graph H. Thus, conditioned on an event $\mathcal{V}(m)$ that ensures the connectivity of H, the occupied block set $V^\dagger$ and the propagation ordering over $V^\dagger$ are uniquely determined. To simplify the analysis, we fix the vector m in what follows, and condition on some event $\mathcal{V}(m) \subset \mathcal{H}$, recalling that $\mathcal{H} = \{H \text{ is connected}\}$. We write $\mathbb{P}_m(\cdot) = \mathbb{P}(\cdot \,|\, \mathcal{V}(m))$ as a reminder. Note that conditioned on $\mathcal{V}(m)$, the labels of vertices are independent, and the edges are independent conditioned on the vertex labels.

We denote the *configuration* of a block as a vector $z = (z(1,1), z(1,-1), z(-1,-1), z(-1,1)) \in Z_+^4$, where each entry represents the count of vertices labeled as $+1$ or -1 by σ_0 and $\widehat{\sigma}$. For $i \in V^\dagger$, the event $\mathcal{C}_i(z)$ signifies that the occupied block V_i possesses a configuration z such that

$$
\begin{aligned}
&|\{u \in V_i, \sigma_0(u) = \sigma_0(u_0), \widehat{\sigma}(u) = 1\}| = z(1,1) \\
&|\{u \in V_i, \sigma_0(u) = \sigma_0(u_0), \widehat{\sigma}(u) = -1\}| = z(1,-1) \\
&|\{u \in V_i, \sigma_0(u) \neq \sigma_0(u_0), \widehat{\sigma}(u) = -1\}| = z(-1,-1) \\
&|\{u \in V_i, \sigma_0(u) \neq \sigma_0(u_0), \widehat{\sigma}(u) = 1\}| = z(-1,1).
\end{aligned}
$$

Consider $i \in V^\dagger \setminus \{i_1\}$ and a configuration $z \in \mathbb{Z}_+^4$. The key observation is that because the labels $\{\widehat{\sigma}(u) : u \in V_i\}$ are determined using disjoint sets of edges, the labels $\{\widehat{\sigma}(u) : u \in V_i\}$ are independent conditioned on $\mathcal{C}_{p(i)}$. Thus, the number of mistakes on V_i can be dominated by a binomial random variable. To formalize this observation, we define constants $M = 5/[(a-b)^2\delta]$, $c_2 = (a-b)^2\delta/4$, and $\eta_2 = \exp(2(a-b)^2 M)$. Let $\mathcal{A}_i$ be the event that $\widehat{\sigma}$ makes at most M mistakes on V_i:

$$
\mathcal{A}_i = \big\{|\{u \in V_i : \widehat{\sigma}(u) \neq \sigma_0(u_0)\sigma_0(u)\}| \leq M\big\}.
$$

The following lemma bounds the probability of misclassifying a given vertex using Algorithm 3.

LEMMA 4.9. *Suppose that $a, b \in [0,1]$ and $a \neq b$, and fix $i \in V^\dagger \setminus \{i_1\}$. Fix $z \in \mathbb{Z}_+^4$ such that $z(1,1) + z(1,-1) + z(-1,-1) + z(-1,1) = m_{p(i)}$ and $z(1,-1) + z(-1,1) \leq M$ (so that $\mathcal{C}_{p(i)}(z) \subset \mathcal{A}_{p(i)}$). Then for any $u \in V_i$, we have*

$$
\mathbb{P}_m\big(\widehat{\sigma}(u) \neq \sigma_0(u_0)\sigma_0(u) \,\big|\, \mathcal{C}_{p(i)}(z)\big) \leq \eta_2 n^{-c_2}.
$$

Proof. We consider the case $a > b$. Let $\mathcal{J}_+ = \{|\{v \in V_{p(i)} : \widehat{\sigma}(v) = 1\}| \geq |\{v \in V_{p(i)} : \widehat{\sigma}(v) = -1\}|\}$. We first study the case when $\mathcal{J}_+$ holds. In this context, Lines 1-8 of Algorithm 3 are executed. Conditioned on any $\mathcal{C}_{p(i)}(z)$, we have $|\{v \in V_{p(i)} : \widehat{\sigma}(v) = 1\}| = z(1,1) + z(-1,1)$. Among these vertices $v \in V_{p(i)}$ with $\widehat{\sigma}(v) = 1$, $z(1,1)$ vertices have ground truth label $\sigma_0(u_0)$ and $z(-1,1)$ of them have label $-\sigma_0(u_0)$. We now bound the probability of making a mistake, meaning that $\widehat{\sigma}(u) \neq \sigma_0(u_0)\sigma_0(u)$.

If $\sigma_0(u) = \sigma_0(u_0)$, let $\{X_i\}_{i=1}^{z(1,1)}$ and $\{Y_i\}_{i=1}^{z(-1,1)}$ be independent random variables with $X_i \sim \text{Bernoulli}(a)$ and $Y_i \sim \text{Bernoulli}(b)$, and $Z = \sum_{i=1}^{z(1,1)} X_i + \sum_{i=1}^{z(-1,1)} Y_i$ with mean $\mu_Z = z(1,1)a + z(-1,1)b$. For any $u \in V_i$, we recall that $d_1^+(u, \widehat{\sigma}, V_{p(i)}) = |\{v \in V_{p(i)} : \widehat{\sigma}(v) = 1, \{u,v\} \in E\}|$ and observe that conditioned on $\{\sigma_0(u) = \sigma_0(u_0), \mathcal{C}_{p(i)}(z)\}$, the degree profile $d_1^+(u, \widehat{\sigma}, V_{p(i)})$ has the same distribution as Z. Thus, Hoeffding's inequality yields

$$\begin{aligned}
&\mathbb{P}_m\big(\widehat{\sigma}(u) \neq 1 \,\big|\, \sigma_0(u) = \sigma_0(u_0), \mathcal{C}_{p(i)}(z)\big) \\
&\quad = \mathbb{P}_m\Big(d_1^+(u, \widehat{\sigma}, V_{p(i)}) < (a+b)|\{v \in V_{p(i)} : \widehat{\sigma}(v) = 1\}|/2 \,\Big|\, \sigma_0(v) = \sigma_0(u_0), \mathcal{C}_{p(i)}(z)\Big) \\
&\quad = \mathbb{P}_m\big(Z < (a+b)(z(1,1) + z(-1,1))/2\big) \\
&\quad = \mathbb{P}_m\big(Z - \mu_Z < -(a-b)(z(1,1) - z(-1,1))/2\big) \\
&\quad \leq \exp\big(-\frac{(a-b)^2(z(1,1) - z(-1,1))^2}{2(z(1,1) + z(-1,1))}\big).
\end{aligned}$$

We recall that $\mathcal{J}_+$ implies $|\{v \in V_{p(i)} : \widehat{\sigma}(v) = 1\}| \geq |V_{p(i)}|/2 \geq \delta \log n/2$, and $z(1,-1) + z(-1,1) \leq M$. It follows that $z(1,1) + z(-1,1) \geq \delta \log n/2$ and $z(1,1) \geq \delta \log n/2 - M$. Thus,

$$\begin{aligned}
\mathbb{P}_m\big(\widehat{\sigma}(u) \neq 1 \,\big|\, \sigma_0(u) = \sigma_0(u_0), \mathcal{C}_{p(i)}(z)\big) &\leq \exp\big(-\frac{(a-b)^2(z(1,1) - M)^2}{2(z(1,1) + M)}\big) \\
&\leq \exp\big(-(a-b)^2(z(1,1) - 3M)/2\big) \leq \eta_2 \exp\big(-(a-b)^2\delta \log n/4\big) = \eta_2 n^{-c_2},
\end{aligned} \tag{4.5}$$

where the last two inequalities hold since $(z(1,1) - M)^2/(z(1,1) + M) \geq z(1,1) - 3M$ and $z(1,1) \geq \delta \log n/2 - M$.

Similarly, when $\sigma_0(u) \neq \sigma_0(u_0)$, let $\{X_i\}_{i=1}^{z(-1,1)}$ and $\{Y_i\}_{i=1}^{z(1,1)}$ be independent random variables with $X_i \sim \text{Bernoulli}(a)$ and $Y_i \sim \text{Bernoulli}(b)$, and $\widetilde{Z} = \sum_{i=1}^{z(1,1)} Y_i + \sum_{i=1}^{z(-1,1)} X_i$ with mean $\mu_{\widetilde{Z}} = z(1,1)b + z(-1,1)a$. For any $u \in V_i$, we observe that $d_1^+(u, \widehat{\sigma}, V_{p(i)})$ has the same distribution as $\widetilde{Z}$, conditioned on $\big\{\sigma_0(u) \neq \sigma_0(u_0), \mathcal{C}_{p(i)}(z)\big\}$. By similar steps as the case $\sigma_0(u) = \sigma_0(u_0)$, we obtain

$$\begin{aligned}
&\mathbb{P}_m\big(\widehat{\sigma}(u) \neq -1 \,\big|\, \sigma_0(v) \neq \sigma_0(u_0), \mathcal{C}_{p(i)}(z)\big) \\
&\quad = \mathbb{P}_m\Big(d_1^+(u, \widehat{\sigma}, V_{p(i)}) \geq (a+b)|\{v \in V_{p(i)} : \widehat{\sigma}(v) = 1\}|/2 \,\Big|\, \sigma_0(v) \neq \sigma_0(u_0), \mathcal{C}_{p(i)}(z)\Big) \\
&\quad = \mathbb{P}\big(\widetilde{Z} \geq (a+b)(z(1,1) + z(-1,1))/2\big) \\
&\quad = \mathbb{P}\big(\widetilde{Z} - \mu_{\widetilde{Z}} \geq (a-b)(z(1,1) - z(-1,1))/2\big) \\
&\quad \leq \exp\big(-\frac{(a-b)^2(z(1,1) - z(-1,1))^2}{2(z(1,1) + z(-1,1))}\big) \\
&\quad \leq \eta_2 n^{-c_2}.
\end{aligned} \tag{4.6}$$

The bounds (4.5) and (4.6) together imply

$$\mathbb{P}_m\big(\widehat{\sigma}(u) \neq \sigma_0(u_0)\sigma_0(u) \,\big|\, \mathcal{C}_{p(i)}(z)\big) \leq \eta_2 n^{-c_2}.$$

We can derive symmetric analysis for z such that $\mathcal{J}_+^c$ holds, in which case Algorithm 3 executes Lines 9-16. The proof is complete for the case $a > b$. The analysis for the case $b > a$ is similar. ∎

Before proceeding further and showing the success of the propagation, we state a lemma that, with high probability, all blocks contain $O(\log n)$ vertices.

LEMMA 4.10. *For the blocks obtained from Line 3 in Algorithm 5, there exists a constant $\Delta > 0$ such that*

$$\mathbb{P}\Big(\bigcap_{i=1}^{n/(\chi \log n)} \big\{ |V_i| < \Delta \log n \big\} \Big) = 1 - o(1).$$

Proof. For a block B_i with $\mathrm{vol}(B_i) = \chi \log n$, we have $|V_i| \sim \mathrm{Poisson}(\lambda\chi \log n)$. Thus, the Chernoff bound in Lemma 4.1 implies that, for $\Delta > (\lambda\chi + 1 + \sqrt{2\lambda\chi + 1})$, we have

$$\mathbb{P}(|V_i| \geq \Delta \log n) \leq \exp\Big(- \frac{(\Delta - \lambda\chi)^2 \log n}{2\Delta} \Big) = n^{-\frac{(\Delta - \lambda\chi)^2}{2\Delta}} < n^{-1},$$

where the last inequality holds by straightforward calculation. Thus, the union bound gives that

$$\mathbb{P}\Big(\bigcap_{i=1}^{n/(\chi \log n)} \big\{ |V_i| < \Delta \log n \big\} \Big) = 1 - \mathbb{P}\Big(\bigcup_{i=1}^{n/(\chi \log n)} \big\{ |V_i| \geq \Delta \log n \big\} \Big) > 1 - \frac{n}{\chi \log n} \cdot n^{-1} = 1 - o(1).$$

□

For $\Delta > 0$ given by Lemma 4.10, we define $\mathcal{I}$ as follows and have $\mathbb{P}(\mathcal{I}) = 1 - o(1)$.

$$\mathcal{I} = \bigcap_{i=1}^{n/(\chi \log n)} \{ |V_i| < \Delta \log n \}.$$

The following lemma concludes that Phase I makes few mistakes on occupied blocks during the propagation.

LEMMA 4.11. *Let $G \sim GSBM(\lambda, n, a, b, d)$ with $\lambda\nu_d > 1$, $a, b \in [0, 1]$, and $a \neq b$, and $\widehat{\sigma} : V \to \{-1, 0, 1\}$ be the output of Phase I in Algorithm 5 on input G. Suppose m is such that $\mathcal{V}(m) \subset \mathcal{I} \cap \mathcal{H}$. Lines 9-10 of Algorithm 5 ensure that*

$$\mathbb{P}_m\Big(\bigcap_{i \in V^\dagger} \mathcal{A}_i \Big) \geq \big(1 - \eta_1 \Delta n^{-c_1} \log n\big)\Big(1 - \frac{\eta_3 n^{-\frac{1}{8}}}{\chi \log n}\Big).$$

Proof. Consider $i_j \in V^\dagger$ for $2 \leq j \leq |V^\dagger|$, and fix $z \in \mathbb{Z}_+^4$ such that

$$z(1,1) + z(1,-1) + z(-1,-1) + z(-1,1) = m_{p(i_j)} \text{ and } z(1,-1) + z(-1,1) \leq M. \tag{4.7}$$

Observe that the events that $u \in V_{i_j}$ is mislabeled by $\widehat{\sigma}$ are mutually independent conditioned on $\mathcal{C}_{p(i_j)}(z)$. Lemma 4.9 shows that each individual vertex in V_{i_j} is misclassified with probability at most $\eta_2 n^{-c_2}$, conditioned on $\mathcal{C}_{p(i_j)}(z)$. It follows that conditioned on $\mathcal{C}_{p(i_j)}(z)$,

$$|\{u \in V_{i_j} : \widehat{\sigma}(u) \neq \sigma_0(u_0)\sigma_0(u)\}| \overset{\text{st}}{\preceq} \mathrm{Bin}\left(\Delta \log n, \eta_2 n^{-c_2}\right) =: \xi.$$

Let $\mu_\xi = \mathbb{E}[\xi] = \eta_2 \Delta n^{-c_2} \log n$. Using the Chernoff bound (Lemma 4.3), we obtain

$$\begin{aligned}
\mathbb{P}_m\big(\mathcal{A}_{i_j}^c \,\big|\, \mathcal{C}_{p(i_j)}(z)\big) &= \mathbb{P}_m\big(|\{u \in V_{i_j} : \widehat{\sigma}(u) \neq \sigma_0(u_0)\sigma_0(u)\}| > M \,\big|\, \mathcal{C}_{p(i_j)}(z)\big) \\
&\leq \mathbb{P}(\xi > M) \\
&= \mathbb{P}\big(\xi - \mu_\xi > (M/\mu_\xi - 1)\mu_\xi\big) \\
&\leq e^{M - \mu_\xi}(\mu_\xi / M)^M \\
&\leq (e\eta_2 \Delta / M)^M (\log n)^M n^{-c_2 M} \\
&\leq \eta_3 n^{-9/8}. \qquad (4.8)
\end{aligned}$$

The last inequality holds since $c_2 M = 5/4$ by definition and $(\log n)^M \le n^{1/8}$ for large enough n. Since $\mathcal{A}_{i_j}$ is independent of $\{\mathcal{A}_{i_k} : k < j, k \neq p(i_j)\}$ conditioned on $\mathcal{C}_{p(i_j)}$, (4.8) implies

$$\mathbb{P}_m\Big(\mathcal{A}_{i_j}^c \,\Big|\, \mathcal{C}_{p(i_j)}(z), \bigcap_{k<j: i_k \neq p(i_j)} \mathcal{A}_{i_k}\Big) \le \eta_3 n^{-9/8}.$$

Furthermore, since (4.8) is a uniform bound over all z satisfying (4.7), it follows that

$$\mathbb{P}_m\Big(\mathcal{A}_{i_j}^c \mid \bigcap_{k<j} \mathcal{A}_{i_k}\Big) \le \eta_3 n^{-9/8}.$$

Thus, combining Proposition 4.3 with the preceding bound, we have

$$\begin{aligned}
\mathbb{P}_m\Big(\bigcap_{i\in V^\dagger} \mathcal{A}_i\Big) &= \mathbb{P}_m(\mathcal{A}_{i_1}) \cdot \prod_{j=2}^{|V^\dagger|} \mathbb{P}_m(\mathcal{A}_{i_j} \,|\, \mathcal{A}_{i_{j-1}}, \cdots, \mathcal{A}_{i_1}) \\
&\ge \big(1 - \eta_1 \Delta n^{-c_1} \log n\big)\Big(1 - \eta_3 n^{-\frac{9}{8}}\Big)^{|V^\dagger|-1} \\
&\ge \big(1 - \eta_1 \Delta n^{-c_1} \log n\big)\Big(1 - \eta_3 n^{-\frac{9}{8}}\Big)^{\frac{n}{\chi \log n}} \\
&\ge \big(1 - \eta_1 \Delta n^{-c_1} \log n\big)\Big(1 - \frac{\eta_3 n^{-\frac{1}{8}}}{\chi \log n}\Big),
\end{aligned}$$

where we use the fact that there are $n/\chi \log n$ blocks in total along with Bernoulli's inequality. ∎

Combining the aforementioned results, we now prove the success of Phase I in Theorem 4.1. We highlight that since $\eta > 0$ is arbitrary, the following equation (4.10) implies Theorem 2.4.

THEOREM 4.1. *Given GSBM(λ, n, a, b, d) with $a, b \in [0,1]$, $a \neq b$, and $d = 1$ and $\lambda > 1$, or $d \ge 2$ and $\lambda \nu_d > 1$. Fix any $\eta > 0$. Let $\kappa = \nu_d(1 + \sqrt{d}\chi^{1/d})^d/\chi$. Let $\widehat{\sigma}$ be the labeling obtained from Phase I with $\chi > 0$ satisfying* (4.1) *and $\delta > 0$ satisfying* (4.2) *and $\delta < \eta/\kappa$, respectively. Then there exists a constant M such that $\widehat{\sigma}$ makes at most M mistakes on every occupied block, with high probability,*

$$\mathbb{P}\Big(\bigcap_{i\in V^\dagger} \big\{|\{v \in V_i : \widehat{\sigma}(v) \neq \sigma_0(u_0)\sigma_0(v)\}| \le M\big\}\Big) = 1 - o(1). \tag{4.9}$$

Moreover, it follows that

$$\mathbb{P}\big(|\{v \in V : \widehat{\sigma}(v) \neq \sigma_0(u_0)\sigma_0(v)\}| \le \eta n/(\chi\kappa)\big) = 1 - o(1) \tag{4.10}$$

and

$$\mathbb{P}\Big(\bigcap_{u\in V} \big\{|v \in \mathcal{N}(u) \colon \widehat{\sigma}(v) \neq \sigma_0(u_0)\sigma_0(v)| \le \eta \log n\big\}\Big) = 1 - o(1). \tag{4.11}$$

Proof. Fixing any $\eta > 0$, we consider $\chi > 0$ satisfying (4.1) and $\delta > 0$ satisfying (4.2) and $\delta < \eta/\kappa$, respectively. Given any m such that $\mathcal{V}(m) \subset \mathcal{I} \cap \mathcal{H}$, for occupied blocks, Proposition 4.11 yields the existence of a constant $M > 0$ such that

$$\mathbb{P}_m\Big(\bigcap_{i\in V^\dagger} \{|\{v \in V_i : \widehat{\sigma}(v) \neq \sigma_0(u_0)\sigma_0(v)\}| \le M\}\Big) \ge \big(1 - \eta_1 \Delta n^{-c_1} \log n\big)\Big(1 - \frac{\eta_3 n^{-\frac{1}{8}}}{\chi \log n}\Big).$$

Since the above bound is uniform over all m such that $\mathcal{V}(m) \subset \mathcal{I} \cap \mathcal{H}$, we have

$$\mathbb{P}\Big(\bigcap_{i\in V^\dagger} \big\{|v \in V_i \colon \widehat{\sigma}(v) \neq \sigma_0(u_0)\sigma_0(v)| \le \delta \log n\big\}\Big)$$

$$\geq \sum_{m\colon \mathcal{V}(m)\subset\mathcal{I}\cap\mathcal{H}} \mathbb{P}_m\Big(\bigcap_{i\in V^\dagger}\big\{|v\in V_i\colon \widehat{\sigma}(v)\neq\sigma_0(u_0)\sigma_0(v)|\leq\delta\log n\big\}\Big)\cdot\mathbb{P}\big(\mathcal{V}(m)\big)$$

$$\geq \big(1-\eta_1\Delta n^{-c_1}\log n\big)\Big(1-\frac{\eta_3 n^{-\frac{1}{8}}}{\chi\log n}\Big)\cdot\mathbb{P}\big(\mathcal{I}\cap\mathcal{H}\big)=1-o(1),$$

where the last step holds by Propositions 4.1 and 4.2, and Lemma 4.10. Thus, we have proven (4.9).

Since $\delta\log n > M$ for n large enough, it follows that

$$\mathbb{P}\Big(\bigcap_{i\in[n/\chi\log n]}\{|\{v\in V_i:\widehat{\sigma}(v)\neq\sigma_0(u_0)\sigma_0(v)\}|\leq\delta\log n\}\Big)=1-o(1). \tag{4.12}$$

On the one hand, if $\widehat{\sigma}$ makes fewer than $\delta\log n$ mistakes on V_i for all $i\in[n/(\chi\log n)]$, then $\widehat{\sigma}$ makes fewer than $\delta n/\chi\leq\eta n/(\chi\kappa)$ mistakes in $\mathcal{S}_{d,n}$. Thus, (4.10) follows from (4.12). On the other hand, if $\widehat{\sigma}$ makes fewer than $\delta\log n$ mistakes on V_i for all $i\in[n/(\chi\log n)]$, then there will be fewer than $\delta\kappa\log n\leq\eta\log n$ mistakes in all vertices' neighborhood since each neighborhood $\mathcal{N}(u)$ intersects at most κ blocks. Thus, (4.11) also follows from (4.12). ∎

5 Phase II: Proof of exact recovery

Before proving Theorem 2.2, we first show a concentration bound. We define vectors in $\mathbb{R}^4$,

$$x=\lambda\nu_d\log n[a,1-a,b,1-b]/2,\quad y=\lambda\nu_d\log n[b,1-b,a,1-a]/2, \tag{5.1}$$

and random variables $\widetilde{D}=[D_1^+,D_1^-,D_{-1}^+,D_{-1}^-]\sim\text{Poisson}(x)$, and X as a linear function of $\widetilde{D}$,

$$X=-\log\big(\frac{a}{b}\big)\big(D_1^+-D_{-1}^+\big)-\log\big(\frac{1-a}{1-b}\big)\big(D_1^--D_{-1}^-\big). \tag{5.2}$$

For any $t\in[0,1]$, let $D_t(x\|y)=\sum_{i\in[4]}(tx_i+(1-t)y_i-x_i^ty_i^{1-t})$ be an f-divergence. Let $D_+(x\|y)=\max_{t\in[0,1]}D_t(x\|y)=\max_{t\in[0,1]}D_t(y\|x)$ be the Chernoff-Hellinger divergence, as introduced by [5]. In particular, when x and y are defined in (5.1), the maximum is achieved at $t=1/2$ and we have $D_+(x\|y)=\lambda\nu_d(1-\sqrt{ab}-\sqrt{(1-a)(1-b)})\log n$.

Lemma 5.1. *For any constants $\rho>0$ and $\eta>0$, it holds for X defined in* (5.2) *that*

$$\mathbb{P}\big(X\geq-\rho\eta\log n\big)\leq n^{-\lambda\nu_d(1-\sqrt{ab}-\sqrt{(1-a)(1-b)})+\rho\eta/2}.$$

Proof. We will apply the Chernoff bound on X. First, we compute its moment-generating function. For $\widetilde{D}=[D_1^+,D_1^-,D_{-1}^+,D_{-1}^-]=(D_i)_{i=1}^4\sim\text{Poisson}(x)$, the definition of X in (5.2) can be written as

$$X=-\sum_{i=1}^4[D_i\log(x_i/y_i)-(x_i-y_i)].$$

We recall that for $\xi\sim\text{Poisson}(\mu)$ and $s\in\mathbb{R}$, we have $\mathbb{E}[\exp(s\xi)]=\exp[\mu(e^s-1)]$. Thus, we have

$$\begin{aligned}\mathbb{E}(e^{tX})&=\mathbb{E}\Big[\exp\big(-t\sum_{i=1}^4[D_i\log(x_i/y_i)-(x_i-y_i)]\big)\Big]\\&=\prod_{i=1}^4\exp(t(x_i-y_i))\cdot\mathbb{E}[\exp(t\log(y_i/x_i)D_i)]\\&=\prod_{i=1}^4\exp(t(x_i-y_i)+x_i(e^{t\log(y_i/x_i)}-1))\\&=\prod_{i=1}^4\exp((t-1)x_i-ty_i+x_i^{1-t}y_i^t)\end{aligned}$$

$$= \exp(-\sum_{i=1}^{4}((1-t)x_i + ty_i - x_i^{1-t}y_i^t)) = \exp(-D_t(y\|x)).$$

Therefore, the Chernoff bound ensures that for any $t > 0$, we have

$$\mathbb{P}\big(X \geq -\rho\eta \log n\big) \leq \frac{\mathbb{E}(e^{tX})}{e^{-t\rho\eta \log n}} = n^{t\rho\eta} \cdot \exp(-D_t(y\|x)).$$

It follows that

$$\begin{aligned}\mathbb{P}\big(X \geq -\rho\eta \log n\big) &\leq \inf_{t>0}\big\{n^{t\rho\eta} \cdot \exp(-D_t(y\|x))\big\} \\ &\leq n^{\rho\eta/2} \cdot \exp(-D_+(x\|y)) \\ &= n^{-\lambda\nu_d(1-\sqrt{ab}-\sqrt{(1-a)(1-b)})+\rho\eta/2}.\end{aligned}$$

□

Now we present the proof of Theorem 2.2, which ensures that Algorithm 5 achieves exact recovery.

Proof. [Proof of Theorem 2.2] We first fix a constant $c > \lambda$ and let $\mathcal{E}_0 = \{|V| < cn\}$. Since $|V| \sim \text{Poisson}(\lambda n)$, the Chernoff bound in Lemma 4.1 gives that

$$\mathbb{P}(\mathcal{E}_0^c) = \mathbb{P}(|V| > cn) \leq \exp\big(-\frac{(c-\lambda)^2 n}{2c}\big) = o(1).$$

For $\eta > 0$ to be determined, let $\mathcal{E}_1$ be the event that $\widehat{\sigma}$ makes at most $\eta \log n$ mistakes in the neighborhood for all vertices (Phase I succeeds); that is,

$$\mathcal{E}_1 = \bigcap_{u\in V}\big\{|v \in \mathcal{N}(u)\colon \widehat{\sigma}(v) \neq \sigma_0(u_0)\sigma_0(v)| \leq \eta \log n\big\}.$$

Theorem 4.1 ensures that $\mathbb{P}(\mathcal{E}_1) = 1 - o(1)$. Let $\mathcal{E}_2'$ be the event that Algorithm 5 achieves exact recovery and $\mathcal{E}_2$ be the event that all vertices are labeled correctly relative to $\sigma_0(u_0)$; that is,

$$\mathcal{E}_2' = \Big\{\bigcap_{u\in V}\{\widetilde{\sigma}(u) = \sigma_0(u)\}\Big\}\bigcup\Big\{\bigcap_{u\in V}\{\widetilde{\sigma}(u) = -\sigma_0(u)\}\Big\}, \quad \mathcal{E}_2 = \bigcap_{u\in V}\{\widetilde{\sigma}(u) = \sigma_0(u_0)\sigma_0(u)\}.$$

Then we have $\mathbb{P}(\mathcal{E}_2') \geq \mathbb{P}(\mathcal{E}_2)$. Since $\mathbb{P}(\mathcal{E}_0), \mathbb{P}(\mathcal{E}_1) = 1 - o(1)$, it follows that

$$\mathbb{P}(\mathcal{E}_2^c) \leq \mathbb{P}(\mathcal{E}_2^c \cap \mathcal{E}_1 \cap \mathcal{E}_0) + \mathbb{P}(\mathcal{E}_1^c) + \mathbb{P}(\mathcal{E}_0^c) = \mathbb{P}(\mathcal{E}_2^c \cap \mathcal{E}_1 \cap \mathcal{E}_0) + o(1). \tag{5.3}$$

Note that we analyze $\mathbb{P}(\mathcal{E}_2^c \cap \mathcal{E}_1 \cap \mathcal{E}_0)$ rather than $\mathbb{P}(\mathcal{E}_2^c \,|\, \mathcal{E}_1, \mathcal{E}_0)$, in order to preserve the data distribution. Next, we would like to show that the probability of misclassifying a vertex v is $o(1/n)$, and conclude that the probability of misclassifying *any* vertex is $o(1)$. To formalize such an argument, sample $N \sim \text{Poisson}(\lambda n)$, and generate $\max\{N, cn\}$ points in the region $\mathcal{S}_{d,n}$ uniformly at random. Note that on the event $\mathcal{E}_0$, we have $\max\{N, cn\} = cn$. Label the points in order, and set $\widehat{\sigma}(u_0) = 1$. In this way, the first N points form a Poisson point process with intensity λ. We can simulate Algorithm 5 on the first N points. To bound the failure probability of Phase II, we can assume that any $v \in \{N+1, \ldots, cn\}$ must also be classified (by thresholding $\tau(v, \sigma)$, computed only using edge/non-edge observations between v and $u \in [N]$)). For $v \in [cn]$, let

$$\mathcal{E}_2(v) = \{\widetilde{\sigma}(v) = \sigma_0(u_0)\sigma_0(v)\}.$$

Then

$$\mathcal{E}_2^c \cap \mathcal{E}_1 \cap \mathcal{E}_0 \subseteq \bigcup_{v=1}^{cn}\{\mathcal{E}_2(v)^c \cap \mathcal{E}_1 \cap \mathcal{E}_0\} \subseteq \bigcup_{v=1}^{cn}\{\mathcal{E}_2(v)^c \cap \mathcal{E}_1\},$$

so that a union bound yields

$$\mathbb{P}\left(\mathcal{E}_2^c \cap \mathcal{E}_1 \cap \mathcal{E}_0\right) \le \sum_{v=1}^{cn} \mathbb{P}\left(\mathcal{E}_2(v)^c \cap \mathcal{E}_1\right). \tag{5.4}$$

Fix $v \in [cn]$. In order to bound $\mathbb{P}\left(\mathcal{E}_2(v)^c \cap \mathcal{E}_1\right)$, we classify v according to running the `Refine` algorithm with respect to edge/non-edge observations between v and $u \in [N]$. Analyzing $\mathcal{E}_2(v)^c \cap \mathcal{E}_1$ now reduces to analyzing robust Poisson testing. Let $W(v) = \{\sigma : \mathcal{N}(v) \to \{-1, 0, 1\}\}$ and d_H be the Hamming distance. We define the set of all estimators that differ from σ_0 on at most $\eta \log n$ vertices in $\mathcal{N}(v)$, relative to $\sigma_0(u_0)$, as

$$\begin{aligned} W'(v;\eta) &= \{\sigma \in W(v) \colon d_H(\sigma(\cdot), \sigma_0(u_0)\sigma_0(\cdot)) \le \eta \log n\} \\ &= \{\sigma \in W(v) \colon d_H(\sigma_0(u_0)\sigma(\cdot), \sigma_0(\cdot)) \le \eta \log n\}. \end{aligned}$$

Let $\mathcal{E}_v$ be the event that there exists $\sigma \in W'(v;\eta)$ such that Poisson testing with respect to σ fails on vertex v when $\mathcal{E}_2$ holds; that is,

$$\begin{aligned} \mathcal{E}_v = &\Big[\big\{\sigma_0(v) = 1\big\} \bigcap \Big(\bigcup_{\sigma \in W'(v;\eta)} \big\{\tau(v, \sigma_0(u_0)\sigma) \le 0\big\}\Big)\Big] \\ &\bigcup \Big[\big\{\sigma_0(v) = -1\big\} \bigcap \Big(\bigcup_{\sigma \in W'(v;\eta)} \big\{\tau(v, \sigma_0(u_0)\sigma) \ge 0\big\}\Big)\Big]. \end{aligned} \tag{5.5}$$

We provide some insights into the definition of $\mathcal{E}_v$. Recall that $\sigma_{\mathsf{genie}}(v) = \operatorname{sign}(\tau(v, \sigma_0))$ defined in (3.1) picks the event with the larger likelihood between $\{\sigma_0(v) = 1\}$ and $\{\sigma_0(v) = -1\}$. Thus, for example, suppose that $\sigma_0(v) = 1$, then $\sigma_{\mathsf{genie}}(v)$ makes a mistake when $\tau(v, \sigma_0) \le 0$. We consider any $\sigma \in W'(v;\eta)$. Since $\sigma_0(u_0)\sigma(u) = \sigma_0(u)$ for most $u \in \mathcal{N}(v)$, $d(v, \sigma_0(u_0)\sigma)$ and $d(v, \sigma_0)$ and thus $\tau(v, \sigma_0(u_0)\sigma)$ and $\tau(v, \sigma_0)$ are close. Formalizing the intuition, suppose that $\sigma_0(v) = 1$. If $\sigma_0(u_0) = 1$, then for $\mathcal{E}_2$ to hold, we must classify v as $+1$ to be correct relative to $\sigma_0(u_0)$. Thus, v is misclassified relative to σ whenever $\tau(v, \sigma) \le 0$. If $\sigma_0(v) = 1$ and $\sigma_0(u_0) = -1$, then we must classify v as -1. Then v is misclassified relative to σ whenever $\tau(v, \sigma) \ge 0$. As a summary, failure in the case $\sigma_0(v) = 1$ means $\tau(v, \sigma_0(u_0)\sigma) \le 0$.

It follows that

$$\mathbb{P}\left(\mathcal{E}_2(v)^c \cap \mathcal{E}_1\right) \le \mathbb{P}(\mathcal{E}_v). \tag{5.6}$$

We aim to show that for $\eta > 0$ sufficiently small, $\mathbb{P}(\mathcal{E}_v) = n^{-(1+\Omega(1))}$. Due to the uniform prior on $\sigma_0(v)$, we have

$$\mathbb{P}(\mathcal{E}_v) = \frac{1}{2}\big[\mathbb{P}(\mathcal{E}_v \mid \sigma_0(v) = 1) + \mathbb{P}(\mathcal{E}_v \mid \sigma_0(v) = -1)\big]. \tag{5.7}$$

We now bound the first term in (5.7). Let $D \in \mathbb{Z}_+^4$ represent the ground-truth degree profile of vertex v. We consider a realization $D = d(v, \sigma_0)$ and the induced $\tau(v, \sigma_0)$. Next, we bound the distance $|\tau(v, \sigma_0(u_0)\sigma) - \tau(v, \sigma_0)|$ for any $\sigma \in W'(v;\eta)$. We note that the edges and non-edges are fixed in a given graph G; that is, for any $\sigma \in W(v)$, we have

$$\begin{aligned} d_1^+(u, \sigma_0(u_0)\sigma) + d_{-1}^+(u, \sigma_0(u_0)\sigma) &= d_1^+(u, \sigma_0) + d_{-1}^+(u, \sigma_0), \\ d_1^-(u, \sigma_0(u_0)\sigma) + d_{-1}^-(u, \sigma_0(u_0)\sigma) &= d_1^-(u, \sigma_0) + d_{-1}^-(u, \sigma_0). \end{aligned}$$

Let $\alpha = d_1^+(u, \sigma_0(u_0)\sigma) - d_1^+(u, \sigma_0) = -(d_{-1}^+(u, \sigma_0(u_0)\sigma) - d_{-1}^+(u, \sigma_0))$ and $\beta = d_1^-(u, \sigma_0(u_0)\sigma) - d_1^-(u, \sigma_0) = -(d_{-1}^-(u, \sigma_0(u_0)\sigma) - d_{-1}^-(u, \sigma_0))$. It follows that

$$\begin{aligned} \tau(v, \sigma_0(u_0)\sigma) - \tau(v, \sigma_0) &= \log\big(\frac{1-a}{1-b}\big)[d_1^-(u, \sigma_0(u_0)\sigma) - d_1^-(u, \sigma_0) - (d_{-1}^-(u, \sigma_0(u_0)\sigma) - d_{-1}^-(u, \sigma_0))] \\ &\quad + \log\big(\frac{a}{b}\big)[d_1^+(u, \sigma_0(u_0)\sigma) - d_1^+(u, \sigma_0) - (d_{-1}^+(u, \sigma_0(u_0)\sigma) - d_{-1}^+(u, \sigma_0))] \\ &= 2\Big[\alpha \cdot \log\big(\frac{a}{b}\big) + \beta \cdot \log\big(\frac{1-a}{1-b}\big)\Big]. \end{aligned}$$

For any $\sigma \in W'(v;\eta)$, recalling that $d_H(\sigma_0(u_0)\sigma(\cdot), \sigma_0(\cdot)) \le \eta \log n$, we have $|\alpha| \le \eta \log n$ and $|\beta| \le \eta \log n$. Thus, we define $\rho = 2 \cdot [|\log(a/b)| + |\log((1-a)/(1-b))|]$ and have

$$\big|\tau(v, \sigma_0(u_0)\sigma) - \tau(v, \sigma_0)\big| \le 2\Big[|\alpha| \cdot \big|\log\big(\frac{a}{b}\big)\big| + |\beta| \cdot \big|\log\big(\frac{1-a}{1-b}\big)\big|\Big] \le \rho\eta \log n.$$

We define a set $Y \subset \mathbb{Z}_+^4$ as follows:

$$Y = \Big\{d = (d_1^+, d_1^-, d_{-1}^+, d_{-1}^-) \in \mathbb{Z}_+^4 : \log\big(\frac{a}{b}\big)\big(d_1^+ - d_{-1}^+\big) + \log\big(\frac{1-a}{1-b}\big)\big(d_1^- - d_{-1}^-\big) \le \rho\eta \log n\Big\}.$$

Conditioned on $\{\sigma_0(v) = 1\}$, Poisson testing fails relative to σ when $\tau(v, \sigma_0(u_0)\sigma) \le 0$. Thus,

$$\begin{aligned}
&\mathbb{P}\big(\mathcal{E}_v \,\big|\, \sigma_0(v) = 1\big) \\
&= \sum_{d \in \mathbb{Z}_+^4} \mathbb{P}\Big(\{D = d\} \bigcap \big\{ \min_{\sigma \in W'(v;\eta)} \tau(v, \sigma_0(u_0)\sigma) \le 0\big\} \,\big|\, \sigma_0(v) = 1\Big) \\
&\le \sum_{d \in \mathbb{Z}_+^4} \mathbb{P}\Big(\{D = d\} \bigcap \big\{\tau(v, \sigma_0) \le \rho\eta \log n\big\} \,\big|\, \sigma_0(v) = 1\Big) \\
&= \sum_{d \in Y} \mathbb{P}(D = d \mid \sigma_0(v) = 1).
\end{aligned}$$

To bound the above summation, we consider random variables $\widetilde{D} \sim \text{Poisson}(x)$ with x defined in (5.1) and X defined in (5.2). Recalling that $D \sim \widetilde{D}$ conditioned on $\sigma_0(v) = 1$, Lemma 5.1 gives that

$$\begin{aligned}
\mathbb{P}\big(\mathcal{E}_v \,\big|\, \sigma_0(v) = 1\big) &\le \sum_{d \in Y} \mathbb{P}\big(D = d \,\big|\, \sigma_0(v) = 1\big) \\
&= \mathbb{P}\big(\widetilde{D} \in Y\big) \\
&= \mathbb{P}\big(X \ge -\rho\eta \log n\big) \\
&\le n^{-\lambda\nu_d(1 - \sqrt{ab} - \sqrt{(1-a)(1-b)}) + \rho\eta/2}.
\end{aligned}$$

Since $\lambda\nu_d(1 - \sqrt{ab} - \sqrt{(1-a)(1-b)}) > 1$, we take $\eta = (\lambda\nu_d(1 - \sqrt{ab} - \sqrt{(1-a)(1-b)}) - 1)/\rho > 0$ and conclude that

$$\mathbb{P}\big(\mathcal{E}_v \,\big|\, \sigma_0(v) = 1\big) \le n^{-\frac{1}{2}(\lambda\nu_d(1 - \sqrt{ab} - \sqrt{(1-a)(1-b)}) + 1)} = o(1/n).$$

Similarly, we study the case conditioned on $\{\sigma_0(v) = -1\}$. Let $Y' = \{d = (d_1^+, d_1^-, d_{-1}^+, d_{-1}^-) \in \mathbb{Z}_+^4 : \log(a/b)(d_1^+ - d_{-1}^+) + \log((1-a)/(1-b))(d_1^- - d_{-1}^-) \ge -\rho\eta \log n\}$. The definition of $\mathcal{E}_v$ in (5.5) gives that

$$\begin{aligned}
&\mathbb{P}\big(\mathcal{E}_v \,\big|\, \sigma_0(v) = -1\big) \\
&= \sum_{d \in \mathbb{Z}_+^4} \mathbb{P}\Big(\{D = d\} \bigcap \big\{ \max_{\sigma \in W'(v;\eta)} \tau(v, \sigma_0(u_0)\sigma) \ge 0\big\} \,\big|\, \sigma_0(v) = -1\Big) \\
&\le \sum_{d \in \mathbb{Z}_+^4} \mathbb{P}\Big(\{D = d\} \bigcap \big\{\tau(v, \sigma_0) \ge -\rho\eta \log n\big\} \,\big|\, \sigma_0(v) = -1\Big) \\
&= \sum_{d \in Y'} \mathbb{P}(D = d \mid \sigma_0(v) = -1).
\end{aligned}$$

For the same $\widetilde{D} = [D_1^+, D_1^-, D_{-1}^+, D_{-1}^-] \sim \text{Poisson}(\lambda\nu_d \log n[a, 1-a, b, 1-b]/2)$, note that condition on $\sigma_0(v) = -1$, we have $D \sim [D_{-1}^+, D_{-1}^-, D_1^+, D_1^-]$. Thus, with the same X defined in (5.2), we have

$$\sum_{d \in Y'} \mathbb{P}(D = d \mid \sigma_0(v) = -1) = \mathbb{P}\big([D_{-1}^+, D_{-1}^-, D_1^+, D_1^-] \in Y'\big)$$

$$= \mathbb{P}\Big(\log\big(\frac{a}{b}\big)\big(D^+_{-1} - D^+_1\big) + \log\big(\frac{1-a}{1-b}\big)\big(D^-_{-1} - D^-_1\big) \geq -\rho\eta \log n \Big)$$
$$= \mathbb{P}(X \geq -\rho\eta \log n).$$

Thus, similarly, Lemma 5.1 gives that $\mathbb{P}(\mathcal{E}_v \,|\, \sigma_0(v) = -1) \leq n^{-\frac{1}{2}(\lambda\nu_d(1-\sqrt{ab}-\sqrt{(1-a)(1-b)})+1)}$. Therefore, the above bound together with (5.4), (5.6), and (5.7) implies $\mathbb{P}(\mathcal{E}_2^c \cap \mathcal{E}_1 \cap \mathcal{E}_0) = o(1)$. Finally, we have $\mathbb{P}((\mathcal{E}_2')^c) \leq \mathbb{P}(\mathcal{E}_2^c) = o(1)$ due to (5.3). ∎

6 Impossibility: Proof of Theorem 2.3

In this section, we prove the impossibility of exact recovery under the given conditions and complete the proof of Theorem 2.3. Recalling that Theorem 2.1 (Theorem 3.7 in [2]) has already established the impossibility when $\lambda > 0$, $d \in \mathbb{N}$, and $0 \leq b < a \leq 1$ satisfying (2.1). Here, we extend the same result to the case where the requirement $a > b$ is dropped.

PROPOSITION 6.1. *Let $\lambda > 0$, $d \in \mathbb{N}$, and $a, b \in [0,1]$ satisfy (2.1) and let $G_n \sim GSBM(\lambda, n, a, b, d)$. Then any estimator $\widetilde{\sigma}$ fails to achieve exact recovery.*

Proof. We note that the analysis of Theorem 2.1 builds upon Lemma 8.2 in [2], which itself relies on Lemma 11 from [5]. Lemma 11 provides the error exponent for hypothesis testing between Poisson random vectors, forming the basis for the impossibility result. Notably, only the CH-divergence criterion $\lambda\nu_d(1-\sqrt{ab}-\sqrt{(1-a)(1-b)}) < 1$ is needed to ensure the indistinguishability of the two Poisson distributions. Therefore, the impossibility in Theorem 2.1 can be readily extended to the case where the condition $a > b$ is dropped. ∎

Moreover, we show the impossibility of exact recovery for $d = 1$ and $\lambda < 1$.

PROPOSITION 6.2. *When $d = 1$, let $0 < \lambda < 1$ and $a, b \in [0,1]$ and let $G_n \sim GSBM(\lambda, n, a, b, d)$. Then any estimator $\widetilde{\sigma}$ fails to achieve exact recovery.*

Proof. When $d = 1$, we partition the interval $[-n/2, n/2]$ into $n/\log n$ blocks of length $\log n$ each. Notably, if there are $k \geq 2$ mutually non-adjacent empty blocks, the interval gets divided into $k \geq 2$ disjoint segments that lack mutual visibility. In such scenarios, achieving exact recovery becomes impossible as we can randomly flip the signs of one segment. Formally, suppose that there are k segments, where the ith segment contains blocks $\{B_j : j \in \mathrm{seg}(i)\}$ for $\mathrm{seg}(i) \subset [n/\log n]$. Then for any $s \in \{\pm 1\}^k$, the labeling σ_0 has the same posterior probability as $\sigma(\cdot; s)$, defined as

$$\sigma(v; s) = \sigma_0(v) \sum_{i \in [k]} s_i \sum_{j \in \mathrm{seg}(i)} \mathbb{1}_{\{v \in B_j\}}.$$

It follows that the error probability of the genie-aided estimator is at least $1 - 2/2^k = 1 - 1/2^{k-1}$, conditioned on there being k segments. Let $\mathcal{X}$ be the event of having at least two non-adjacent empty blocks (and thus two segments). The aforementioned observation means that if $\mathcal{X}$ holds, the error probability is at least $1/2$, and thus the exact recovery is unachievable.

We now prove that $\mathbb{P}(\mathcal{X}) = 1 - o(1)$ if $\lambda < 1$. Let $\mathcal{Y}_k$ be the event of having exactly k empty blocks, among which at least two of them are non-adjacent. Recalling that each block is independently empty with probability $\exp(-\lambda \log n) = n^{-\lambda}$, we have

$$\begin{aligned}
\mathbb{P}(\mathcal{X}) = \sum_{k=2}^{n/\log n} \mathbb{P}(\mathcal{Y}_k) &= \sum_{k=2}^{n/\log n - 1} \Big(\binom{n/\log n}{k} - n/\log n \Big) \big(n^{-\lambda}\big)^k \big(1 - n^{-\lambda}\big)^{n/\log n - k} \\
&\geq \sum_{k=1}^{n/\log n} \binom{n/\log n}{k} (n^{-\lambda})^k (1 - n^{-\lambda})^{n/\log n - k} - \frac{n}{\log n} (1 - n^{-\lambda})^{n/\log n} \sum_{k=1}^{n/\log n} \big[n^{-\lambda}/(1 - n^{-\lambda})\big]^k \\
&\geq 1 - \big(1 - n^{-\lambda}\big)^{n/\log n} - \big(1 - n^{-\lambda}\big)^{n/\log n} \cdot \frac{n}{\log n} \cdot \frac{n^{-\lambda}}{1 - 2n^{-\lambda}} \\
&\geq 1 - \big(1 - n^{-\lambda}\big)^{n/\log n} \cdot \big(1 + 2n^{1-\lambda}/\log n\big)
\end{aligned}$$

$$= 1 - \left[(1-n^{-\lambda})^{n^{\lambda}}\right]^{n^{1-\lambda}/\log n} \cdot \left(1 + 2n^{1-\lambda}/\log n\right)$$
$$= 1 - O\big(\exp(-n^{1-\lambda}/\log n) \cdot (1 + 2n^{1-\lambda}/\log n)\big) = 1 - o(1),$$

where the second inequality follows by calculating the Binomial series and the geometric series, and the last inequality holds since $1 - 2n^{-\lambda} \geq 1/2$ for large enough n. ∎

In summary, by combining Propositions 6.1 and 6.2, we complete the proof of Theorem 2.3.

7 Further related work

Our work contributes to the growing literature on community recovery in random geometric graphs, beginning with latent space models proposed in the network science and sociology literature (see for example [18, 19]). There have been several models for community detection in geometric graphs. The most similar to the one we study is the Soft Geometric Block Model (Soft GBM), proposed by Avrachenkov et al [7]. The main difference between their model and the GSBM is that the positions of the vertices are unknown. Avrachenkov et al [7] proposed a spectral algorithm for almost exact recovery, clustering communities using a higher-order eigenvector of the adjacency matrix. Using a refinement procedure similar to ours, [7] also achieved exact recovery, though only in the denser linear average degree regime.

A special case of the Soft GBM is the Geometric Block Model (GBM), proposed by Galhotra et al [14] with follow-up work including [10, 15]. In the GBM, community assignments are generated independently, and latent vertex positions are generated uniformly at random on the unit sphere. Edges are then formed according to parameters $\{\beta_{i,j}\}$, where pair of vertices u, v in communities i, j with locations Z_u, Z_v are connected if $\langle Z_u, Z_v\rangle \leq \beta_{i,j}$.

In the previously mentioned models, the vertex positions do not depend on the community assignments. In contrast, Abbe et al [3] proposed the Gaussian-Mixture Block Model (GMBM), where (latent) vertex positions are determined according to a mixture of Gaussians, one for each community. Edges are formed between all pairs of vertices whose distance falls below a threshold. A similar model was recently studied by Li and Schramm [24] in the high-dimensional setting. Additionally, Péché and Perchet [26] studied a geometric perturbation of the SBM, where vertices are generated according to a mixture of Gaussians, and the probability of connecting a pair of vertices is given by the sum of the SBM parameter and a function of the latent positions.

In addition, some works [6, 13] consider the task of recovering the geometric representation (locations) of the vertices in random geometric graphs as a form of community detection. Their setting differs significantly from ours. We refer to the survey [12] for an overview of the recent developments in non-parametric inference in random geometric graphs.

8 Conclusions and future directions

Our work identifies the information-theoretic threshold for exact recovery in the two-community, balanced, symmetric GSBM. A natural direction for future work is to consider the case of multiple communities, with general community membership probabilities and general edge probabilities. We believe that the information-theoretic threshold will again be given by a CH-divergence criterion, and a variant of our two-phase approach will achieve the threshold.

It would also be interesting to study other spatial network inference problems. For example, consider $\mathbb{Z}_2$-synchronization [4, 8, 22], a signal recovery problem motivated by applications to clock synchronization [16], robotics [28], and cryogenic electron microscopy [30]. In the standard version of the problem, each vertex is assigned an unknown label $x(v) \in \{\pm 1\}$. For each pair (u, v), we observe $x(u)x(v) + \sigma W_{uv}$, where $\sigma > 0$ and $W_{uv} \sim \mathcal{N}(0, 1)$. Now suppose that the vertices are generated according to a Poisson point process, and we observe $x(u)x(v) + \sigma W_{uv}$ only for mutually visible vertices, which models a signal recovery problem with spatially limited observations. An open question is then whether our two-phase approach can be adapted to this synchronization problem.

Acknowledgements. J.G. was supported in part by NSF CCF-2154100. X.N. and E.W. were supported in part by NSF ECCS-2030251 and CMMI-2024774.

References

[1] E. Abbe. Community detection and stochastic block models: recent developments. *The Journal of Machine Learning Research*, 18(1):6446–6531, 2017.

[2] E. Abbe, F. Baccelli, and A. Sankararaman. Community detection on Euclidean random graphs. *Information and Inference: A Journal of the IMA*, 10(1):109–160, 2021.

[3] E. Abbe, E. Boix-Adsera, P. Ralli, and C. Sandon. Graph powering and spectral robustness. *SIAM Journal on Mathematics of Data Science*, 2(1):132–157, 2020.

[4] E. Abbe, J. Fan, K. Wang, and Y. Zhong. Entrywise eigenvector analysis of random matrices with low expected rank. *Annals of Statistics*, 48(3):1452, 2020.

[5] E. Abbe and C. Sandon. Community detection in general stochastic block models: Fundamental limits and efficient algorithms for recovery. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, pages 670–688. IEEE, 2015.

[6] E. Araya Valdivia and D. C. Yohann. Latent distance estimation for random geometric graphs. *Advances in Neural Information Processing Systems*, 32, 2019.

[7] K. Avrachenkov, A. Bobu, and M. Dreveton. Higher-order spectral clustering for geometric graphs. *Journal of Fourier Analysis and Applications*, 27(2):22, 2021.

[8] A. S. Bandeira, N. Boumal, and A. Singer. Tightness of the maximum likelihood semidefinite relaxation for angular synchronization. *Mathematical Programming*, 163:145–167, 2017.

[9] S. Boucheron, G. Lugosi, and P. Massart. *Concentration inequalities: A nonasymptotic theory of independence*. Oxford University Press, 2013.

[10] E. Chien, A. Tulino, and J. Llorca. Active learning in the Geometric Block Model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3641–3648, 2020.

[11] V. Cohen-Addad, F. Mallmann-Trenn, and D. Saulpic. Community recovery in the degree-heterogeneous stochastic block model. In *Conference on Learning Theory*, pages 1662–1692. PMLR, 2022.

[12] Q. Duchemin and Y. De Castro. Random geometric graph: Some recent developments and perspectives. *High Dimensional Probability IX: The Ethereal Volume*, pages 347–392, 2023.

[13] R. Eldan, D. Mikulincer, and H. Pieters. Community detection and percolation of information in a geometric setting. *Combinatorics, Probability and Computing*, 31(6):1048–1069, 2022.

[14] S. Galhotra, A. Mazumdar, S. Pal, and B. Saha. The geometric block model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[15] S. Galhotra, A. Mazumdar, S. Pal, and B. Saha. Community recovery in the geometric block model. *arXiv preprint arXiv:2206.11303*, 2022.

[16] A. Giridhar and P. R. Kumar. Distributed clock synchronization over wireless networks: Algorithms and analysis. In *Proceedings of the 45th IEEE Conference on Decision and Control*, pages 4915–4920. IEEE, 2006.

[17] B. Hajek, Y. Wu, and J. Xu. Achieving exact cluster recovery threshold via semidefinite programming. *IEEE Transactions on Information Theory*, 62(5):2788–2797, 2016.

[18] M. S. Handcock, A. E. Raftery, and J. M. Tantrum. Model-based clustering for social networks. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 170(2):301–354, 2007.

[19] P. D. Hoff, A. E. Raftery, and M. S. Handcock. Latent space approaches to social network analysis. *Journal of the American Statistical Association*, 97(460):1090–1098, 2002.

[20] P. W. Holland, K. B. Laskey, and S. Leinhardt. Stochastic blockmodels: First steps. *Social Networks*, 5(2):109–137, 1983.

[21] A. Ivić, E. Krätzel, M. Kühleitner, and W. Nowak. Lattice points in large regions and related arithmetic functions: recent developments in a very classic topic. *Elementare und analytische Zahlentheorie, Franz Steiner*, pages 89–128, 2006.

[22] A. Javanmard, A. Montanari, and F. Ricci-Tersenghi. Phase transitions in semidefinite relaxations. *Proceedings of the National Academy of Sciences*, 113(16):E2218–E2223, 2016.

[23] J. F. C. Kingman. *Poisson Processes*, volume 3. Clarendon Press, 1992.

[24] S. Li and T. Schramm. Spectral clustering in the Gaussian mixture block model. *arXiv preprint arXiv:2305.00979*, 2023.

[25] E. Mossel, J. Neeman, and A. Sly. Consistency thresholds for the planted bisection model. In *Proceedings of the Forty-Seventh Annual ACM Symposium on Theory of Computing*, pages 69–75, 2015.

[26] S. Péché and V. Perchet. Robustness of community detection to random geometric perturbations. *Advances in Neural Information Processing Systems*, 33:17827–17837, 2020.

[27] A. Rapoport. Spread of information through a population with socio-structural bias: I. Assumption of transitivity. *The Bulletin of Mathematical Biophysics*, 15:523–533, 1953.

[28] D. M. Rosen, L. Carlone, A. S. Bandeira, and J. J. Leonard. A certifiably correct algorithm for synchronization over the special Euclidean group. In *Algorithmic Foundations of Robotics XII: Proceedings of the Twelfth Workshop on the Algorithmic Foundations of Robotics*, pages 64–79. Springer, 2020.

[29] A. Sankararaman and F. Baccelli. Community detection on Euclidean random graphs. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2181–2200. SIAM, 2018.

[30] A. Singer. Angular synchronization by eigenvectors and semidefinite programming. *Applied and Computational Harmonic Analysis*, 30(1):20–36, 2011.