Projected Tensor Power Method for Hypergraph Community Recovery

Jinxin Wang [1] **Yuen-Man Pun** [2] **Xiaolu Wang** [1] **Peng Wang** [3] **Anthony Man-Cho So** [1]

Abstract

This paper investigates the problem of exact community recovery in the symmetric d-uniform ($d \geq 2$) hypergraph stochastic block model (d-HSBM). In this model, a d-uniform hypergraph with n nodes is generated by first partitioning the n nodes into $K \geq 2$ equal-sized disjoint communities and then generating hyperedges with a probability that depends on the community memberships of d nodes. Despite the non-convex and discrete nature of the maximum likelihood estimation problem, we develop a simple yet efficient iterative method, called the *projected tensor power method*, to tackle it. As long as the initialization satisfies a partial recovery condition in the logarithmic degree regime of the problem, we show that our proposed method can exactly recover the hidden community structure down to the information-theoretic limit with high probability. Moreover, our proposed method exhibits a competitive time complexity of $\mathcal{O}(n \log^2 n / \log\log n)$ when the aforementioned initialization condition is met. We also conduct numerical experiments to validate our theoretical findings.

1. Introduction

Community detection (also known as graph clustering) is a fundamental task in various scientific and engineering fields ranging from data mining (Cabreros et al., 2016; Shi & Malik, 2000) to network analysis (Girvan & Newman, 2002). One celebrated and perhaps the simplest probabilistic model for generating random graphs with community structure is the stochastic block model (SBM) (Holland et al., 1983), which tends to exhibit more edge connections in communities and fewer edge connections across communities. Over the past decades, SBM has served as an important benchmark for validating and comparing various statistical theories and computational methods for community detection under different settings; see Abbe (2017) for a recent comprehensive survey of SBM. Despite the great success of SBM achieved on graph data, pairwise interactions represented by SBM are inadequate for modeling complex relational information in many real-world applications. For example, in social/academic networks, many cooperative relations like chat groups and co-author lists may consist of more than two people. Other applications involving such kind of high-order relations include congress voting networks (Lee et al., 2017), molecular interaction networks (Michoel & Nachtergaele, 2012), as well as high-order graph matching (Duchenne et al., 2011). Hence, it is natural and of keen interest to study an analogous model of SBM for capturing the aforementioned high-order relations.

In this work, to capture high-order interactions among multiple objects, we focus on the symmetric d-uniform hypergraph stochastic block model (d-HSBM) (Ghoshdastidar & Dukkipati, 2014; 2017; Kim et al., 2018; Chien et al., 2018; Ahn et al., 2018; Ke et al., 2019; Cole & Zhu, 2020; Zhang & Tan, 2022)—a natural extension of the SBM—and study the problem of exact community recovery. Specifically, in the d-HSBM, n nodes are partitioned into $K \geq 2$ unknown equal-sized non-overlapping communities, and each subset of nodes with cardinality d independently forms an order-d hyperedge with probability p if these d nodes are in the same community and with probability q otherwise. The goal is to identify the underlying community structure exactly based on a realization of such a random hypergraph. In the logarithmic degree regime of the d-HSBM, i.e., $p = \alpha \log n / n^{d-1}$ and $q = \beta \log n / n^{d-1}$ for some $\alpha > \beta > 0$, it has been recently established in Kim et al. (2017; 2018); Zhang & Tan (2022) that there exists a sharp phase transition around a threshold: It is possible to exactly identify the underlying communities with high probability if $\frac{(\sqrt{\alpha}-\sqrt{\beta})^2}{K^{d-1}(d-1)!} > 1$ and is impossible to recover the communities with non-vanishing probability if $\frac{(\sqrt{\alpha}-\sqrt{\beta})^2}{K^{d-1}(d-1)!} < 1$. On top of this breakthrough, a natural question arises: Can we design a computationally tractable algorithm that achieves exact recovery down to the aforementioned information-theoretic limit? In the past few decades, many computa-

[1]Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Hong Kong [2]CIICADA Lab, School of Engineering, Australian National University, Canberra [3]Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor. Correspondence to: Anthony Man-Cho So <manchoso@se.cuhk.edu.hk>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

tional methods have been developed in addressing this question, such as spectral clustering methods (Ghoshdastidar & Dukkipati, 2015; 2017; Chien et al., 2018; Ahn et al., 2018; Zhang & Tan, 2022), semidefinite programming-based methods (Kim et al., 2018; Gaudio & Joshi, 2022), and tensor decomposition-based methods (Ke et al., 2019; Han et al., 2022). However, some of these methods lack theoretical guarantees for exact recovery with high probability at the information-theoretic limit. Moreover, most of these methods have a time complexity of at least $\mathcal{O}(n^2)$, which is less favorable in contemporary large-scale problems.

In the symmetric d-HSBM with $K \geq 2$ communities, the maximum likelihood (ML) estimation problem takes the form

$$\max_{\boldsymbol{H} \in \mathbb{R}^{n \times K}} \left\{ \left\langle \mathcal{A}, \boldsymbol{H}^{\otimes d} \right\rangle : \ \boldsymbol{H} \in \mathcal{H} \right\}. \tag{MLE}$$

Here, $\mathcal{A}$ is the adjacency tensor of the observed hypergraph (see its definition in (8)), $\boldsymbol{H}^{\otimes d}$ is the outer product of the matrix $\boldsymbol{H}$ (see its definition in (2)),

$$\mathcal{H} = \left\{ \boldsymbol{H} \in \{0,1\}^{n \times K} : \boldsymbol{H}\mathbf{1}_K = \mathbf{1}_n, \boldsymbol{H}^\top \mathbf{1}_n = m\mathbf{1}_K \right\}$$

is the discrete feasible set characterizing the possible community assignment of n nodes into K clusters, $\mathbf{1}_n$ (resp. $\mathbf{1}_K$) is the all-one vector of dimension n (resp. K), and $m = n/K$ is the number of nodes in each community. It is known that an ML estimator can achieve exact recovery with high probability down to the information-theoretic limit; see, e.g., Kim et al. (2018, Proposition 1; Theorem 1). Although solving the non-convex problem (MLE) is NP-hard in the worst case, recent advances in different applications of non-convex optimization, including phase retrieval (Candes et al., 2015; Chen et al., 2019), low-rank matrix recovery (Chi et al., 2019; Li et al., 2020), low-rank tensor decomposition (Richard & Montanari, 2014; Huang et al., 2022; Han et al., 2022), phase/group synchronization (Liu et al., 2017; Zhong & Boumal, 2018; Zhu et al., 2021; Ling, 2022), community detection (Wang et al., 2021a;b; 2020; 2022), and graph matching (Araya et al., 2022), suggest that it could be possible to develop some simple iterative method that solves problem (MLE) down to the information-theoretic limit. In this work, we propose a simple and scalable method, called *projected tensor power method* (PTPM), to tackle the discrete optimization problem (MLE) and establish its exact recovery guarantee down to the information-theoretic limit. In contrast to the existing works on *generalized power methods* for solving non-convex optimization problems where theoretical analyses are performed upon a *quadratic* objective (Liu et al., 2017; Zhong & Boumal, 2018; Zhu et al., 2021; Ling, 2022; Wang et al., 2021a;b; 2022; Araya et al., 2022), it is worth highlighting that this paper establishes global optimality and fast convergence rate of PTPM for a *polynomial* optimization problem. Thus, our work expands the repertoire of globally solvable non-convex optimization problems by generalized power methods.

1.1. Related Literature

There are several different goals for community detection over hypergraphs. One is *exact recovery* (also known as *strong consistency*), which is to identify the true underlying community structures with high probability based on a realization of a random hypergraph. Regarding the associated information-theoretic limit, in the logarithmic degree regime of the symmetric d-HSBM with $K = 2$, it was proved in Kim et al. (2018, Theorem 1) that exact recovery is achievable if and only if $\frac{(\sqrt{\alpha}-\sqrt{\beta})^2}{2^{d-1}(d-1)!} > 1$. Later, the exact recovery threshold $\frac{(\sqrt{\alpha}-\sqrt{\beta})^2}{K^{d-1}(d-1)!} > 1$ was extended to scenarios with $K \geq 2$ in Zhang & Tan (2022, Theorem 2). Another two goals are *almost exact recovery* (also known as *weak consistency*) and *partial recovery*, respectively. The former aims at identifying the true communities with a vanishing fraction of misclassified vertices, while the latter merely aims at correctly identifying a constant fraction of vertices; see Abbe (2017) for further details of these goals.

Apart from the information-theoretic limits mentioned above, many efforts have been made to develop algorithms for the exact recovery of the d-HSBM over the past few years.

Spectral methods. One popular method is spectral clustering, which generally involves three steps: (i) constructing a data matrix, (ii) performing eigendecomposition of the data matrix, and (iii) applying the k-means clustering algorithm to the eigenvectors. For example, in Ghoshdastidar & Dukkipati (2014; 2015; 2017), the authors first constructed a weight matrix based on either the hypergraph Laplacian or the tensor unfolding, and then applied the k-means clustering algorithms to the leading K eigenvectors of the obtained weight matrix. However, the theoretical results therein require the hypergraph to be dense, and hence, the condition for exact recovery is not optimal. Moreover, spectral clustering methods generally require polynomial running time. Recently, Gaudio & Joshi (2022) proved that the spectral method based on the weight matrix (also called the similarity matrix) of a constructed weighted graph can already achieve exact recovery without performing k-means clustering. Although the approach of projecting the original hypergraph into a weighted graph allows one to directly apply existing methods for graph networks, such a method is not optimal for exact recovery due to the loss of information (Ke et al., 2019) when constructing the similarity matrix from the observed adjacency tensor.

Semidefinite programming (SDP)-based methods. The SDP-based methods with $K = 2$ have been considered in Kim et al. (2018); Gaudio & Joshi (2022). However, the

conditions for achieving exact recovery are not optimal as the SDP-based methods only utilize the similarity matrix instead of the original adjacency tensor. Moreover, solving large-scale SDP is usually computationally heavy.

Two-stage methods. It was suggested in Abbe (2017); Kim et al. (2018) that a local refinement method together with an initialization satisfying partial recovery can possibly lead to exact recovery. Building on such a high-level idea, there are some follow-up works for tackling the problem via two-stage methods. Chien et al. (2019) considered a two-stage algorithm that starts from a weakly consistent initialization and then refines it by a local maximum likelihood estimation method for each node separately. It recovers the communities exactly with high probability down to information-theoretic threshold in $\mathcal{O}(n^3 \log n)$ time; see Chien et al. (2019, Section IV.C). Zhang & Tan (2022) also proposed a two-stage algorithm, in which a hypergraph spectral clustering step in the first stage ensures weak consistency and a follow-up local refinement stage guarantees exact recovery. Their proposed method achieves exact recovery at the information-theoretic limit in polynomial time. In addition, Ke et al. (2019) considered a degree-corrected hypergraph SBM and developed a two-step approach including a regularized high-order orthogonal iteration algorithm and the k-means clustering starting from a suitable initialization. Their proposed method generalizes the celebrated tensor power method for tensor principal component analysis (PCA) (Richard & Montanari, 2014; Huang et al., 2022). The setting therein is more general as it can deal with degree heterogeneity. Nevertheless, their theoretical results are not optimal when applied to the symmetric d-HSBM (Ke et al., 2019, Corollary 1). Other interesting two-stage methods include a high-order extension of the Lloyd algorithm for clustering under the general tensor block model (Han et al., 2022).

We summarize the above related works in Table 1.

Table 1. Comparison of recovery conditions and time complexities of the discussed methods for exact recovery in the d-HSBM ($K \geq 2$).

References	Optimal	Complexities
Ghoshdastidar & Dukkipati (2015)	✗	Polynomial
Gaudio & Joshi (2022) (Spectral method)	✗	Polynomial
Kim et al. (2018), Gaudio & Joshi (2022) (SDP)	✗	Polynomial
Chien et al. (2019)	✔	$\mathcal{O}(n^3 \log n)$
Zhang & Tan (2022)	✔	Polynomial
Ours	✔	$\mathcal{O}\left(\frac{n \log^2 n}{\log \log n}\right)$

1.2. Our Contributions

In this work, we tackle the non-convex discrete optimization problem (MLE) via a simple and scalable projected tensor power method. Specifically, given an initialization satisfying a certain partial recovery condition, we refine the estimate via projected tensor power iteration successively. In the logarithmic degree regime of the d-HSBM, we prove that PTPM can exactly recover the underlying community labels within $\mathcal{O}(\log n / \log \log n)$ iterations with high probability at the information-theoretic limit. Moreover, each iteration requires only $\mathcal{O}(n \log n)$ time. Therefore, the overall time complexity of PTPM given a qualified initialization would be $\mathcal{O}(n \log^2 n / \log \log n)$, which is competitive to the state-of-the-art methods. Besides the simplicity of PTPM, we remark that it only requires an initialization satisfying the partial recovery condition, which is much milder than the conditions imposed in the majority of existing two-stage methods; see, e.g., Chien et al. (2019) and (Zhang & Tan, 2022). As a result, we provide an affirmative answer to the question raised in Abbe (2017); Kim et al. (2018) that whether a local refinement method together with an initialization only satisfying the partial recovery can lead to exact recovery. As a byproduct of our analysis, we leverage the Kahn-Szemerédi argument and provide a novel concentration inequality for dealing with the extremely sparse adjacency tensor (see Lemma 4.3). Our bound is much tighter than existing concentration bounds for tensors (see, e.g., Zhou & Zhu (2021, Theorem 2.3; Remark 2.2; Lemma 6.1)) and thus can be of independent interest to applications with high-order relations.

Our work also contributes to the emerging provable non-convex optimization area. In particular, despite the non-convex and discrete nature of problem (MLE), our proposed PTPM solves the problem efficiently and optimally given a carefully designed initialization. Moreover, prior to our work, the analyses of the generalized power method are performed on non-convex optimization problems with quadratic objectives; see, e.g., Boumal (2016); Chen & Candès (2018); Zhong & Boumal (2018); Zhu et al. (2021); Wang et al. (2021b); Araya et al. (2022). By sharp contrast, our analysis is performed on an optimization problem with a *polynomial* objective, which expands the repertoire of globally solvable non-convex optimization problems.

The rest of this paper is organized as follows. In Section 2, we review some basic concepts in tensor algebra that will be used throughout the paper. Next, we introduce the d-HSBM and the proposed PTPM in Section 3 and present our main results in Section 4. We then report some numerical results in Section 5 and conclude in Section 6.

2. Notation and Preliminaries

We use bold uppercase letters $\boldsymbol{A}, \boldsymbol{B}, \ldots$ to denote matrices and $\boldsymbol{A}_j$ and A_{ij} to denote the j-th column and the (i,j)-th entry of $\boldsymbol{A}$, respectively. We use calligraphic letters $\mathcal{A}, \mathcal{B}, \ldots$ to denote tensors of order three or higher. For instance, an order-d tensor $\mathcal{A} \in \mathbb{R}^{p_1 \times \cdots \times p_d}$ represents a d-way array of size $p_1 \times p_2 \times \cdots \times p_d$. The $(i_1, \ldots, i_d)$-th entry of a tensor $\mathcal{A}$ is denoted by $\mathcal{A}_{i_1,\ldots,i_d}$. If $p_1 = p_2 = \cdots = p_d = n$, we simply write $\mathcal{A} \in T^d(\mathbb{R}^n)$. We say a tensor $\mathcal{A} \in T^d(\mathbb{R}^n)$ is *symmetric* if $\mathcal{A}_{i_1,\ldots,i_d} = \mathcal{A}_{j_1,\ldots,j_d}$ whenever $(j_1, \ldots, j_d)$ is a permutation of $(i_1, \ldots, i_d)$ and in this case we write $\mathcal{A} \in S^d(\mathbb{R}^n)$. The *inner product* of two tensors $\mathcal{X}, \mathcal{Y}$ with the same dimension is defined as $\langle \mathcal{X}, \mathcal{Y} \rangle = \sum_{i_1,\ldots,i_d} \mathcal{X}_{i_1,\ldots,i_d} \mathcal{Y}_{i_1,\ldots,i_d}$. The *Frobenius norm* of a tensor $\mathcal{X}$ is defined as $\|\mathcal{X}\|_F = \langle \mathcal{X}, \mathcal{X} \rangle^{1/2}$. The *multilinear multiplication* of a tensor $\mathcal{X} \in \mathbb{R}^{r_1 \times \cdots \times r_d}$ by matrices $\boldsymbol{U}_k \in \mathbb{R}^{p_k \times r_k}$ for $k = 1, \ldots, d$ is defined as

$$
\begin{aligned}
&(\mathcal{X} \times_1 \mathbf{U}_1 \times_2 \cdots \times_d \mathbf{U}_d)_{i_1,\ldots,i_d} \\
&= \sum_{j_1=1}^{r_1} \cdots \sum_{j_d=1}^{r_d} \mathcal{X}_{j_1,\ldots,j_d} (\mathbf{U}_1)_{i_1 j_1} \cdots (\mathbf{U}_d)_{i_d j_d},
\end{aligned}
$$

which outputs an order-d $(p_1, \ldots, p_d)$-dimensional tensor. For a vector $\boldsymbol{x} \in \mathbb{R}^n$, the outer product $\boldsymbol{x}^{\otimes d}$ is a tensor $\mathcal{X} \in S^d(\mathbb{R}^n)$ with

$$
\mathcal{X}_{i_1,\ldots,i_d} = x_{i_1} \times \cdots \times x_{i_d}. \tag{1}
$$

For a matrix $\boldsymbol{H} \in \mathbb{R}^{n \times K}$, the outer product $\boldsymbol{H}^{\otimes d}$ is a tensor $\mathcal{X} \in S^d(\mathbb{R}^n)$ (Kolda & Bader, 2009, Eq. (3.4)) with

$$
\mathcal{X}_{i_1,\ldots,i_d} = \sum_{k=1}^{K} H_{i_1 k} \times \cdots \times H_{i_d k}. \tag{2}
$$

Equivalently, $\mathcal{X} = \sum_{k=1}^{K} (\boldsymbol{H}_k)^{\otimes d}$. For a tensor $\mathcal{A} \in S^d(\mathbb{R}^n)$, the *multilinear operation* $\mathcal{A}\left[\boldsymbol{H}^{\otimes(d-1)}\right]$ (see Huang et al. (2022, Eq. (2); Section 1.2) and Richard & Montanari (2014)) outputs an $n \times K$ matrix with its (i,k)-th entry given by

$$
\begin{aligned}
&\left(\mathcal{A}\left[\boldsymbol{H}^{\otimes(d-1)}\right]\right)_{ik} \\
&= \sum_{1 \le i_2,\ldots,i_d \le n} (\mathcal{A}_{i,i_2,\ldots,i_d} \times H_{i_2 k} \times \cdots \times H_{i_d k}) \\
&= \left\langle \mathcal{A}, \boldsymbol{e}_i \otimes (\boldsymbol{H}_k)^{\otimes(d-1)} \right\rangle,
\end{aligned} \tag{3}
$$

where $\boldsymbol{e}_i \in \mathbb{R}^n$ stands for the zero vector except for the i-th entry being one and the outer product operation $\boldsymbol{e}_i \otimes (\boldsymbol{H}_k)^{\otimes(d-1)}$ is defined as

$$
\begin{aligned}
&\left(\boldsymbol{e}_i \otimes (\boldsymbol{H}_k)^{\otimes(d-1)}\right)_{j,i_2,\ldots,i_d} \\
&= (\boldsymbol{e}_i)_j \cdot \left((\boldsymbol{H}_k)^{\otimes(d-1)}\right)_{i_2,\ldots,i_d}
\end{aligned} \tag{4}
$$

for $j = 1, \ldots, n$ and $1 \le i_2, \ldots, i_d \le n$. For a tensor $\mathcal{A} \in S^d(\mathbb{R}^n)$, the *mode-1 matricization* of $\mathcal{A}$, denoted by $\mathcal{M}(\mathcal{A}) \in \mathbb{R}^{n \times n^{d-1}}$, is defined as

$$
(\mathcal{M}(\mathcal{A}))_{ij} = \mathcal{A}_{i,i_2,\ldots,i_d} \text{ with } j = 1 + \sum_{k=2}^{d} (i_k - 1) n^{k-2}; \tag{5}
$$

see, e.g., Kolda & Bader (2009, Section 2.4) and Han et al. (2022, Section 2.1). For a matrix $\boldsymbol{H} \in \mathbb{R}^{n \times K}$, we use $\boldsymbol{H}^{\odot(d-1)} \in \mathbb{R}^{n^{d-1} \times K}$ to represent the *Khatri-Rao product* (also known as the column-wise Kronecker product) of $\boldsymbol{H}$, which is defined as

$$
\left(\boldsymbol{H}^{\odot(d-1)}\right)_{jk} = H_{i_2 k} \times H_{i_3 k} \times \cdots \times H_{i_d k} \tag{6}
$$

with $j = 1 + \sum_{k=2}^{d} (i_k - 1) n^{k-2}$. Combining (5) and (6) yields a useful fact

$$
\mathcal{A}\left[\boldsymbol{H}^{\otimes(d-1)}\right] = \mathcal{M}(\mathcal{A}) \left(\boldsymbol{H}^{\odot(d-1)}\right). \tag{7}
$$

The readers are referred to Kolda & Bader (2009); Sidiropoulos et al. (2017); Cichocki et al. (2016) for a more detailed introduction to tensor algebra. In addition, we use Π_K to denote the collections of all $K \times K$ permutation matrices and use $\mathbf{Bern}(p)$ to denote the Bernoulli random variable with parameter p. Given a positive integer n, we denote by $[n]$ the set $\{1, \ldots, n\}$. Given a discrete set S, we denote by $|S|$ the cardinality of S. If two random variables X and Y are equal in distribution, we write $X \overset{\mathrm{d}}{=} Y$.

3. Projected Tensor Power Method

We formally state the symmetric d-HSBM in the following definition.

Definition 3.1 (Symmetric d-HSBM)**.** Let $n \ge 2$ be the number of nodes, $K \ge 2$ be the number of communities, and $p, q \in (0, 1]$ be the probability parameters of generating hyperedges. Furthermore, let $\boldsymbol{H}^* \in \mathcal{H}$ represent a hidden partition of n nodes into K equal-sized disjoint communities. A random hypergraph is generated according to the symmetric d-HSBM with parameters (n, d, K, p, q) and $\boldsymbol{H}^*$ if the adjacency tensor $\mathcal{A} \in T^d(\mathbb{R}^n)$ of such a hypergraph is symmetric and the elements

$$
\{\mathcal{A}_{i_1,i_2,\ldots,i_d}\}_{1 \le i_1 < i_2 < \cdots < i_d \le n}
$$

are generated independently by

$$
\mathcal{A}_{i_1,i_2,\ldots,i_d} \sim \begin{cases} \mathbf{Bern}(p), & \text{if } \left(\boldsymbol{h}^*_{i_1} \circ \boldsymbol{h}^*_{i_2} \circ \cdots \circ \boldsymbol{h}^*_{i_d}\right) \mathbf{1}_K = 1, \\ \mathbf{Bern}(q), & \text{if } \left(\boldsymbol{h}^*_{i_1} \circ \boldsymbol{h}^*_{i_2} \circ \cdots \circ \boldsymbol{h}^*_{i_d}\right) \mathbf{1}_K = 0, \end{cases} \tag{8}
$$

where $\boldsymbol{h}^*_i$ is the i-th row of $\boldsymbol{H}^*$ and "$\circ$" denotes the Hadamard (i.e., element-wise) product. In addition, since each d-uniform hyperedge consists of exactly d nodes, the

diagonal elements of $\mathcal{A}$ are automatically defined to be 0, i.e., $\mathcal{A}_{i_1,\ldots,i_d} = 0$ if some indices among $i_1, \ldots, i_d$ are identical.

Given a realization of a random hypergraph generated by the symmetric d-HSBM, our goal is to exactly recover the underlying communities (i.e., output $\boldsymbol{H}^*\boldsymbol{Q}$ for some $\boldsymbol{Q} \in \Pi_K$) with high probability via a simple iterative procedure. In view of the fact that problem (MLE) is reminiscent of the problem formulation of tensor PCA, a natural attempt is to apply a variant of the tensor power iteration (Richard & Montanari, 2014; Huang et al., 2022). Although many variants of the classic power method have been developed for solving PCA problems with different structural constraints; see, e.g., Journée et al. (2010); Deshpande et al. (2014); Chen & Candès (2018); Zhong & Boumal (2018); Zhu et al. (2021); Wang et al. (2021b), a new variant of the tensor power iteration method has to be developed for problem (MLE) as it involves a polynomial objective with a matrix variable and binary constraints.

Our approach for tackling problem (MLE) is to iteratively apply a tensor power step and a projection step that ensures feasibility of the iterate. Specifically, the projected tensor power iteration for tackling problem (MLE) takes the form

$$\boldsymbol{H}^{t+1} \in \mathcal{T}\left(\mathcal{A}\left[\left(\boldsymbol{H}^t\right)^{\otimes(d-1)}\right]\right) \text{ for } t \geq 1, \tag{9}$$

where $\mathcal{T} : \mathbb{R}^{n\times K} \rightrightarrows \mathbb{R}^{n\times K}$ represents the projection operator onto $\mathcal{H}$; i.e., for any $\boldsymbol{C} \in \mathbb{R}^{n\times K}$,

$$\mathcal{T}(\boldsymbol{C}) \coloneqq \arg\min\left\{\|\boldsymbol{H} - \boldsymbol{C}\|_F : \boldsymbol{H} \in \mathcal{H}\right\}, \tag{10}$$

and the operator $\mathcal{A}\left[\left(\boldsymbol{H}^t\right)^{\otimes(d-1)}\right]$ is defined in (3).

As an iterative method, the global convergence of PTPM relies on a proper initialization. Specifically, the initial point $\boldsymbol{H}^0$ needs to satisfy the partial recovery condition (see, e.g., Dumitriu et al. (2021)):

$$\boldsymbol{H}^0 \in \mathbb{M}_{n,K} \text{ s.t. } \min_{\boldsymbol{Q}\in\Pi_K} \|\boldsymbol{H}^0 - \boldsymbol{H}^*\boldsymbol{Q}\|_F \leq \theta\sqrt{n}, \tag{11}$$

where $\mathbb{M}_{n,K}$ represents the set of such $n \times K$ matrices that each row is all zero except for one element being 1, and θ is a constant that will be specified later. We remark that condition (11) can be satisfied by a host of existing initialization methods. For example, Chien et al. (2019) and Zhang & Tan (2022) proposed spectral initialization methods, which can obtain an initial point $\boldsymbol{H}^0$ satisfying the almost exact recovery condition (see Abbe (2017, Definition 4)). The almost exact recovery condition is much more stringent than (11), and thus these initialization strategies automatically satisfy the partial recovery requirement in (11).

Our proposed PTPM for tackling problem (MLE) is summarized in Algorithm 1. With a qualified initialization $\boldsymbol{H}^0$, it first projects $\boldsymbol{H}^0$ onto $\mathcal{H}$ to guarantee feasibility. Then, it repeatedly refines the estimate by performing N-step projected tensor power iterations.

Algorithm 1 Projected Tensor Power Method for Solving Problem (MLE)

Input: adjacency tensor $\mathcal{A}$, positive integer N
Initialize an $\boldsymbol{H}^0$ satisfying (11)
set $\boldsymbol{H}^1 \leftarrow \mathcal{T}(\boldsymbol{H}^0)$
for $t = 1, 2, \ldots, N$ **do**
  set $\boldsymbol{H}^{t+1} \in \mathcal{T}\left(\mathcal{A}\left[\left(\boldsymbol{H}^t\right)^{\otimes(d-1)}\right]\right)$
end for
Output $\boldsymbol{H}^{N+1}$

Remark 3.2. Although Algorithm 1 is designed for recovering the community of a symmetric d-uniform hypergraph SBM, it is applicable to the *non-uniform* hypergraph stochastic block model: Suppose that the observed hypergraph is non-uniform with the size of the hyperedges ranging from 2 to d_0. We can introduce a set of dummy nodes indexed by -3 to $-d_0$ to reformulate the non-uniform hypergraph as a uniform hypergraph. Specifically, for any hyperedge consisting of nodes $\{i_1, i_2, \ldots, i_d\}$ with $d < d_0$, we add the dummy nodes to the hyperedge such that it is of d_0 nodes $\{i_1, i_2, \ldots, i_d, -(d+1), \ldots, -d_0\}$. Then, given a d_0-uniform hypergraph, PTPM can be applied to recover the hidden community structure.

Remark 3.3. It is less straightforward for PTPM to recover the community structure from an *asymmetric* hypergraph. Specifically, the asymmetry induces an extra set of assignment matrix variables in the ML estimation problem corresponding to each mode of the observed adjacency tensor. Therefore, one may have to include an extra inner loop to update the set of assignment matrices for each mode at every projected tensor power iteration when applying the idea of PTPM. A related algorithm is the classic high-order orthogonal iteration (HOOI) (see, e.g., Kolda & Bader (2009, Figure 4.4)), but it does not involve a projection step.

4. Main Results

We first present the main theorem of this paper, which states the exact recovery of PTPM in Algorithm 1 down to the information-theoretic limit and the explicit iteration/time complexity of the algorithm.

Theorem 4.1. *Let $\mathcal{A} \in S^d(\mathbb{R}^n)$ be an observed adjacency tensor of the random hypergraph generated according to the symmetric d-HSBM in Definition 3.1 with parameters (n, d, K, p, q) and a planted partition $\boldsymbol{H}^* \in \mathcal{H}$. Suppose that $p = \alpha \log n/n^{d-1}, q = \beta \log n/n^{d-1}$ with $\frac{(\sqrt{\alpha}-\sqrt{\beta})^2}{K^{d-1}(d-1)!} > 1$ and n is sufficiently large. Then, there exists a constant $\gamma > 0$, whose value depends only on α, β, d, and K, such that the following statement holds with prob-*

ability at least $1 - n^{-\Omega(1)}$: If the initial point satisfies the partial recovery condition in (11) with

$$\theta = \frac{1}{4} \min \left\{ \frac{1}{\sqrt{K(d-1)}}, \frac{\gamma K^{d-3/2}}{16(d-1)(\alpha-\beta)} \right\}, \quad (12)$$

then Algorithm 1 outputs a true partition in $\left(\lceil 2 \log\log n \rceil + \left\lceil \frac{2\log n}{\log\log n} \right\rceil + 2 \right)$ projected tensor power iterations. Moreover, Algorithm 1 outputs a true partition in $\mathcal{O}(n \log^2 n / \log\log n)$ time.

Let us give the proof outline here before proceeding. Given an initialization $\boldsymbol{H}^0$ satisfying the partial recovery condition (11), we show that its projection onto $\mathcal{H}$, namely $\boldsymbol{H}^1$, still lies in a neighborhood of the ground truth using certain Lipschitz-type inequality (see (57) and (60) for details). Then, by analyzing the effect of the multilinear operation $\mathcal{A}[\cdot]$ and the projection $\mathcal{T}(\cdot)$, we show that each projected tensor power iteration possesses a local contraction property. In other words, the iterate $\boldsymbol{H}^t$ gets closer to the ground truth $\boldsymbol{H}^*\boldsymbol{Q}$ at every iteration (since $\boldsymbol{H}^*\boldsymbol{Q}$ represents the same partition as $\boldsymbol{H}^*$ for any permutation matrix $\boldsymbol{Q}$). Specifically, we show that the distance between the iterate $\boldsymbol{H}^t$ and the ground truth $\boldsymbol{H}^*\boldsymbol{Q}$ shrinks by a factor of $\frac{1}{2}$ at each iteration (see (58)). Moreover, after $N_1 = \lceil 2\log\log n \rceil + 1$ iterations, the iterate gets so close to the ground truth that the contraction factor further reduces to $\mathcal{O}\left(\frac{1}{\sqrt{\log n}}\right)$ (see (61)). Then, after additional $N_2 = \left\lceil \frac{2\log n}{\log\log n} \right\rceil$ iterations, we can upper bound the distance between the iterate $\boldsymbol{H}^{N_1+N_2}$ and the ground truth $\boldsymbol{H}^*\boldsymbol{Q}$ strictly by $\sqrt{2}$ (see (62)). Due to the discrete nature of the feasible set $\mathcal{H}$, this means that the iterate $\boldsymbol{H}^{N_1+N_2}$ has achieved exact recovery (i.e., $\boldsymbol{H}^{N_1+N_2} = \boldsymbol{H}^*\boldsymbol{Q}$). Multiplying the obtained iteration complexity by the time complexity of each projected tensor power iteration, we can derive the total time complexity of PTPM in Theorem 4.1.

In the remaining part of this section, we provide the proof of Theorem 4.1. We break down the analysis of each projected tensor power iteration by studying the effect of the multilinear operation $\mathcal{A}[\cdot]$ (in Lemmas 4.2, 4.3, 4.6, and 4.7) and the projection $\mathcal{T}(\cdot)$ (in Lemma 4.5), respectively, and then establish a local contraction property of the projected tensor power iteration in Proposition 4.8. Based on the contraction property, we derive the iteration complexity of PTPM to achieve exact community recovery in Theorem 4.9 and also the time complexity in Theorem 4.1.

We start with characterizing the effect of the multilinear operation $\mathcal{A}[\boldsymbol{H}^{\otimes(d-1)}]$ in Lemma 4.2. To make the presentation more concise, we slightly change the definition of $\mathcal{A}$ in this lemma: For the diagonal elements with some of the indices $i_1, \ldots, i_d$ being identical, we have $\mathcal{A}_{i_1,\ldots,i_d} \sim \mathbf{Bern}(p)$ if the d nodes are in the same community and $\mathcal{A}_{i_1,\ldots,i_d} \sim \mathbf{Bern}(q)$ otherwise. In fact, the error term incurred by this modification is negligible, on which we will comment in Remark 4.4.

Lemma 4.2. *Suppose that $\varepsilon \in \left(0, 1/\sqrt{K(d-1)}\right)$ and $\boldsymbol{H} \in \mathcal{H}$ such that $\|\boldsymbol{H} - \boldsymbol{H}^*\boldsymbol{Q}\|_F \le \varepsilon\sqrt{n}$ for some $\boldsymbol{Q} \in \Pi_K$. Then, with probability at least $1 - n^{-10}$, we have*

$$\begin{aligned} &\left\| \mathcal{A}\left[\boldsymbol{H}^{\otimes(d-1)}\right] - \mathcal{A}\left[(\boldsymbol{H}^*\boldsymbol{Q})^{\otimes(d-1)}\right] \right\|_F \\ &\le \left(\frac{4(d-1)m^{d-2}\varepsilon n}{\sqrt{K}}(p-q) + C\sqrt{\log n} \right) \|\boldsymbol{H} - \boldsymbol{H}^*\boldsymbol{Q}\|_F \end{aligned} \quad (13)$$

with $C > 0$ being a constant.

The idea of proving Lemma 4.2 is to separately bound

$$\left\| (\mathbb{E}[\mathcal{A}]) \left[\boldsymbol{H}^{\otimes(d-1)}\right] - (\mathbb{E}[\mathcal{A}]) \left[(\boldsymbol{H}^*\boldsymbol{Q})^{\otimes(d-1)}\right] \right\|_F$$

and

$$\left\| \Delta \left[\boldsymbol{H}^{\otimes(d-1)}\right] - \Delta \left[(\boldsymbol{H}^*\boldsymbol{Q})^{\otimes(d-1)}\right] \right\|_F$$

with high probability, where $\Delta = \mathcal{A} - \mathbb{E}[\mathcal{A}]$ is the deviation of $\mathcal{A}$ from its expectation. On one hand, the former term can be computed via the definition of $\mathcal{A}$ and some algebraic inequalities; on the other hand, using (7), the latter term would yield

$$\left\| \mathcal{M}(\Delta) \left(\boldsymbol{H}_k^{\odot(d-1)} - (\boldsymbol{H}^*\boldsymbol{Q})_k^{\odot(d-1)} \right) \right\|_2 \text{ for } k \in [K]; \quad (14)$$

see (38). A natural attempt to bound (14) is to directly apply the inequalities (34) and the results in Zhou & Zhu (2021, Theorem 2.3; Remark 2.2; Lemma 6.1). Yet, this approach would yield a loose upper bound. As a remedy, we present a useful concentration bound of (14), as shown in Lemma 4.3. The trick is to leverage the Kahn-Szemerédi argument (Feige & Ofek, 2005), which has been applied to obtain bounds for the spectral norm of sparse binary random square matrices (Lei & Rinaldo, 2015) and tensors (Zhou & Zhu, 2021). The result is new and can be of interest to other applications with high-order relations.

Lemma 4.3. *Let $d \ge 2$ and $\frac{1}{s_1} \le \ell \le n$ for some constant $s_1 \ge 1$. Let $\boldsymbol{A} \in \mathbb{R}^{n \times n^{d-1}}$ be a random matrix whose (i,j)-th entry independently follows $\mathbf{Bern}(p_{ij})$. Set $\boldsymbol{P} = \mathbb{E}[\boldsymbol{A}]$ and assume that $\xi := n^{d-1} \cdot p_{\max} \ge c_0 \log n$ for $p_{\max} := \max_{i\in[n], j\in[n^{d-1}]} p_{ij}$ and some constant $c_0 > 0$. Given a vector $\boldsymbol{y} \in \mathbb{R}^{n^{d-1}}$ that has exactly $\ell \cdot n^{d-2}$ nonzero elements with half of them taking 1 and the others taking -1, then, for any $r > 0$, there exists a constant $C > 0$ such that*

$$\|(\boldsymbol{A} - \boldsymbol{P})\boldsymbol{y}\|_2 \le C\sqrt{\xi} \cdot \sqrt{\ell} \quad (15)$$

with probability at least $1 - n^{-r}$.

Remark 4.4. The results in Lemma 4.2 still hold when the diagonal entries of $\mathcal{A}$ are 0 (as defined in Definition 3.1). In fact, since the modification on the tensor $\mathcal{A}$ only takes place in the diagonal entries of the tensor, it is negligible when compared with the effect of $\mathbb{E}[\mathcal{A}]$ acting on the matrices $\boldsymbol{H}$ and $\boldsymbol{H}^*\boldsymbol{Q}$. We will leave a detailed discussion in the appendix.

Next, we shift our attention to the projection operator $\mathcal{T}$ and present its Lipschitz-like property in the following lemma (Wang et al., 2021a, Lemma 3).

Lemma 4.5. *Let $\delta > 0, \boldsymbol{C} \in \mathbb{R}^{n\times K}$ be arbitrary and $m = n/K$. If there exists a collection of index sets $\mathcal{I}_1, \dots, \mathcal{I}_K$ satisfying $\cup_{k=1}^K \mathcal{I}_k = [n], \mathcal{I}_k \cap \mathcal{I}_\ell = \emptyset$, and $|\mathcal{I}_k| = m$ such that $\boldsymbol{C}$ satisfies*

$$C_{ik} - C_{i\ell} \geq \delta \tag{16}$$

for all $i \in \mathcal{I}_k$ and $1 \leq k \neq \ell \leq K$. Then, for any $\boldsymbol{V} \in \mathcal{T}(\boldsymbol{C})$, $\boldsymbol{C}' \in \mathbb{R}^{n\times K}$ and $\boldsymbol{V}' \in \mathcal{T}(\boldsymbol{C}')$, we have

$$\|\boldsymbol{V} - \boldsymbol{V}'\|_F \leq \frac{2\|\boldsymbol{C} - \boldsymbol{C}'\|_F}{\delta}. \tag{17}$$

Considering $\boldsymbol{C} = \mathcal{A}\left[(\boldsymbol{H}^*)^{\otimes(d-1)}\right]$, the following two lemmas show that condition (16) in Lemma 4.5 can be satisfied with high probability. We start with a lemma addressing a binomial tail inequality below.

Lemma 4.6. *Let $m = n/K$ and $\alpha > \beta > 0$ be constants. Suppose that $\{W_i\}_{i=1}^{\binom{m-1}{d-1}}$ are i.i.d. $\mathbf{Bern}(\alpha \log n/n^{d-1})$ and $\{Z_i\}_{i=1}^{\binom{m}{d-1}}$ are i.i.d. $\mathbf{Bern}(\beta \log n/n^{d-1})$ that is independent of $\{W_i\}_{i=1}^{\binom{m-1}{d-1}}$. Then, for any $\gamma \in \mathbb{R}$, it holds that*

$$\Pr\left[\sum_{i=1}^{\binom{m-1}{d-1}} W_i - \sum_{i=1}^{\binom{m}{d-1}} Z_i \leq \frac{\gamma}{(d-1)!}\log n\right] \leq n^{-\frac{\binom{m-1}{d-1}(\sqrt{\alpha}-\sqrt{\beta})^2}{n^{d-1}} + \frac{\gamma\log(\alpha/\beta)}{2(d-1)!} + \frac{\left(\binom{m}{d-1}-\binom{m-1}{d-1}\right)(\sqrt{\alpha\beta}-\beta)}{n^{d-1}}}. \tag{18}$$

Lemma 4.6 can be proved by adapting the proof techniques in Abbe et al. (2020, Lemma 8). This turns out to be the key to proving the information-theoretic optimality of our algorithm. Recalling that $\boldsymbol{C} = \mathcal{A}\left[(\boldsymbol{H}^*)^{\otimes(d-1)}\right]$, by the definitions of the adjacency tensor $\mathcal{A}$ and the multilinear operator $\mathcal{A}[\cdot]$, one can compute

$$C_{ik} \overset{\mathrm{d}}{=} (d-1)! \cdot \sum_{i=1}^{\binom{m-1}{d-1}} W_i,\ i \in \mathcal{I}_k = \{i \in [n] : H^*_{ik} = 1\}$$

and

$$C_{i\ell} \overset{\mathrm{d}}{=} (d-1)! \cdot \sum_{i=1}^{\binom{m}{d-1}} Z_i,\ l \neq k,$$

where $\{W_i\}$ are i.i.d. $\mathbf{Bern}(\alpha \log n/n^{d-1})$, and $\{Z_i\}$ are i.i.d. $\mathbf{Bern}(\beta \log n/n^{d-1})$ and independent of $\{W_i\}$. Utilizing the result of Lemma 4.6, we then have the following lemma.

Lemma 4.7. *Let $\alpha > \beta > 0$ be constants. Denote $\boldsymbol{C} = \mathcal{A}\left[(\boldsymbol{H}^*)^{\otimes(d-1)}\right]$ and $\mathcal{I}_k = \{i \in [n] : H^*_{ik} = 1\}$ for all $k \in [K]$. If $\frac{(\sqrt{\alpha}-\sqrt{\beta})^2}{K^{d-1}(d-1)!} > 1$ and n is sufficiently large, then there exists a constant $\gamma > 0$, which depends only on α, β, d, and K, such that for all $i \in \mathcal{I}_k$ and $1 \leq k \neq \ell \leq K$,*

$$C_{ik} - C_{i\ell} \geq \gamma \log n \tag{19}$$

holds with probability at least $1 - n^{-\Omega(1)}$.

Collecting the results from all the lemmas above, we can now show that the projected tensor power iteration possesses a contraction property in a certain neighborhood of $\boldsymbol{H}^*\boldsymbol{Q}$ for some $\boldsymbol{Q} \in \Pi_K$.

Proposition 4.8. *Let $\alpha > \beta > 0$ be constants satisfying $(\sqrt{\alpha} - \sqrt{\beta})^2 > K^{d-1}(d-1)!$. Suppose that $n > \exp(16C^2/\gamma^2)\}$ and n is sufficiently large such that (19) holds. Then, with probability at least $1 - n^{-\Omega(1)}$, for any fixed $\boldsymbol{H} \in \mathcal{H}$ and $\varepsilon \in \left(0, \min\left\{\frac{1}{\sqrt{K(d-1)}}, \frac{\gamma K^{d-3/2}}{16(d-1)(\alpha-\beta)}\right\}\right)$ such that*

$$\|\boldsymbol{H} - \boldsymbol{H}^*\boldsymbol{Q}\|_F \leq \varepsilon\sqrt{n} \tag{20}$$

for some $\boldsymbol{Q} \in \Pi_K$, it holds that

$$\|\boldsymbol{V} - \boldsymbol{H}^*\boldsymbol{Q}\|_F \leq \kappa\|\boldsymbol{H} - \boldsymbol{H}^*\boldsymbol{Q}\|_F \tag{21}$$

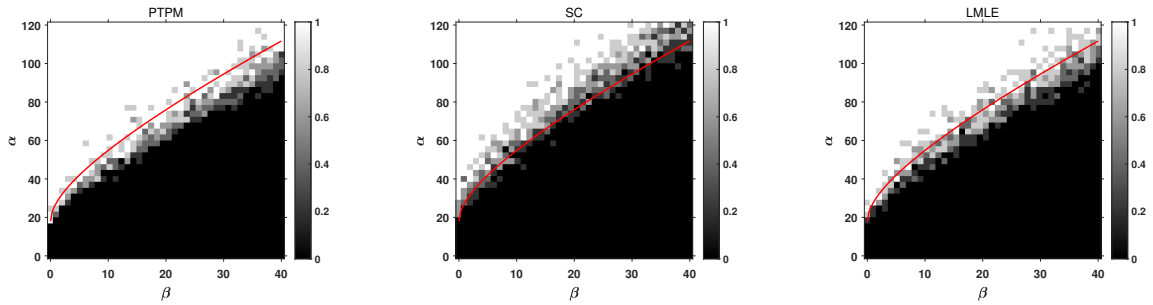
for any $\boldsymbol{V} \in \mathcal{T}\left(\mathcal{A}\left[\boldsymbol{H}^{\otimes(d-1)}\right]\right)$, where

$$\kappa = 4\max\left\{\frac{4(d-1)\varepsilon(\alpha-\beta)}{\gamma K^{d-3/2}}, \frac{C}{\gamma\sqrt{\log n}}\right\} \in (0,1).$$

Armed with the result of the local contraction property, we are ready to derive the iteration complexity of PTPM.

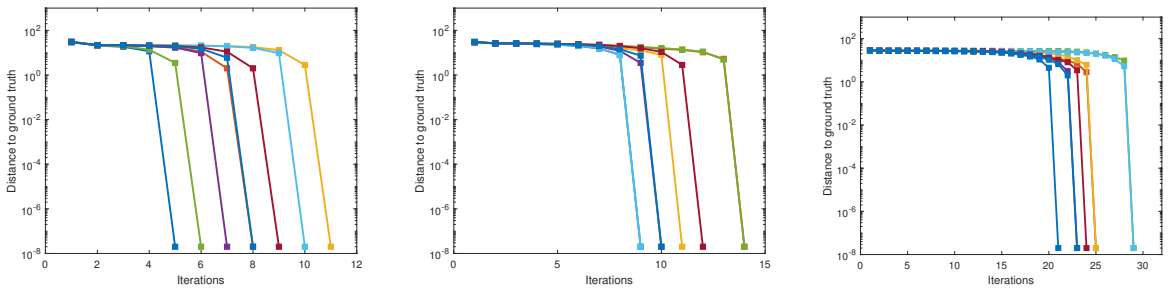
Theorem 4.9. *Denote $\phi = \frac{CK^{d-3/2}}{64(d-1)(\alpha-\beta)}$ and $\theta = \frac{1}{4}\min\left\{\frac{1}{\sqrt{K(d-1)}}, \frac{\gamma K^{d-3/2}}{16(d-1)(\alpha-\beta)}\right\}$. Suppose that $n > \exp\left(\max\left\{\frac{64C^2}{\gamma^2}, \frac{\gamma^2}{C^2}, \frac{\phi^2}{\theta^2}, 2\phi^2, \frac{256C^4}{\gamma^4}\right\}\right)$ and n is sufficiently large such that (19) holds. Then, the following statement holds with probability at least $1 - n^{-\Omega(1)}$: If the initial point $\boldsymbol{H}^0$ satisfies (11), then Algorithm 1 outputs $\boldsymbol{H}^*\boldsymbol{Q}$ within $\left(\lceil 2\log\log n\rceil + \left\lceil\frac{2\log n}{\log\log n}\right\rceil + 2\right)$ projected tensor power iterations.*

Recall that the matrix $\boldsymbol{H}^*\boldsymbol{Q}$ represents the same partition as $\boldsymbol{H}^*$ for any permutation matrix $\boldsymbol{Q}$. Hence, outputting $\boldsymbol{H}^*\boldsymbol{Q}$ in Theorem 4.9 would imply achieving exact recovery in Theorem 4.1. Further multiplying the derived iteration complexity by the time complexity of each projected tensor power iteration, we can obtain the total time complexity of PTPM in Theorem 4.1.



(a) PTPM (b) SC (c) LMLE

Figure 1. Phase transition in hypergraphs generated by 3-HSBM in the setting of $n = 210$, $K = 3$: The x-axis is β, the y-axis is α, and darker pixels represent lower empirical probability of success. The red curve is the information-theoretic threshold $\sqrt{\alpha} - \sqrt{\beta} = \sqrt{18}$.



(a) $(d, K, \alpha, \beta) = (3, 2, 33, 8)$ (b) $(d, K, \alpha, \beta) = (3, 4, 130, 32)$ (c) $(d, K, \alpha, \beta) = (3, 8, 400, 64)$

Figure 2. Convergence performance of PTPM with random initialization: The x-axis is the number of iterations, and the y-axis is the distance from an iterate to the ground truth, i.e., $\min_{\boldsymbol{Q} \in \Pi_K} \|\boldsymbol{H}^t - \boldsymbol{H}^* \boldsymbol{Q}\|_F$, where $\boldsymbol{H}^t$ is the t-th iterate generated by PTPM.

5. Experiment Results

In this section, we report the recovery performance and numerical efficiency of our proposed PTPM for recovering communities on synthetic/real data. We also compare our method with two existing methods, which are the spectral clustering (SC) method in Ghoshdastidar & Dukkipati (2015) and the local MLE (LMLE) method in Chien et al. (2019). In the implementation, we employ Chien et al. (2019, Algorithm 2) for computing an initial point $\boldsymbol{H}^0$ in Algorithm 1 if we do not specify the initialization method. Moreover, to possibly reduce computational time, we implement a simplified version of the LMLE method as stated in Chien et al. (2019, Remark 4.2). The MATLAB function `eigs` for computing the eigenvectors is applied in the SC method and the first stage of the LMLE method. The MATLAB function `kmeans` for computing the partition is used in the SC method. We use the Tensor Toolbox (Kolda et al., 2017) to perform tensor operations and compute $\mathcal{A}\left[(\boldsymbol{H}^t)^{\otimes(d-1)}\right]$ based on (7).

5.1. Phase Transition and Computational Time

We first conduct experiments to examine the phase transition property and the running time of the aforementioned methods for exact recovery under the symmetric 3-HSBM. We set $n = 210, K = 3$ in the experiments and let the parameter α vary from 0 to 120 with increments of 3 and the parameter β vary from 0 to 40 with increments of 1. For each pair of α and β, we generate 5 instances and calculate the success ratio of exact recovery for all the tested methods. The phase transition results are shown in Figure 1. It can be observed that all the methods exhibit a phase transition phenomenon. Moreover, Figure 1(a) indicates that PTPM achieves the optimal recovery threshold, which complements our theoretical findings. Besides, we record the total CPU time consumed by each method for completing the phase transition experiments with different parameters in Table 2. It can be observed that PTPM is faster than SC and substantially faster than LMLE.

Table 2. Total CPU time (in seconds) of the different methods in the phase transition experiments.

Time (s)	PTPM	SC	LMLE
$n = 210, K = 3, d = 3$	**76**	246	> 500
$n = 210, K = 6, d = 3$	**157**	451	> 1000
$n = 420, K = 3, d = 3$	**189**	321	> 1000
$n = 420, K = 6, d = 3$	**330**	628	> 1000

5.2. Convergence Performance

Next, we test the convergence performance of PTPM for exact community recovery under the symmetric d-HSBM. Specifically, we choose three different sets of (d, K, α, β) such that $\frac{(\sqrt{\alpha}-\sqrt{\beta})^2}{K^{d-1}(d-1)!} > 1$ and generate hypergraphs with $n = 480$ nodes. In addition, the initial point $\boldsymbol{H}^0$ in Algorithm 1 is generated via $\boldsymbol{H}^0 \in \mathcal{T}(\boldsymbol{G})$, where $\boldsymbol{G} \in \mathbb{R}^{n\times K}$ is a random Gaussian matrix. In each hypergraph realization, we run PTPM 8 times with different random initial points and then plot the distances of the iterative points to the ground truth, i.e., $\min_{\boldsymbol{Q}\in\Pi_K} \|\boldsymbol{H}^t - \boldsymbol{H}^*\boldsymbol{Q}\|_F$, against the iteration number in Figure 2. We see that PTPM achieves exact community recovery within 30 iterations even with random initialization. This demonstrates the power of PTPM in the ML estimation problem of the symmetric d-HSBM.

We also run a few experiments to assess the performance of PTPM when applying to a *non-uniform* hypergraph with 400 nodes ($n = 400$), where the 2-hyperedges and 3-hyperedges are generated independently via the graph SBM and the 3-HSBM with two underlying communities ($K = 2$), respectively. We follow the procedure mentioned in Remark 3.2 to reformulate each hypergraph as a uniform hypergraph and run the experiment 6 times with random initialization. The results show that exact community recovery could be achieved within 10 iterations, which demonstrates the power of PTPM when it is applied to non-uniform hypergraphs.

5.3. Recovery Accuracy and Efficiency

To evaluate the performance of PTPM, we also conduct a real data experiment shown as follows. Based on the 1984 US Congressional voting records available at the UCI repository[1], we choose two balanced communities ($K = 2$) with $n = 336$ number of Congressmen and randomly generate a 3-way symmetric adjacency tensor according to their votes on four issues (namely, columns 4, 5, 12, and 15 in the record). Specifically, if the three Congressmen indicate the same stance on a specific issue, we generate a 3-uniform hyperedge with a probability of 0.05 and no hyperedge otherwise; cf. Ghoshdastidar & Dukkipati (2017); Chien et al. (2019). To test the performance of PTPM, we generate an initial point $\boldsymbol{H}^0 \in \mathcal{T}(\boldsymbol{G})$ for some randomly generated Gaussian matrix $\boldsymbol{G}$ and terminate PTPM when the iteration number reaches 20. We run each algorithm 10 times and select the solution with the lowest function value as its recovered solution. The misclassification rate and running time are recorded in Table 3. As can be seen, our proposed method has better accuracy and efficiency than existing methods.

[1] http://archive.ics.uci.edu/ml

Table 3. The misclassification rate and the total CPU times (in seconds) of the methods on the UCI dataset.

Algorithms	PTPM	SC	LMLE
Misclassification rate	**0.07**	0.08	0.10
Time (s)	**0.85**	1.56	> 10

6. Conclusion

In this work, we developed a simple method, namely PTPM, for tackling the ML estimation problem of the symmetric d-HSBM. Our theory guarantees that, given an initialization satisfying the mild partial recovery condition, PTPM achieves exact community recovery down to the information-theoretic limit and has a low time complexity of $\mathcal{O}(n \log^2 n/\log\log n)$. One intriguing future direction is to design a simple iterative method (e.g., Wang et al. (2021b, Algorithm 1)) for obtaining a qualified initial point. Another future direction is to apply our method to other complex relational models with underlying community structures, such as multilayer networks (Jing et al., 2021; Fan et al., 2022; Chen et al., 2022) and multiplex networks (Pensky & Wang, 2021; Noroozi & Pensky, 2022).

7. Acknowledgments

The authors thank Prof. Yizhe Zhu, Dr. Yongfeng Li, and Prof. Xiongjun Zhang for their helpful discussions. This research is supported in part by the Hong Kong Research Grants Council (RGC) General Research Fund (GRF) project CUHK 14205421.

References

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

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

Ahn, K., Lee, K., and Suh, C. Hypergraph spectral clustering in the weighted stochastic block model. *IEEE Journal of Selected Topics in Signal Processing*, 12(5):959–974, 2018.

Araya, E., Braun, G., and Tyagi, H. Seeded graph matching for the correlated Wigner model via the projected power method. *arXiv preprint arXiv:2204.04099*, 2022.

Boumal, N. Nonconvex phase synchronization. *SIAM Journal on Optimization*, 26(4):2355–2377, 2016.

Cabreros, I., Abbe, E., and Tsirigos, A. Detecting community structures in HI-C genomic data. In *Proceedings*

of 2016 Annual Conference on Information Science and Systems, pp. 584–589. IEEE, 2016.

Candes, E. J., Li, X., and Soltanolkotabi, M. Phase retrieval via Wirtinger flow: Theory and algorithms. *IEEE Transactions on Information Theory*, 61(4):1985–2007, 2015.

Chen, S., Liu, S., and Ma, Z. Global and individualized community detection in inhomogeneous multilayer networks. *Annals of Statistics*, 50(5):2664–2693, 2022.

Chen, Y. and Candès, E. J. The projected power method: An efficient algorithm for joint alignment from pairwise differences. *Communications on Pure and Applied Mathematics*, 71(8):1648–1714, 2018.

Chen, Y., Chi, Y., Fan, J., and Ma, C. Gradient descent with random initialization: Fast global convergence for nonconvex phase retrieval. *Mathematical Programming*, 176(1):5–37, 2019.

Chi, Y., Lu, Y. M., and Chen, Y. Nonconvex optimization meets low-rank matrix factorization: An overview. *IEEE Transactions on Signal Processing*, 67(20):5239–5269, 2019.

Chien, I., Lin, C.-Y., and Wang, I.-H. Community detection in hypergraphs: Optimal statistical limit and efficient algorithms. In *Proceedings of the 21st International Conference on Artificial Intelligence and Statistics*, pp. 871–879. PMLR, 2018.

Chien, I. E., Lin, C.-Y., and Wang, I.-H. On the minimax misclassification ratio of hypergraph community detection. *IEEE Transactions on Information Theory*, 65(12): 8095–8118, 2019.

Cichocki, A., Lee, N., Oseledets, I., Phan, A.-H., Zhao, Q., and Mandic, D. P. Tensor networks for dimensionality reduction and large-scale optimization: Part 1 Low-rank tensor decompositions. *Foundations and Trends® in Machine Learning*, 9(4-5):249–429, 2016.

Cole, S. and Zhu, Y. Exact recovery in the hypergraph stochastic block model: A spectral algorithm. *Linear Algebra and its Applications*, 593:45–73, 2020.

Deshpande, Y., Montanari, A., and Richard, E. Cone-constrained principal component analysis. In *Advances in Neural Information Processing Systems 27*, pp. 2717–2725, 2014.

Duchenne, O., Bach, F., Kweon, I.-S., and Ponce, J. A tensor-based algorithm for high-order graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(12):2383–2395, 2011.

Dumitriu, I., Wang, H., and Zhu, Y. Partial recovery and weak consistency in the non-uniform hypergraph stochastic block model. *arXiv preprint arXiv:2112.11671*, 2021.

Fan, X., Pensky, M., Yu, F., and Zhang, T. ALMA: Alternating minimization algorithm for clustering mixture multilayer network. *Journal of Machine Learning Research*, 23(330):1–46, 2022.

Feige, U. and Ofek, E. Spectral techniques applied to sparse random graphs. *Random Structures & Algorithms*, 27(2): 251–275, 2005.

Gaudio, J. and Joshi, N. Community detection in the hypergraph SBM: Optimal recovery given the similarity matrix. *arXiv preprint arXiv:2208.12227*, 2022.

Ghoshdastidar, D. and Dukkipati, A. Consistency of spectral partitioning of uniform hypergraphs under planted partition model. In *Advances in Neural Information Processing Systems 27*, pp. 397–405, 2014.

Ghoshdastidar, D. and Dukkipati, A. A provable generalized tensor spectral method for uniform hypergraph partitioning. In *Proceedings of the 32nd International Conference on Machine Learning*, pp. 400–409. PMLR, 2015.

Ghoshdastidar, D. and Dukkipati, A. Consistency of spectral hypergraph partitioning under planted partition model. *Annals of Statistics*, 45(1):289–315, 2017.

Girvan, M. and Newman, M. E. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

Han, R., Luo, Y., Wang, M., Zhang, A. R., et al. Exact clustering in tensor block model: Statistical optimality and computational limit. *Journal of the Royal Statistical Society, Series B*, 84(5):1666–1698, 2022.

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

Huang, J., Huang, D. Z., Yang, Q., and Cheng, G. Power iteration for tensor PCA. *Journal of Machine Learning Research*, 23(128):1–47, 2022.

Jing, B.-Y., Li, T., Lyu, Z., and Xia, D. Community detection on mixture multilayer networks via regularized tensor decomposition. *Annals of Statistics*, 49(6):3181–3205, 2021.

Journée, M., Nesterov, Y., Richtárik, P., and Sepulchre, R. Generalized power method for sparse principal component analysis. *Journal of Machine Learning Research*, 11 (2), 2010.

Ke, Z. T., Shi, F., and Xia, D. Community detection for hypergraph networks via regularized tensor power iteration. *arXiv preprint arXiv:1909.06503*, 2019.

Kim, C., Bandeira, A. S., and Goemans, M. X. Community detection in hypergraphs, spiked tensor models, and sum-of-squares. In *Proceedings of 2017 International Conference on Sampling Theory and Applications*, pp. 124–128. IEEE, 2017.

Kim, C., Bandeira, A. S., and Goemans, M. X. Stochastic block model for hypergraphs: Statistical limits and a semidefinite programming approach. *arXiv preprint arXiv:1807.02884*, 2018.

Kolda, T. G. and Bader, B. W. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009.

Kolda, T. G., Bader, B. W., Acar Ataman, E. N., Dunlavy, D., Bassett, R., Battaglino, C. J., Plantenga, T., Chi, E., Hansen, S., and USDOE. Tensor toolbox for matlab v. 3.0, 3 2017. URL https://www.osti.gov//servlets/purl/1349514.

Lee, S. H., Magallanes, J. M., and Porter, M. A. Time-dependent community structure in legislation cosponsorship networks in the congress of the republic of Peru. *Journal of Complex Networks*, 5(1):127–144, 2017.

Lei, J. and Rinaldo, A. Consistency of spectral clustering in stochastic block models. *Annals of Statistics*, 43(1): 215–237, 2015.

Li, X., Zhu, Z., Man-Cho So, A., and Vidal, R. Nonconvex robust low-rank matrix recovery. *SIAM Journal on Optimization*, 30(1):660–686, 2020.

Ling, S. Improved performance guarantees for orthogonal group synchronization via generalized power method. *SIAM Journal on Optimization*, 32(2):1018–1048, 2022.

Liu, H., Yue, M.-C., and Man-Cho So, A. On the estimation performance and convergence rate of the generalized power method for phase synchronization. *SIAM Journal on Optimization*, 27(4):2426–2446, 2017.

Michoel, T. and Nachtergaele, B. Alignment and integration of complex networks by hypergraph-based spectral clustering. *Physical Review E*, 86(5):056111, 2012.

Noroozi, M. and Pensky, M. Sparse subspace clustering in diverse multiplex network model. *arXiv preprint arXiv:2206.07602*, 2022.

Pensky, M. and Wang, Y. Clustering of diverse multiplex networks. *arXiv preprint arXiv:2110.05308*, 2021.

Richard, E. and Montanari, A. A statistical model for tensor PCA. In *Advances in Neural Information Processing Systems 27*, pp. 2897–2905, 2014.

Shi, J. and Malik, J. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

Sidiropoulos, N. D., De Lathauwer, L., Fu, X., Huang, K., Papalexakis, E. E., and Faloutsos, C. Tensor decomposition for signal processing and machine learning. *IEEE Transactions on Signal Processing*, 65(13):3551–3582, 2017.

Vershynin, R. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

Wang, P., Zhou, Z., and So, A. M.-C. A nearly-linear time algorithm for exact community recovery in stochastic block model. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 10126–10135. PMLR, 2020.

Wang, P., Liu, H., Zhou, Z., and So, A. M.-C. Optimal non-convex exact recovery in stochastic block model via projected power method. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 10828–10838. PMLR, 2021a.

Wang, P., Zhou, Z., and So, A. M.-C. Non-convex exact community recovery in stochastic block model. *Mathematical Programming*, pp. 1–37, 2021b.

Wang, X., Wang, P., and So, A. M.-C. Exact community recovery over signed graphs. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics*, pp. 9686–9710. PMLR, 2022.

Zhang, Q. and Tan, V. Y. Exact recovery in the general hypergraph stochastic block model. *IEEE Transactions on Information Theory*, 2022.

Zhong, Y. and Boumal, N. Near-optimal bounds for phase synchronization. *SIAM Journal on Optimization*, 28(2): 989–1016, 2018.

Zhou, Z. and Zhu, Y. Sparse random tensors: Concentration, regularization and applications. *Electronic Journal of Statistics*, 15(1):2483–2516, 2021.

Zhu, L., Wang, J., and So, A. M.-C. Orthogonal group synchronization with incomplete measurements: Error bounds and linear convergence of the generalized power method. *arXiv preprint arXiv:2112.06556*, 2021.

Appendix

In the appendix, we provide proofs of the technical results presented in Section 4.

A. Proof of Theorem 4.1

Proof. From Theorem 4.9, we know that Algorithm 1 yields a true partition in $\left(\lceil 2\log\log n\rceil + \left\lceil \frac{2\log n}{\log\log n}\right\rceil + 2\right)$ projected tensor power iterations with probability at least $1 - n^{-\Omega(1)}$. Besides, the time complexity of performing the projection operation $\mathcal{T}$ (i.e., solving (10)) is $\mathcal{O}(K^2 n\log n)$ (Wang et al., 2021a, Proposition 1). It remains to show that the time complexity of computing $\mathcal{A}\left[(\boldsymbol{H}^t)^{\otimes(d-1)}\right]$ (or equivalently $\mathcal{M}(\mathcal{A})\left((\boldsymbol{H}^t)^{\odot(d-1)}\right)$ by (7)) is $\mathcal{O}(n\log n)$ with high probability. Since $\mathcal{A}$ is generated according to the symmetric d-HSBM with $p = \alpha\log n/n^{d-1}$ and $q = \beta\log n/n^{d-1}$, by denoting the number of nonzero entries of $\mathcal{A}$ as $\|\mathcal{A}\|_0$, we have

$$\|\mathcal{A}\|_0 \overset{\mathrm{d}}{=} d!\cdot\sum_{i=1}^{K\binom{m}{d}} W_i + d!\cdot\sum_{i=1}^{\binom{n}{d}-K\binom{m}{d}} Z_i, \tag{22}$$

where $\{W_i\}_{i=1}^{K\binom{m}{d}}$ are i.i.d. $\mathbf{Bern}(p)$ and $\{Z_i\}_{i=1}^{\binom{n}{d}-K\binom{m}{d}}$ are i.i.d. $\mathbf{Bern}(q)$, independent of $\{W_i\}_{i=1}^{K\binom{m}{d}}$. The expectation and variance of $\|\mathcal{A}\|_0$ are given by

$$\begin{aligned}
\mathbb{E}[\|\mathcal{A}\|_0] &= d!\cdot\left(K\tbinom{m}{d}p + \left(\tbinom{n}{d} - K\tbinom{m}{d}\right)q\right),\\
\mathrm{Var}\left[\|\mathcal{A}\|_0\right] &= (d!)^2\cdot\left(K\tbinom{m}{d}p(1-p) + \left(\tbinom{n}{d} - K\tbinom{m}{d}\right)q(1-q)\right)\\
&\le (d!)^2\cdot\left(K\tbinom{m}{d}p + \left(\tbinom{n}{d} - K\tbinom{m}{d}\right)q\right).
\end{aligned}$$

Applying Bernstein's inequality (Vershynin, 2018, Theorem 2.8.4) gives

$$\begin{aligned}
&\Pr\left[\left|\|\mathcal{A}\|_0 - d!\cdot\left(K\tbinom{m}{d}p + \left(\tbinom{n}{d} - K\tbinom{m}{d}\right)q\right)\right| \ge 2d!\cdot\left(K\tbinom{m}{d}p + \left(\tbinom{n}{d} - K\tbinom{m}{d}\right)q\right)\right]\\
&= \Pr\left[\left|\frac{\|\mathcal{A}\|_0}{d!} - \left(K\tbinom{m}{d}p + \left(\tbinom{n}{d} - K\tbinom{m}{d}\right)q\right)\right| \ge 2\left(K\tbinom{m}{d}p + \left(\tbinom{n}{d} - K\tbinom{m}{d}\right)q\right)\right]\\
&\le 2\exp\left(-\frac{2\left(K\binom{m}{d}p + \left(\binom{n}{d} - K\binom{m}{d}\right)q\right)^2}{\left(K\binom{m}{d}p + \left(\binom{n}{d} - K\binom{m}{d}\right)q\right) + 2\left(K\binom{m}{d}p + \left(\binom{n}{d} - K\binom{m}{d}\right)q\right)/3}\right)\\
&\le 2\exp\left(-\left(K\tbinom{m}{d}p + \left(\tbinom{n}{d} - K\tbinom{m}{d}\right)q\right)\right)\\
&\le 2\exp\left(-\left(\frac{Km^d}{2^{d-1}d!}p + \frac{n^d}{2^{d-1}d!}q - K\frac{m^d}{d!}q\right)\right)\\
&= 2n^{-\left(\frac{\beta}{2^{d-1}d!} + \frac{K\alpha}{2^{d-1}K^d d!} - \frac{K\beta}{K^d d!}\right)n} \le 2n^{-\left(\frac{K\alpha}{2^{d-1}K^d d!}\right)n},
\end{aligned}$$

where we assume $m - d + 1 \ge m/2$ in the third inequality and the last inequality is due to $K \ge 2$. It holds with probability at least $1 - 2n^{-\left(\frac{K\alpha}{2^{d-1}K^d d!}\right)n}$ that the number of nonzero entries in $\mathcal{A}$ is less than $3d!\cdot\left(K\binom{m}{d}p + \left(\binom{n}{d} - K\binom{m}{d}\right)q\right) \le \frac{3\alpha + 3K^{d-1}\beta}{K^{d-1}}n\log n$. Thus, the time complexity of computing $\mathcal{A}\left[(\boldsymbol{H}^t)^{\otimes(d-1)}\right]$ is $\frac{3\alpha+3K^{d-1}\beta}{K^{d-1}}n\log n$ with probability at least $1 - 2n^{-\left(\frac{K\alpha}{2^{d-1}K^d d!}\right)n}$. The desired bound on the total time complexity of Algorithm 1 then follows. □

B. Proof of Lemma 4.2

Proof. Let $\Delta = \mathcal{A} - \mathbb{E}[\mathcal{A}]$. Without loss of generality, we assume that $\boldsymbol{H}^* = \boldsymbol{I}_K \otimes \mathbf{1}_m \in \mathbb{R}^{n\times K}$, which implies that

$$(\boldsymbol{H}^*)^{\otimes d} = \left(\boldsymbol{I}_K^{\otimes d}\right)\otimes\left(\mathbf{1}_m^{\otimes d}\right) \in S^d(\mathbb{R}^n);$$

see Cichocki et al. (2016, Section 2.1) for the Kronecker product of tensors. By the definition of $\mathcal{A}$, one can verify that

$$\mathbb{E}[\mathcal{A}] = \mathcal{B}\otimes\left(\mathbf{1}_m^{\otimes d}\right) = q\cdot\left(\mathbf{1}_n^{\otimes d}\right) + (p-q)\cdot(\boldsymbol{H}^*)^{\otimes d}, \tag{23}$$

where $\mathcal{B} \in S^d(\mathbb{R}^K)$ is such that $\mathcal{B}_{k_1,\ldots,k_d}$ equals to p if $k_1 = \cdots = k_d$ and q otherwise. Let us decompose $(\boldsymbol{H}_j)^{\otimes(d-1)}, j \in [K]$ into several orthogonal parts:

$$(\boldsymbol{H}_j)^{\otimes(d-1)} = \sum_{k=1}^{K} ((\boldsymbol{H}^*\boldsymbol{Q})_k)^{\otimes(d-1)} Z_{kj} + \mathcal{G}_j. \tag{24}$$

Here, $\boldsymbol{Z} \in \mathbb{R}^{K\times K}$ is given by

$$Z_{kj} = \frac{1}{m^{d-1}} \left\langle (\boldsymbol{H}_j)^{\otimes(d-1)}, ((\boldsymbol{H}^*\boldsymbol{Q})_k)^{\otimes(d-1)} \right\rangle = \frac{1}{m^{d-1}} \left(\langle \boldsymbol{H}_j, (\boldsymbol{H}^*\boldsymbol{Q})_k \rangle\right)^{d-1} \tag{25}$$

and $\mathcal{G}_j \in S^{d-1}(\mathbb{R}^n)$ satisfies $\left\langle \mathcal{G}_j, ((\boldsymbol{H}^*\boldsymbol{Q})_k)^{\otimes(d-1)} \right\rangle = 0$ for $k \in [K]$. It follows from the definition in (3) that for $i \in [n], k \in [K]$,

$$\left((\mathbf{1}_n^{\otimes d}) \left[\boldsymbol{H}^{\otimes(d-1)} \right] \right)_{ik} = \left(\mathbf{1}_n^\top \boldsymbol{H}_k \right)^{d-1} = m^{d-1} = \left((\mathbf{1}_n^{\otimes d}) \left[(\boldsymbol{H}^*\boldsymbol{Q})^{\otimes(d-1)} \right] \right)_{ik}. \tag{26}$$

In addition, it is implied by (3) that

$$\begin{aligned}
&\left(\left((\boldsymbol{H}^*)^{\otimes d} \right) \left[\boldsymbol{H}^{\otimes(d-1)} \right] \right)_{ik} \\
&= \left(\left((\boldsymbol{I}_K \otimes \mathbf{1}_m)^{\otimes d} \right) \left[\boldsymbol{H}^{\otimes(d-1)} \right] \right)_{ik} \\
&= \sum_{1\le i_2,\ldots,i_d\le n} \left((\boldsymbol{I}_K \otimes \mathbf{1}_m)^{\otimes d} \right)_{i,i_2,\ldots,i_d} \times \boldsymbol{H}_{i_2k} \times \cdots \times \boldsymbol{H}_{i_dk} \\
&= \sum_{1\le i_2,\ldots,i_d\le n} \sum_{j=1}^{K} (\boldsymbol{I}_K \otimes \mathbf{1}_m)_{ij} \times (\boldsymbol{I}_K \otimes \mathbf{1}_m)_{i_2j} \times \cdots \times (\boldsymbol{I}_K \otimes \mathbf{1}_m)_{i_dj} \times \boldsymbol{H}_{i_2k} \times \cdots \times \boldsymbol{H}_{i_dk} \\
&= \sum_{j=1}^{K} (\boldsymbol{I}_K \otimes \mathbf{1}_m)_{ij} \left(\sum_{1\le i_2,\ldots,i_d\le n} (\boldsymbol{I}_K \otimes \mathbf{1}_m)_{i_2j} \times \cdots \times (\boldsymbol{I}_K \otimes \mathbf{1}_m)_{i_dj} \times \boldsymbol{H}_{i_2k} \times \cdots \times \boldsymbol{H}_{i_dk} \right) \\
&= \sum_{j=1}^{K} (\boldsymbol{I}_K \otimes \mathbf{1}_m)_{ij} \left(\mathbf{1}_n^\top \left((\boldsymbol{I}_K \otimes \mathbf{1}_m)_j \circ \boldsymbol{H}_k \right) \right)^{d-1} \\
&= \sum_{j=1}^{K} (\boldsymbol{I}_K \otimes \mathbf{1}_m)_{ij} D_{jk}.
\end{aligned} \tag{27}$$

Here, " $\circ$ " represents element-wise multiplication, $\boldsymbol{H}_{j,k} \in \mathbb{R}^m$ denotes the subvector of $\boldsymbol{H}_k$ from H_{i_1k} to H_{i_2k} with $i_1 = (j-1)\cdot m + 1$ and $i_2 = j \cdot m$, and $\boldsymbol{D} \in \mathbb{R}^{K\times K}$ is the matrix given by $D_{jk} = (\mathbf{1}_m^\top \boldsymbol{H}_{j,k})^{d-1}$ for $j \in [K], k \in [K]$. One can verify that $\boldsymbol{D} = m^{d-1}\boldsymbol{Q}\boldsymbol{Z}$ according to (25). Combining (26), (27) as well as (23) yields

$$\begin{aligned}
(\mathbb{E}[\mathcal{A}]) \left[\boldsymbol{H}^{\otimes(d-1)} \right] - (\mathbb{E}[\mathcal{A}]) \left[(\boldsymbol{H}^*\boldsymbol{Q})^{\otimes(d-1)} \right] &= (p-q) \cdot \left(\left((\boldsymbol{H}^*)^{\otimes d} \right) \left[\boldsymbol{H}^{\otimes(d-1)} \right] - \left((\boldsymbol{H}^*)^{\otimes d} \right) \left[(\boldsymbol{H}^*\boldsymbol{Q})^{\otimes(d-1)} \right] \right) \\
&= (p-q) \cdot (\boldsymbol{I}_K \otimes \mathbf{1}_m)(m^{d-1}\boldsymbol{Q}\boldsymbol{Z} - m^{d-1}\boldsymbol{Q}).
\end{aligned}$$

This implies that

$$\left\| (\mathbb{E}[\mathcal{A}]) \left[\boldsymbol{H}^{\otimes(d-1)} \right] - (\mathbb{E}[\mathcal{A}]) \left[(\boldsymbol{H}^*\boldsymbol{Q})^{\otimes(d-1)} \right] \right\|_F \le \sqrt{m}(p-q)m^{d-1} \|\boldsymbol{I}_K - \boldsymbol{Z}\|_F. \tag{28}$$

Next, we provide an upper bound for the term $\|\boldsymbol{I}_K - \boldsymbol{Z}\|_F$. Note that

$$\|\boldsymbol{I}_K - \boldsymbol{Z}\|_F^2 = \sum_{k=1}^{K} (1 - Z_{kk})^2 + \sum_{k\neq\ell} Z_{k\ell}^2. \tag{29}$$

Since $Z_{k\ell} \in [0,1]$ for $k, \ell \in [K]$, it follows that

$$\begin{aligned}\|\boldsymbol{I}_K - \boldsymbol{Z}\|_F &\le \sum_{k=1}^{K} |1 - Z_{kk}| + \sum_{k\neq\ell} |Z_{k\ell}| = \sum_{k=1}^{K}(1 - Z_{kk}) + \sum_{k\neq\ell} Z_{k\ell} \\ &\le \sum_{k=1}^{K}(1 - Z_{kk}) + \sum_{k\neq\ell} Z_{k\ell}^{\frac{1}{d-1}}. \end{aligned} \tag{30}$$

By the definition of $\boldsymbol{Z}$ in (25), we have

$$\sum_{\ell=1}^{K} Z_{k\ell}^{\frac{1}{d-1}} = \frac{1}{m}\sum_{\ell=1}^{K}\langle \boldsymbol{H}_\ell, (\boldsymbol{H}^*\boldsymbol{Q})_k\rangle = 1 \tag{31}$$

for $k \in [K]$. Then, we can further estimate (30) by

$$\|\boldsymbol{I}_K - \boldsymbol{Z}\|_F \le \sum_{k=1}^{K}(1 - Z_{kk}) + \sum_{k=1}^{K}\left(1 - Z_{kk}^{\frac{1}{d-1}}\right) \le 2\sum_{k=1}^{K}(1 - Z_{kk}). \tag{32}$$

Here, again, we have used the fact that $Z_{kk} \in [0,1]$ for $k \in [K]$. The orthogonal decomposition in (24) gives

$$\sum_{j=1}^{K}\langle \mathcal{G}_j, \mathcal{G}_j\rangle = m^{d-1}K - m^{d-1}\sum_{k=1}^{K}\sum_{j=1}^{K} Z_{kj}^2. \tag{33}$$

In addition, using the fact that

$$m^{d-1} - s^{d-1} = (m-s)\left(m^{d-2} + m^{d-3}s + \cdots + ms^{d-3} + s^{d-2}\right) \text{ for } 0 \le s \le m,$$

one can verify the following inequalities:

$$m^{d-2}\|\boldsymbol{H} - \boldsymbol{H}^*\boldsymbol{Q}\|_F^2 \le \sum_{k=1}^{K}\left\|(\boldsymbol{H}_k)^{\otimes(d-1)} - ((\boldsymbol{H}^*\boldsymbol{Q})_k)^{\otimes(d-1)}\right\|_F^2 \le (d-1)m^{d-2}\|\boldsymbol{H} - \boldsymbol{H}^*\boldsymbol{Q}\|_F^2. \tag{34}$$

According to $\|\boldsymbol{H} - \boldsymbol{H}^*\boldsymbol{Q}\|_F \le \varepsilon\sqrt{n}$, we have

$$\begin{aligned}\sum_{k=1}^{K}\|(\boldsymbol{H}_k)^{\otimes(d-1)} - ((\boldsymbol{H}^*\boldsymbol{Q})_k)^{\otimes(d-1)}\|_F^2 &= m^{d-1}\left(\sum_{k=1}^{K}(1 - Z_{kk})^2 + \sum_{k\neq j} Z_{kj}^2\right) + \sum_{j=1}^{K}\langle \mathcal{G}_j, \mathcal{G}_j\rangle \\ &= m^{d-1}\|\boldsymbol{I}_K - \boldsymbol{Z}\|_F^2 + \sum_{j=1}^{K}\langle \mathcal{G}_j, \mathcal{G}_j\rangle \le (d-1)m^{d-2}\varepsilon^2 n,\end{aligned}$$

where the first equality is due to (24) and the inequality comes from (34). This, together with (29) and (33), implies that

$$\sum_{k=1}^{K} Z_{kk} \ge \left(1 - \frac{(d-1)\varepsilon^2}{2}\right)K. \tag{35}$$

Then, for any $\ell \in [K]$, we have

$$Z_{\ell\ell} \ge \left(1 - \frac{(d-1)\varepsilon^2}{2}\right)K - \sum_{k\neq\ell} Z_{kk} \ge 1 - \frac{K}{2}(d-1)\varepsilon^2 \ge \frac{1}{2}, \tag{36}$$

where the second inequality is due to $Z_{kk} \le 1$ for all $k \in [K]$ and the last inequality is from $\varepsilon \in (0, 1/\sqrt{K(d-1)})$. According to (33), we know that

$$\begin{aligned}\frac{1}{m^{d-1}}\sum_{j=1}^{K}\langle \mathcal{G}_j, \mathcal{G}_j\rangle &= K - \sum_{k=1}^{K}\sum_{\ell=1}^{K} Z_{k\ell}^2 \ge K - \sum_{k=1}^{K} Z_{kk}^2 - \sum_{k\neq \ell} Z_{k\ell}^{\frac{1}{d-1}} \\ &= K - \sum_{k=1}^{K} Z_{kk}^2 - \sum_{k=1}^{K}\left(1 - Z_{kk}^{\frac{1}{d-1}}\right) = \sum_{k=1}^{K} Z_{kk}^{\frac{1}{d-1}}\left(1 - Z_{kk}^{2-\frac{1}{d-1}}\right) \\ &\ge \sum_{k=1}^{K}\left(\frac{1}{2}\right)^{\frac{1}{d-1}}(1 - Z_{kk}), \end{aligned} \tag{37}$$

where the first inequality is due to $Z_{k\ell} \in [0,1]$ for $k \in [K], \ell \in [K]$, the second equality comes from (31), and the last inequality uses (36) and $2 - \frac{1}{d-1} \ge 1$ when $d \ge 2$. This, together with (32) and (34), gives

$$\begin{aligned}\|\boldsymbol{I}_K - \boldsymbol{Z}\|_F &\le 2^{\frac{1}{d-1}}\frac{2}{m^{d-1}}\sum_{j=1}^{K}\langle \mathcal{G}_j, \mathcal{G}_j\rangle \le \frac{4}{m^{d-1}}\sum_{k=1}^{K}\left\|(\boldsymbol{H}_k)^{\otimes(d-1)} - ((\boldsymbol{H}^*\boldsymbol{Q})_k)^{\otimes(d-1)}\right\|_F^2 \\ &\le \frac{4}{m^{d-1}}(d-1)m^{d-2}\|\boldsymbol{H} - \boldsymbol{H}^*\boldsymbol{Q}\|_F^2 \le \frac{4(d-1)\varepsilon\sqrt{n}}{m}\|\boldsymbol{H} - \boldsymbol{H}^*\boldsymbol{Q}\|_F.\end{aligned}$$

Next, we consider the term $\left\|\Delta\left[\boldsymbol{H}^{\otimes(d-1)}\right] - \Delta\left[(\boldsymbol{H}^*\boldsymbol{Q})^{\otimes(d-1)}\right]\right\|_F$ and provide an upper bound for it. Note that

$$\left\|\Delta\left[\boldsymbol{H}_k^{\otimes(d-1)}\right] - \Delta\left[(\boldsymbol{H}^*\boldsymbol{Q})_k^{\otimes(d-1)}\right]\right\|_2 = \left\|\mathcal{M}(\Delta)\left(\boldsymbol{H}_k^{\odot(d-1)} - (\boldsymbol{H}^*\boldsymbol{Q})_k^{\odot(d-1)}\right)\right\|_2. \tag{38}$$

From Lemma 4.3, we know that

$$\left\|\mathcal{M}(\Delta)\left(\boldsymbol{H}_k^{\odot(d-1)} - (\boldsymbol{H}^*\boldsymbol{Q})_k^{\odot(d-1)}\right)\right\|_2 \le C\sqrt{\log n}\|\boldsymbol{H}_k - (\boldsymbol{H}^*\boldsymbol{Q})_k\|_2, \tag{39}$$

which implies that

$$\left\|\Delta\left[\boldsymbol{H}^{\otimes(d-1)}\right] - \Delta\left[(\boldsymbol{H}^*\boldsymbol{Q})^{\otimes(d-1)}\right]\right\|_F \le C\sqrt{\log n}\|\boldsymbol{H} - \boldsymbol{H}^*\boldsymbol{Q}\|_F. \tag{40}$$

The desired result (13) is then established, because

$$\begin{aligned}&\left\|\mathcal{A}\left[\boldsymbol{H}^{\otimes(d-1)}\right] - \mathcal{A}\left[(\boldsymbol{H}^*\boldsymbol{Q})^{\otimes(d-1)}\right]\right\|_F \\ \le &\left\|(\mathbb{E}[\mathcal{A}])\left[\boldsymbol{H}^{\otimes(d-1)}\right] - (\mathbb{E}[\mathcal{A}])\left[(\boldsymbol{H}^*\boldsymbol{Q})^{\otimes(d-1)}\right]\right\|_F + \left\|\Delta\left[\boldsymbol{H}^{\otimes(d-1)}\right] - \Delta\left[(\boldsymbol{H}^*\boldsymbol{Q})^{\otimes(d-1)}\right]\right\|_F \\ = &\left(\frac{4(d-1)m^{d-2}\varepsilon n}{\sqrt{K}}(p-q) + C\sqrt{\log n}\right)\|\boldsymbol{H} - \boldsymbol{H}^*\boldsymbol{Q}\|_F.\end{aligned}$$

This completes the proof. □

C. Proof of Lemma 4.3

Proof. By replacing $\boldsymbol{y}$ with $\frac{\boldsymbol{y}}{\|\boldsymbol{y}\|_2}$ in (15), it suffices to prove that for any $r > 0$ and any vector $\boldsymbol{y} \in \mathbb{R}^{n^{d-1}}$ with $\ell \cdot n^{d-2}$ nonzero elements, in which half of them take values of $1/\sqrt{\ell \cdot n^{d-2}}$ and others take $-1/\sqrt{\ell \cdot n^{d-2}}$, there exists a constant $C > 0$ such that

$$\|(\boldsymbol{A} - \boldsymbol{P})\boldsymbol{y}\|_2 \le C\sqrt{\xi/n^{d-2}}.$$

The fact that

$$\|(\boldsymbol{A} - \boldsymbol{P})\boldsymbol{y}\|_2 = \sup_{\boldsymbol{x}\in\mathbb{R}^n, \|\boldsymbol{x}\|_2\le 1} \boldsymbol{x}^\top(\boldsymbol{A} - \boldsymbol{P})\boldsymbol{y} \tag{41}$$

motivates us to utilize the Kahn-Szemerédi argument to provide an upper bound for (41). The idea of the proof is to discretize the set $\{\boldsymbol{x} \in \mathbb{R}^n \colon \|\boldsymbol{x}\|_2 \le 1\}$ into a finite set of grid points and estimate the supremum of $\boldsymbol{x}^T(\boldsymbol{A} - \boldsymbol{P})\boldsymbol{y}$ by dividing the pairs of vectors $(\boldsymbol{x}, \boldsymbol{y})$ into two parts: (i) the small entries of $\boldsymbol{x}$ and $\boldsymbol{y}$, which we call the *light part*; and (ii) the larger entries of $\boldsymbol{x}$ and $\boldsymbol{y}$, which we call the *heavy part*.

Discretization Fix $\delta \in (0,1)$, for example $\delta = \frac{1}{2}$, and define the sets $S_t \coloneqq \{\boldsymbol{x} \in \mathbb{R}^n : \|\boldsymbol{x}\|_2 \le t\}$ and

$$T \coloneqq \left\{ \boldsymbol{x} = (x_1, \ldots, x_n)^\top \in S_1 : \sqrt{n} x_i / \delta \in \mathbb{Z}, \forall i \in [n] \right\}.$$

Following Lei & Rinaldo (2015, Supplementary material: Lemma 2.1), we have the following inequality:

$$\|(\boldsymbol{A} - \boldsymbol{P})\boldsymbol{y}\|_2 = \sup_{\boldsymbol{x} \in \mathbb{R}^n, \|\boldsymbol{x}\|_2 \le 1} \boldsymbol{x}^\top (\boldsymbol{A} - \boldsymbol{P})\boldsymbol{y} \le (1-\delta)^{-1} \max_{\boldsymbol{x} \in T} \left| \boldsymbol{x}^\top (\boldsymbol{A} - \boldsymbol{P})\boldsymbol{y} \right|. \tag{42}$$

Note that, for any vector $\boldsymbol{x} \in T$,

$$\boldsymbol{x}^\top (\boldsymbol{A} - \boldsymbol{P})\boldsymbol{y} = \sum_{i \in [n], j \in [n^{d-1}]} x_i y_j \left(a_{ij} - p_{ij}\right).$$

Consider the light pairs

$$\mathcal{L} = \mathcal{L}(\boldsymbol{x}, \boldsymbol{y}) \coloneqq \left\{ (i,j) : |x_i y_j| \le \sqrt{\frac{\xi}{n^d}} \right\}$$

and heavy pairs

$$\overline{\mathcal{L}} = \overline{\mathcal{L}}(\boldsymbol{x}, \boldsymbol{y}) \coloneqq \left\{ (i,j) : |x_i y_j| > \sqrt{\frac{\xi}{n^d}} \right\}.$$

Then, it follows from (42) that

$$\begin{aligned}
\|(\boldsymbol{A} - \boldsymbol{P})\boldsymbol{y}\|_2 &\le (1-\delta)^{-1} \max_{\boldsymbol{x} \in T} \left| \sum_{i \in [n], j \in [n^{d-1}]} x_i y_j \left(a_{ij} - p_{ij}\right) \right| \\
&\le (1-\delta)^{-1} \left(\max_{\boldsymbol{x} \in T} \left| \sum_{(i,j) \in \mathcal{L}} x_i y_j \left(a_{ij} - p_{ij}\right) \right| + \max_{\boldsymbol{x} \in T} \left| \sum_{(i,j) \in \overline{\mathcal{L}}} x_i y_j \left(a_{ij} - p_{ij}\right) \right| \right).
\end{aligned}$$

Bounding the light part Denote $\boldsymbol{W} \coloneqq \boldsymbol{A} - \boldsymbol{P}$ and $u_{ij} \coloneqq x_i y_j \mathbb{1}_{\left(|x_i y_j| \le \sqrt{\xi/n^d}\right)}$ for $i \in [n], j \in [n^{d-1}]$. The light part of $\boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{y}$ is given by

$$\sum_{i \in [n], j \in [n^{d-1}]} w_{ij} u_{ij}.$$

Since $|u_{ij}| \le \sqrt{\xi/n^d}$, the term $w_{ij} u_{ij}$ is of mean zero and bounded in absolute value by $\sqrt{\xi/n^d}$. By Bernstein's inequality (Vershynin, 2018, Theorem 2.8.4), we have

$$\begin{aligned}
\Pr\left[\left| \sum_{i,j} w_{ij} u_{ij} \right| \ge c \sqrt{\frac{\xi}{n^{d-2}}} \right] &\le 2 \exp\left(\frac{-\frac{1}{2} c^2 \xi / n^{d-2}}{\sum_{i,j} p_{ij}\left(1 - p_{ij}\right) u_{ij}^2 + \frac{1}{3} \sqrt{\frac{\xi}{n^d}} c \sqrt{\frac{\xi}{n^{d-2}}}} \right) \\
&\le 2 \exp\left(\frac{-\frac{1}{2} c^2 \xi / n^{d-2}}{p_{\max} \sum_{i,j} u_{ij}^2 + \frac{c\xi}{3n^{d-1}}} \right) \\
&\le 2 \exp\left(\frac{-3c^2 \xi / n^{d-2}}{6\xi / n^{d-1} + \frac{2c\xi}{n^{d-1}}} \right) \le 2 \exp\left(\frac{-3c^2}{6 + 2c} n \right).
\end{aligned}$$

The third inequality follows from the facts that $\xi \ge n^{d-1} p_{\max}$ and

$$\sum_{i,j} u_{ij}^2 \le \sum_{i,j} x_i^2 y_j^2 = \|\boldsymbol{x}\|_2^2 \|\boldsymbol{y}\|_2^2 \le 1.$$

Applying the union bound and the volume bound $|T| \le e^{n \log(7/\delta)}$ (Lei & Rinaldo, 2015), we obtain

$$\Pr\left[\max_{\boldsymbol{x} \in T} \left| \sum_{(i,j) \in \mathcal{L}(\boldsymbol{x}, \boldsymbol{y})} x_i y_j w_{ij} \right| \ge c \sqrt{\frac{\xi}{n^{d-2}}} \right] \le 2 \exp\left(- \left(\frac{3c^2}{6 + 2c} - \log\left(\frac{7}{\delta}\right) \right) n \right). \tag{43}$$

Bounding the heavy part The more challenging part of the proof is to show that the heavy part

$$\max_{\boldsymbol{x}\in T}\left|\sum_{(i,j)\in\overline{\mathcal{L}}(\boldsymbol{x},\boldsymbol{y})} x_i y_j w_{ij}\right|$$

is upper bounded by $c\sqrt{\frac{\xi}{n^{d-2}}}$ with high probability for some universal constant c. Observe that the expectation of $\sum_{(i,j)\in\overline{\mathcal{L}}(\boldsymbol{x},\boldsymbol{y})} x_i y_j a_{ij}$ can be well controlled:

$$\begin{aligned}\left|\sum_{(i,j)\in\overline{\mathcal{L}}(\boldsymbol{x},\boldsymbol{y})} x_i y_j p_{ij}\right| &= \left|\sum_{(i,j)\in\overline{\mathcal{L}}(\boldsymbol{x},\boldsymbol{y})} \frac{x_i^2 y_j^2}{x_i y_j} p_{ij}\right| \le \sum_{(i,j)\in\overline{\mathcal{L}}(\boldsymbol{x},\boldsymbol{y})} \frac{x_i^2 y_j^2}{|x_i y_j|} p_{ij} \\ &\le p_{\max}\sqrt{\frac{n^d}{\xi}} \sum_{(i,j)\in\overline{\mathcal{L}}(\boldsymbol{x},\boldsymbol{y})} x_i^2 y_j^2 \le p_{\max}\sqrt{\frac{n^d}{\xi}} = \sqrt{\frac{\xi}{n^{d-2}}},\end{aligned}$$

where the second inequality comes from the definition of heavy pairs. Then, it suffices to show that

$$\left|\sum_{(i,j)\in\overline{\mathcal{L}}(\boldsymbol{x},\boldsymbol{y})} x_i y_j a_{ij}\right| = \mathcal{O}\left(\sqrt{\frac{\xi}{n^{d-2}}}\right) \tag{44}$$

with high probability.

We will focus on the heavy pairs (i,j) such that $x_i > 0, y_j > 0$ and denote

$$\overline{\mathcal{L}}_1 := \left\{(i,j)\in\overline{\mathcal{L}} : x_i > 0, y_j > 0\right\}.$$

The other three cases are similar. Notice that if $y_j \neq 0$, given the assumption that $\ell \le n$, we have $|y_j| = \frac{1}{\sqrt{n^{d-1}}}\sqrt{\frac{n}{\ell}} \ge \frac{\delta}{\sqrt{n^{d-1}}}$. In what follows, we use the following notation:

- $I_1 := \left\{i : \frac{\delta}{\sqrt{n}} \le x_i \le \frac{2\delta}{\sqrt{n}}\right\}, I_s := \left\{i : \frac{\delta}{\sqrt{n}}2^{s-1} < x_i \le \frac{\delta}{\sqrt{n}}2^s\right\}$ for $s = 2,3,\ldots,\left\lceil \log_2 \frac{\sqrt{n}}{\delta}\right\rceil$.
- $J_1 := \left\{j : \frac{\delta}{\sqrt{n^{d-1}}} \le y_j \le \frac{2\delta}{\sqrt{n^{d-1}}}\right\}, J_t := \left\{j : \frac{\delta}{\sqrt{n^{d-1}}}2^{t-1} < y_j \le \frac{\delta}{\sqrt{n^{d-1}}}2^t\right\}$ for $t = 2,3,\ldots,\left\lceil \log_2 \frac{\sqrt{n^{d-1}}}{\delta}\right\rceil$.
- $e(I,J) := \sum_{i\in I, j\in J} a_{ij}$.
- $\mu(I,J) := \mathbb{E}[e(I,J)],\ \bar{\mu}(I,J) := p_{\max}|I||J|$. For simplicity, we will use μ and $\bar{\mu}$ when we do not need to specify their dependence on I and J.
- $\lambda_{st} := e\left(I_s, J_t\right)/\bar{\mu}_{st}$, where $\bar{\mu}_{st} := \bar{\mu}\left(I_s, J_t\right)$.
- $\alpha_s := |I_s|\, 2^{2s}/n,\ \beta_t := |J_t|\, 2^{2t}/n^{d-1},\ \sigma_{st} := \lambda_{st}\sqrt{\xi}2^{-(s+t)}$.

The following two lemmas are important to the rest of the proof.

Lemma C.1 (Bounded degree)**.** *For $c > 0$, there exists a constant $c_1 = c_1(c)$ such that with probability at least $1 - n^{-c}$,*

$$\sum_{j\in[n^{d-1}]} a_{ij} \le c_1\xi \ \textit{for all}\ \ i \in [n]. \tag{45}$$

Proof. The result follows directly by applying Bernstein's inequality and the union bound. □

Lemma C.2 (Bounded discrepancy)**.** *For any $c > 0$, there exist constants $c_2 = c_2(c) > 1$ and $c_3 = c_3(c) > 1$ such that with probability at least $1 - 2n^{-c}$, for a fixed index set $J \subseteq [n^{d-1}]$ and any index set $I \subseteq [n]$ with $|I| \le |J|/n^{d-2}$, at least one of the following hold:*
1) $\frac{e(I,J)}{\bar{\mu}(I,J)} \le ec_2$;
2) $e(I,J)\log\frac{e(I,J)}{\bar{\mu}(I,J)} \le c_3\frac{|J|}{n^{d-2}}\log\frac{n^{d-1}}{|J|}$.

Proof. Suppose that the event in (45) holds. If $|J|/n^{d-2} \geq n/e$, then (45) implies that $\frac{e(I,J)}{\xi|I||J|/n^{d-1}} \leq \frac{|I|c_1\xi}{\xi|I|/e} \leq c_1 e$. If $|J|/n^{d-2} < n/e$, then by Zhou & Zhu (2021, Lemma 4.5), we have for any $\tau > 1$,

$$\begin{aligned}
\Pr[e(I,J) \geq \tau\bar{\mu}(I,J)] &\leq \Pr\left[\sum_{i\in I, j\in J}(a_{ij} - p_{ij}) \geq \tau\bar{\mu}(I,J) - \sum_{i\in I, j\in J} p_{ij}\right] \\
&\leq \Pr\left[\sum_{i\in I, j\in J} w_{ij} \geq (\tau-1)\bar{\mu}(I,J)\right] \\
&\leq \exp\left((\tau-1)\bar{\mu} - \tau\bar{\mu}\log\tau\right) \leq \exp\left[-\frac{1}{2}(\tau\log\tau)\bar{\mu}\right], \qquad (46)
\end{aligned}$$

where the last inequality holds when $\tau \geq 8$. For any given $c_3 > 0$, let $t(I,J)$ denote the unique value of t satisfying $t\log t = \frac{c_3|J|/n^{d-2}}{\bar{\mu}(I,J)}\log\frac{n^{d-1}}{|J|}$. Let $\tau(I,J) := \max\{8, t(I,J)\}$. Then, by (46), we have

$$\Pr[e(I,J) \geq \tau(I,J)\bar{\mu}(I,J)] \leq \exp\left[-\frac{1}{2}\bar{\mu}(I,J)\tau(I,J)\log\tau(I,J)\right] \leq \exp\left[-\frac{1}{2}c_3\frac{|J|}{n^{d-2}}\log\frac{n^{d-1}}{|J|}\right].$$

Let $\Omega := \left\{(I,J) : |I| \leq g = \frac{|J|}{n^{d-2}} \leq \frac{n}{e}\right\}$. We bound

$$\begin{aligned}
\Pr\left[\exists (I,J)\in\Omega, e(I,J) \geq \tau(I,J)\bar{\mu}(I,J)\right] &\leq \sum_{(I,J)\in\Omega}\exp\left[-\frac{1}{2}c_3\frac{|J|}{n^{d-2}}\log\frac{n^{d-1}}{|J|}\right] \\
&\leq \sum_{h:1\leq h\leq g\leq n/e}\ \sum_{I:|I|=h}\exp\left[-\frac{1}{2}c_3 g\log\frac{n}{g}\right] \\
&\leq \sum_{h:1\leq h\leq g\leq n/e}\binom{n}{h}\exp\left[-\frac{1}{2}c_3 g\log\frac{n}{g}\right].
\end{aligned}$$

Since $\binom{n}{h} \leq (\frac{ne}{h})^h$ for any integer $1 \leq h \leq n$, the last line above is bounded by

$$\begin{aligned}
\sum_{h:1\leq h\leq g\leq n/e}\left(\frac{ne}{h}\right)^h\exp\left[-\frac{1}{2}c_3 g\log\frac{n}{g}\right] &= \sum_{h:1\leq h\leq g\leq n/e}\exp\left[-\frac{1}{2}c_3 g\log\frac{n}{g} + h\log\frac{n}{h} + h\right] \\
&\leq \sum_{h:1\leq h\leq g\leq n/e}\exp\left[-\frac{1}{2}c_3 g\log\frac{n}{g} + g\log\frac{n}{g} + g\right] \\
&\leq \sum_{h:1\leq h\leq g\leq n/e}\exp\left[-\frac{1}{2}(c_3-4)g\log\frac{n}{g}\right] \\
&\leq \sum_{h:1\leq h\leq g\leq n/e} n^{-\frac{1}{2}(c_3-4)} \leq n^{-\frac{1}{2}(c_3-6)},
\end{aligned}$$

where the inequalities repeatedly use the assumption that $h \leq g \leq n/e$ and the fact that $t \mapsto t\log\frac{n}{t}$ is increasing on $[1, n/e]$.

Hence, with probability at least $1 - n^{-\frac{1}{2}(c_3-6)}$, we know $e(I,J) \leq \tau(I,J)\bar{\mu}(I,J)$ for all $|I| \leq |J|/n^{d-2} \leq n/e$. We further divide the set of pairs (I,J) satisfying $|I| \leq |J|/n^{d-2} \leq n/e$ into two groups by the value of $\tau(I,J)$. For the pairs satisfying $\tau(I,J) = 8$, we get

$$e(I,J) \leq \tau(I,J)\bar{\mu}(I,J) = 8\bar{\mu}(I,J).$$

For all the other pairs, we have $\tau(I,J) = t(I,J) > 8$ and $\frac{e(I,J)}{\bar{\mu}(I,J)} \leq t(I,J)$. It follows that

$$\frac{e(I,J)}{\bar{\mu}(I,J)}\log\frac{e(I,J)}{\bar{\mu}(I,J)} \leq t(I,J)\log t(I,J) = \frac{c_3|J|/n^{d-2}}{\bar{\mu}(I,J)}\log\frac{n^{d-1}}{|J|},$$

which implies that

$$e(I,J)\log\frac{e(I,J)}{\bar{\mu}(I,J)} \le c_3\frac{|J|}{n^{d-2}}\log\frac{n^{d-1}}{|J|}.$$

The desired result follows by letting $c_2 = \max\{c_1, 8\}$ and $c_3 = 2c+6$. □

Now, we write the left-hand side of (44) as

$$\sum_{(i,j)\in\bar{\mathcal{L}}_1} x_i y_j a_{ij} \le \sum_{(s,t):\, 2^{s+t}\ge\sqrt{\xi}} e\left(I_s, J_t\right)\frac{2^s\delta}{\sqrt{n}}\frac{2^t\delta}{\sqrt{n^{d-1}}} = \delta^2\sqrt{\frac{\xi}{n^{d-2}}}\sum_{(s,t):\, 2^{s+t}\ge\sqrt{\xi}}\alpha_s\beta_t\sigma_{st}. \tag{47}$$

We estimate this sum by splitting the pairs of (s,t) into twelve different categories. Denote by $J_{t'}$ the nonempty set in $\left\{J_t : t = 1,\dots,\left\lceil\log_2\frac{\sqrt{n^{d-1}}}{\delta}\right\rceil\right\}$. Let

$$\mathcal{C} := \left\{s : 2^{s+t'} \ge \sqrt{\xi}, |I_s| \le |J_{t'}|/n^{d-2}\right\}$$

and define the following sets:

- $\mathcal{C}_1 := \{s\in\mathcal{C} : \sigma_{st'} \le 1\}$,
- $\mathcal{C}_2 := \{s\in\mathcal{C}\backslash\mathcal{C}_1 : \lambda_{st'} \le ec_2\}$,
- $\mathcal{C}_3 := \left\{s\in\mathcal{C}\backslash\left(\mathcal{C}_1\cup\mathcal{C}_2\right) : 2^s \ge \sqrt{\xi}2^{t'}\right\}$,
- $\mathcal{C}_4 := \left\{s\in\mathcal{C}\backslash\left(\mathcal{C}_1\cup\mathcal{C}_2\cup\mathcal{C}_3\right) : \log\lambda_{st'} > \frac{1}{4}\left(2t'\log 2 + \log(\beta_{t'}^{-1})\right)\right\}$,
- $\mathcal{C}_5 := \left\{s\in\mathcal{C}\backslash\left(\mathcal{C}_1\cup\mathcal{C}_2\cup\mathcal{C}_3\cup\mathcal{C}_4\right) : 2t'\log 2 \ge \log(\beta_{t'}^{-1})\right\}$,
- $\mathcal{C}_6 := \{s\in\mathcal{C}\backslash\left(\mathcal{C}_1\cup\mathcal{C}_2\cup\mathcal{C}_3\cup\mathcal{C}_4\cup\mathcal{C}_5\right)\}$.

The other six categories can be defined using a similar partition of

$$\mathcal{C}' := \left\{s : 2^{s+t'} \ge \sqrt{\xi}, |I_s| > |J_{t'}|/n^{d-2}\right\}$$

and can be analyzed similarly. We now analyze each of the six cases separately. To that end, we will repeatedly make use of the following estimates:

$$\sum_s \alpha_s \le \sum_i |2x_i/\delta|^2 \le 4\delta^{-2}, \beta_{t'} \le l\cdot n^{d-2}2^{2t'}/n^{d-1}, \text{ and } \beta_t = 0 \text{ for } t\ne t'.$$

In addition, we have $\beta_{t'} = \sum_t \beta_t \le 4\delta^{-2}$.

Indices in $\mathcal{C}_1$: Since $\sigma_{st'} \le 1$, we have

$$\sum_s \alpha_s\beta_{t'}\sigma_{st'}\mathbb{1}_{(s\in\mathcal{C}_1)} \le 16\delta^{-4}. \tag{48}$$

Indices in $\mathcal{C}_2$: By definition, it holds that $\sigma_{st'} = \lambda_{st'}\sqrt{\xi}2^{-(s+t')} \le \lambda_{st'} \le ec_2$. Hence,

$$\sum_s \alpha_s\beta_{t'}\sigma_{st'}\mathbb{1}_{(s\in\mathcal{C}_2)} \le ec_2 16\delta^{-4}. \tag{49}$$

Indices in $\mathcal{C}_3$: We know from Lemma C.1 that $e(I_s, J_{t'}) \le c_1|I_s|\xi$ holds with probability at least $1-n^{-c}$. It follows that $\lambda_{st'} \le c_1 n^{d-1}/|J_{t'}|$ and

$$\sum_s \alpha_s\beta_{t'}\sigma_{st'}\mathbb{1}_{(s\in\mathcal{C}_3)} \le \sum_s \alpha_s\frac{|J_{t'}|2^{2t'}}{n^{d-1}}\frac{c_1 n^{d-1}}{|J_{t'}|}\sqrt{\xi}2^{-(s+t')}\mathbb{1}_{(s\in\mathcal{C}_3)} \le c_1\sum_s \alpha_s\frac{\sqrt{\xi}}{2^{s-t'}}\mathbb{1}_{(s\in\mathcal{C}_3)} \le 4c_1\delta^{-2},$$

where the last inequality comes from $2^{s-t'} \geq \sqrt{\xi}$.

To cope with $\mathcal{C}_4, \mathcal{C}_5$, and $\mathcal{C}_6$, we rely on the second case in Lemma C.2, which is equivalent to

$$\begin{aligned} & \lambda_{st'} |I_s||J_{t'}| \frac{\xi}{n^{d-1}} \log \lambda_{st'} \leq c_3 \frac{|J_{t'}|}{n^{d-2}} \log \frac{n^{d-1}}{|J_{t'}|} \\ \iff & \sigma_{st'} \alpha_s \log \lambda_{st'} \leq c_3 \frac{2^{s-t'}}{\sqrt{\xi}} \log \frac{n^{d-1}}{|J_{t'}|} \\ \iff & \sigma_{st'} \alpha_s \log \lambda_{st'} \leq c_3 \frac{2^{s-t'}}{\sqrt{\xi}} \big(2t' \log 2 + \log(\beta_{t'}^{-1})\big). \end{aligned} \tag{50}$$

Indices in $\mathcal{C}_4$: The inequality $\log \lambda_{st'} > \frac{1}{4}\big(2t' \log 2 + \log(\beta_{t'}^{-1})\big)$ and (50) imply that $\sigma_{st'}\alpha_s \leq 4c_3 2^{s-t'}/\sqrt{\xi}$. Then, we have

$$\sum_s \alpha_s \beta_{t'} \sigma_{st'} \mathbb{1}_{(s \in \mathcal{C}_4)} = \beta_{t'} \sum_s \alpha_s \sigma_{st'} \mathbb{1}_{(s \in \mathcal{C}_4)} \leq 8c_3 \beta_{t'} \leq 32 c_3 \delta^{-2},$$

where the first inequality is from $s \notin \mathcal{C}_3$.

Indices in $\mathcal{C}_5$: In this case, we have $2t' \log 2 \geq \log(\beta_{t'}^{-1})$. Also, since $s \notin \mathcal{C}_4$, we have $\log \lambda_{st'} \leq \frac{1}{4}\big(2t' \log 2 + \log(\beta_{t'}^{-1})\big) \leq t' \log 2$. Thus, $\lambda_{st'} \leq 2^{t'}$. Besides, since $s \notin \mathcal{C}_1$, we have $1 \leq \sigma_{st'} = \lambda_{st'}\sqrt{\xi} 2^{-(s+t')} \leq \sqrt{\xi} 2^{-s}$. It follows that $2^s \leq \sqrt{\xi}$.

Since $s \notin \mathcal{C}_2$, we have $\log \lambda_{st'} \geq 1$. Together with $2t' \log 2 \geq \log \beta_{t'}^{-1}$, (50) implies that

$$\sigma_{st'} \alpha_s \leq c_3 \frac{2^{s-t'}}{\sqrt{\xi}} 4t' \log 2.$$

Hence,

$$\begin{aligned} \sum_s \alpha_s \beta_{t'} \sigma_{st'} \mathbb{1}_{(s \in \mathcal{C}_5)} &= \beta_{t'} \sum_s \alpha_s \sigma_{st'} \mathbb{1}_{(s \in \mathcal{C}_5)} \leq \beta_{t'} \sum_s c_3 \frac{2^{s-t'}}{\sqrt{\xi}} 4t' (\log 2) \mathbb{1}_{(s \in \mathcal{C}_5)} \\ &\leq 4c_3 (\log 2) \beta_{t'} t' 2^{-t'} \sum_s \frac{2^s}{\sqrt{\xi}} \mathbb{1}_{(s \in \mathcal{C}_5)} \leq 4c_3 (\log 2) \beta_{t'} \leq 16 c_3 \delta^{-2}. \end{aligned}$$

Indices in $\mathcal{C}_6$: In this case, we have $2t' \log 2 < \log(\beta_{t'}^{-1})$. Since $s \notin \mathcal{C}_2 \cup \mathcal{C}_4$, we have $\log \lambda_{st'} \geq 1$ and $\log \lambda_{st'} \leq \frac{1}{2} \log \beta_{t'}^{-1} \leq \log \beta_{t'}^{-1}$. Now, we compute

$$\begin{aligned} \sum_s \alpha_s \beta_{t'} \sigma_{st'} \mathbb{1}_{(s \in \mathcal{C}_6)} &= \sum_s \alpha_s \beta_{t'} \lambda_{st'} \sqrt{\xi} 2^{-(s+t')} \mathbb{1}_{(s \in \mathcal{C}_6)} \\ &\leq \sum_s \alpha_s \sqrt{\xi} 2^{-(s+t')} \mathbb{1}_{(s \in \mathcal{C}_6)} \leq \sum_s \alpha_s \leq 4\delta^{-2}. \end{aligned}$$

The proof is complete. □

D. Proof of Remark 4.4

Proof. Let $\mathcal{R} \in S^d(\mathbb{R}^n)$ be a tensor, whose entries are given by

$$\mathcal{R}_{i_1, i_2, \ldots, i_d} = \begin{cases} p, & \text{if there exist } i_j, i_k \text{ such that } i_j = i_k \text{ and } i_1, \ldots, i_d \text{ belong to the same community;} \\ q, & \text{if there exist } i_j, i_k \text{ such that } i_j = i_k \text{ and } i_1, \ldots, i_d \text{ belong to different communities;} \\ 0, & \text{otherwise.} \end{cases}$$

Then, the expectation of the adjacency tensor defined in Definition 3.1 can be written as

$$\mathbb{E}[\mathcal{A}] = q \cdot \big(\mathbf{1}_n^{\otimes d}\big) + (p - q) \cdot (\boldsymbol{H}^*)^{\otimes d} - \mathcal{R}. \tag{51}$$

Following the argument in (28), it remains to show that the effect brought by the multilinear operator of the tensor $\mathcal{R}$ on the estimation error is negligible. Indeed, consider

$$\begin{aligned}\left\|\mathcal{R}\left[\boldsymbol{H}^{\otimes(d-1)}\right]-\mathcal{R}\left[(\boldsymbol{H}^*\boldsymbol{Q})^{\otimes(d-1)}\right]\right\|_F^2 &= \left\|\mathcal{M}(\mathcal{R})\left(\boldsymbol{H}^{\odot(d-1)}-(\boldsymbol{H}^*\boldsymbol{Q})^{\odot(d-1)}\right)\right\|_F^2 \\ &\le \|\mathcal{M}(\mathcal{R})\|_F^2\left\|\boldsymbol{H}^{\odot(d-1)}-(\boldsymbol{H}^*\boldsymbol{Q})^{\odot(d-1)}\right\|_F^2. \end{aligned} \tag{52}$$

Then, on the one hand, since the number of nonzero elements of $\mathcal{R}$ is at most $n\binom{d}{2}n^{d-2}$, we can estimate $\|\mathcal{M}(\mathcal{R})\|_F^2$ by

$$\|\mathcal{M}(\mathcal{R})\|_F^2 \le p^2\binom{d}{2}n^{d-1}.$$

On the other hand, upon applying (34), we see that

$$\left\|\boldsymbol{H}^{\odot(d-1)}-(\boldsymbol{H}^*\boldsymbol{Q})^{\odot(d-1)}\right\|_F^2=\sum_{k=1}^K\left\|\boldsymbol{H}_k^{\otimes(d-1)}-(\boldsymbol{H}^*\boldsymbol{Q})_k^{\otimes(d-1)}\right\|_F^2 \le (d-1)m^{d-2}\|\boldsymbol{H}-\boldsymbol{H}^*\boldsymbol{Q}\|_F^2.$$

Combining the above two facts, (52) then implies that

$$\left\|\mathcal{R}\left[\boldsymbol{H}^{\otimes(d-1)}\right]-\mathcal{R}\left[(\boldsymbol{H}^*\boldsymbol{Q})^{\otimes(d-1)}\right]\right\|_F \le \mathcal{O}(pn^{d-3/2})\cdot\|\boldsymbol{H}-\boldsymbol{H}^*\boldsymbol{Q}\|_F,$$

which is dominated by $\mathcal{O}((p-q)n^{d-1})\cdot\|\boldsymbol{H}-\boldsymbol{H}^*\boldsymbol{Q}\|_F$ (i.e., the first term on the right-hand side of (13)). □

E. Proof of Lemma 4.7

Proof. Since $\mathcal{A}$ is generated according to the d-HSBM, we know that for all $i\in\mathcal{I}_k$ with $\ell\neq k$,

$$C_{ik}-C_{i\ell}=\sum_{1\le i_2,\dots,i_d\le n}\mathcal{A}_{i,i_2,\dots,i_d}H^*_{i_2k}\dots H^*_{i_dk}-\sum_{1\le i_2,\dots,i_d\le n}\mathcal{A}_{i,i_2,\cdots,i_d}H^*_{i_2\ell}\cdots H^*_{i_d\ell} \tag{53}$$

$$\overset{\mathrm{d}}{=}(d-1)!\cdot\sum_{i=1}^{\binom{m-1}{d-1}}W_i-(d-1)!\cdot\sum_{i=1}^{\binom{m}{d-1}}Z_i, \tag{54}$$

where $\{W_i\}$ are i.i.d. $\mathbf{Bern}(\alpha\log n/n^{d-1})$ and $\{Z_i\}$ are i.i.d. $\mathbf{Bern}(\beta\log n/n^{d-1})$ that are independent of $\{W_i\}$. By Lemma 4.6 and the fact that n is sufficiently large, we have

$$\Pr\left[C_{ik}-C_{i\ell}\ge\gamma\log n,\forall i\in\mathcal{I}_k,1\le k\neq\ell\le K\right]\ge 1-Kn^{-c_2/2} \tag{55}$$

with

$$c_2=\frac{(\sqrt{\alpha}-\sqrt{\beta})^2}{K^{d-1}(d-1)!}-\frac{\gamma\log(\alpha/\beta)}{2(d-1)!}-1>0. \tag{56}$$

This completes the proof. □

F. Proof of Proposition 4.8

Proof. Suppose that $\boldsymbol{V}\in\mathcal{T}\left(\mathcal{A}[\boldsymbol{H}^{\otimes(d-1)}]\right)$. From Wang et al. (2021a, Lemma 9), it follows that

$$\boldsymbol{V}\boldsymbol{Q}^\top\in\mathcal{T}\left(\mathcal{A}[\boldsymbol{H}^{\otimes(d-1)}]\boldsymbol{Q}^\top\right)$$

for any $\boldsymbol{Q}\in\Pi_K$. In addition, by Lemma 4.2, Lemma 4.5, and the fact that

$$\mathcal{A}[(\boldsymbol{H}^*)^{\otimes(d-1)}]\boldsymbol{Q}=\mathcal{A}[(\boldsymbol{H}^*\boldsymbol{Q})^{\otimes(d-1)}],\ \boldsymbol{Q}\in\Pi_K,$$

we have

$$\begin{aligned}\|\boldsymbol{V}-\boldsymbol{H}^*\boldsymbol{Q}\|_F = \|\boldsymbol{V}\boldsymbol{Q}^\top-\boldsymbol{H}^*\|_F &\le \frac{2\left\|\mathcal{A}[\boldsymbol{H}^{\otimes(d-1)}]\boldsymbol{Q}^\top-\mathcal{A}[(\boldsymbol{H}^*)^{\otimes(d-1)}]\right\|_F}{\gamma\log n}\\ &= \frac{2\left\|\mathcal{A}[\boldsymbol{H}^{\otimes(d-1)}]-\mathcal{A}[(\boldsymbol{H}^*\boldsymbol{Q})^{\otimes(d-1)}]\right\|_F}{\gamma\log n}\\ &\le \left(\frac{8(d-1)m^{d-2}\varepsilon n}{\sqrt{K}}(p-q)+2C\sqrt{\log n}\right)\|\boldsymbol{H}-\boldsymbol{H}^*\boldsymbol{Q}\|_F/(\gamma\log n)\\ &\le \left(\frac{8(d-1)\varepsilon(\alpha-\beta)}{\gamma K^{d-3/2}}+\frac{2C}{\gamma\sqrt{\log n}}\right)\|\boldsymbol{H}-\boldsymbol{H}^*\boldsymbol{Q}\|_F\\ &\le 4\max\left\{\frac{4(d-1)\varepsilon(\alpha-\beta)}{\gamma K^{d-3/2}},\frac{C}{\gamma\sqrt{\log n}}\right\}\|\boldsymbol{H}-\boldsymbol{H}^*\boldsymbol{Q}\|_F.\end{aligned}$$

This completes the proof. □

G. Proof of Theorem 4.9

Proof. The iteration $\boldsymbol{H}^1\in\mathcal{T}(\boldsymbol{H}^0)$ yields

$$\|\boldsymbol{H}^1-\boldsymbol{H}^*\boldsymbol{Q}\|_F=\|\boldsymbol{H}^1\boldsymbol{Q}^\top-\boldsymbol{H}^*\|_F\le 2\|\boldsymbol{H}^0\boldsymbol{Q}^\top-\boldsymbol{H}^*\|_F=2\|\boldsymbol{H}^0-\boldsymbol{H}^*\boldsymbol{Q}\|_F, \tag{57}$$

where the inequality comes from Lemma 4.5. The following proof is divided into two parts. We first show that for all $t\ge 2$, $\boldsymbol{H}^t\in\mathcal{H}$ satisfies

$$\|\boldsymbol{H}^t-\boldsymbol{H}^*\boldsymbol{Q}\|_F\le\frac{1}{2}\|\boldsymbol{H}^{t-1}-\boldsymbol{H}^*\boldsymbol{Q}\|_F \quad\text{and}\quad \|\boldsymbol{H}^t-\boldsymbol{H}^*\boldsymbol{Q}\|_F\le 2\theta\sqrt{n}, \tag{58}$$

and estimate the iteration number N_1 such that

$$\|\boldsymbol{H}^{N_1}-\boldsymbol{H}^*\boldsymbol{Q}\|_F\le 2\phi\sqrt{\frac{n}{\log n}}. \tag{59}$$

Suppose that $\boldsymbol{H}^0\in\mathbb{M}_{n,K}$ satisfies (11). Combining this initialization condition with (57) gives

$$\boldsymbol{H}^1\in\mathcal{H} \quad\text{and}\quad \|\boldsymbol{H}^1-\boldsymbol{H}^*\boldsymbol{Q}\|_F\le 2\theta\sqrt{n}. \tag{60}$$

According to Proposition 4.8, we have

$$\begin{aligned}\|\boldsymbol{H}^2-\boldsymbol{H}^*\boldsymbol{Q}\|_F &\le 4\max\left\{\frac{4(d-1)\theta(\alpha-\beta)}{\gamma K^{d-3/2}},\frac{C}{\gamma\sqrt{\log n}}\right\}\|\boldsymbol{H}^1-\boldsymbol{H}^*\boldsymbol{Q}\|_F\\ &\le 4\max\left\{\frac{1}{8},\frac{C}{\gamma\sqrt{\log n}}\right\}\|\boldsymbol{H}^1-\boldsymbol{H}^*\boldsymbol{Q}\|_F=\frac{1}{2}\|\boldsymbol{H}^1-\boldsymbol{H}^*\boldsymbol{Q}\|_F\le 2\theta\sqrt{n},\end{aligned}$$

where the equality comes from $n\ge\exp(64C^2/\gamma^2)$. Therefore, (58) holds for $t=2$. By a simple inductive argument, we can show that (58) holds for $t\ge 3$. Let $N_1:=\lceil 2\log\log n\rceil+1$. It then follows from (58) that

$$\begin{aligned}\|\boldsymbol{H}^{N_1}-\boldsymbol{H}^*\boldsymbol{Q}\|_F &\le\left(\frac{1}{2}\right)^{\lceil 2\log\log n\rceil}\|\boldsymbol{H}^1-\boldsymbol{H}^*\boldsymbol{Q}\|_F\le\left(\frac{1}{2}\right)^{2\log\log n}2\theta\sqrt{n}\le\left(\frac{1}{2}\right)^{\log\log n+2\log\left(\frac{\theta}{\phi}\right)}2\theta\sqrt{n}\\ &\le\left(\frac{1}{2}\right)^{\frac{\log\log n+2\log\left(\frac{\theta}{\phi}\right)}{2\log 2}}2\theta\sqrt{n}=2\phi\sqrt{\frac{n}{\log n}},\end{aligned}$$

where the third inequality is from $n\ge\exp\left(\gamma^2/C^2\right)\ge\exp\left(\theta^2/\phi^2\right)$ and the last inequality comes from $2\log 2\ge 1$. Thus, (59) holds for $N_1=\lceil 2\log\log n\rceil+1$.

Next, we show that for all $k \geq 1$, $\boldsymbol{H}^{N_1+k} \in \mathcal{H}$ satisfies $\|\boldsymbol{H}^{N_1+k} - \boldsymbol{H}^*\boldsymbol{Q}\|_F \leq 2\phi\sqrt{n/\log n}$ and

$$\|\boldsymbol{H}^{N_1+k} - \boldsymbol{H}^*\boldsymbol{Q}\|_F \leq \frac{4C}{\gamma\sqrt{\log n}}\|\boldsymbol{H}^{N_1+k-1} - \boldsymbol{H}^*\boldsymbol{Q}\|_F, \tag{61}$$

and compute the iteration number N_2 such that

$$\|\boldsymbol{H}^{N_2+N_1} - \boldsymbol{H}^*\boldsymbol{Q}\|_F < \sqrt{2}. \tag{62}$$

Since $n \geq \exp\left(\phi^2/\theta^2\right)$, it follows that $2\phi/\sqrt{\log n} \leq 4\phi/\sqrt{\log n} \leq 4\theta$. According to Proposition 4.8, we obtain

$$\|\boldsymbol{H}^{N_1+1} - \boldsymbol{H}^*\boldsymbol{Q}\|_F \leq 4\max\left\{\frac{8(d-1)\phi(\alpha-\beta)}{\gamma K^{d-3/2}\sqrt{\log n}}, \frac{C}{\gamma\sqrt{\log n}}\right\}\|\boldsymbol{H}^{N_1} - \boldsymbol{H}^*\boldsymbol{Q}\|_F \leq \frac{4C}{\gamma\sqrt{\log n}}\|\boldsymbol{H}^{N_1} - \boldsymbol{H}^*\boldsymbol{Q}\|_F.$$

Then, (61) holds for $k = 1$. We can show that (61) holds for $k \geq 2$ by a simple inductive argument. Let $N_2 := \left\lceil \frac{2\log n}{\log\log n} \right\rceil$. According to $n \geq \exp(256C^4/\gamma^4)$ and $n \geq \exp\left(2\phi^2\right)$, we have $\log\log n \geq 4\log(4C/\gamma)$ and $2\phi/\sqrt{\log n} < \sqrt{2}$. This, together with (61), yields

$$\begin{aligned}\|\boldsymbol{H}^{N_1+N_2} - \boldsymbol{H}^*\boldsymbol{Q}\|_F &\leq \left(\frac{4C}{\gamma\sqrt{\log n}}\right)^{\left\lceil \frac{2\log n}{\log\log n}\right\rceil}\|\boldsymbol{H}^{N_1} - \boldsymbol{H}^*\boldsymbol{Q}\|_F \leq 2\phi\sqrt{\frac{n}{\log n}}\left(\frac{4C}{\gamma\sqrt{\log n}}\right)^{\frac{2\log n}{\log\log n}} \\ &\leq 2\phi\sqrt{\frac{n}{\log n}}\left(\frac{4C}{\gamma\sqrt{\log n}}\right)^{\frac{\log n}{\log\log n + 2\log(\gamma/(4C))}} = \frac{2\phi}{\sqrt{\log n}} < \sqrt{2}.\end{aligned}$$

Thus, (62) holds for $N_2 = \left\lceil \frac{2\log n}{\log\log n} \right\rceil$.

Once (62) holds, we have $\boldsymbol{H}^{N_1+N_2} = \boldsymbol{H}^*\boldsymbol{Q}$. This, together with $\mathcal{T}\left(\mathcal{A}[(\boldsymbol{H}^*\boldsymbol{Q})^{\otimes(d-1)}]\right) = \{\boldsymbol{H}^*\boldsymbol{Q}\}$, gives the desired result. □