Optimal Non-Convex Exact Recovery in Stochastic Block Model via Projected Power Method

Peng Wang [1] **Huikang Liu** [2] **Zirui Zhou** [3] **Anthony Man-Cho So** [1]

Abstract

In this paper, we study the problem of exact community recovery in the symmetric stochastic block model, where a graph of n vertices is randomly generated by partitioning the vertices into $K \geq 2$ equal-sized communities and then connecting each pair of vertices with probability that depends on their community memberships. Although the maximum-likelihood formulation of this problem is discrete and non-convex, we propose to tackle it directly using projected power iterations with an initialization that satisfies a partial recovery condition. Such an initialization can be obtained by a host of existing methods. We show that in the logarithmic degree regime of the considered problem, the proposed method can exactly recover the underlying communities at the information-theoretic limit. Moreover, with a qualified initialization, it runs in $\mathcal{O}(n \log^2 n/ \log\log n)$ time, which is competitive with existing state-of-the-art methods. We also present numerical results of the proposed method to support and complement our theoretical development.

1. Introduction

Community detection is a fundamental task in network analysis and has found wide applications in diverse fields, such as social science (Girvan & Newman, 2002), physics (Newman & Girvan, 2004), and machine learning (Shi & Malik, 2000), just to name a few. As the study of community detection grows, a large variety of theories and algorithms have been proposed in the past decades for addressing different tasks under different settings. To better validate and compare these theories and algorithms, the stochastic block model (SBM), which tends to generate graphs containing underlying community structures, is widely used as a canonical model for studying community detection. In particular, substantial advances have been made in recent years on understanding the fundamental limits of community detection and developing algorithms for tackling different recovery tasks in the SBM; see, e.g., Abbe (2017) and the references therein.

In this work, we consider the problem of exactly recovering the communities in the symmetric SBM. Specifically, given n nodes that are partitioned into $K \geq 2$ unknown communities of equal size, a random graph is generated by independently connecting each pair of vertices with probability p if they belong to the same community and with probability q otherwise. The goal is to recover the underlying communities exactly by only observing one realization of the graph. In the logarithmic degree regime of the considered SBM, i.e., $p = \alpha \log n/n$ and $q = \beta \log n/n$ for some $\alpha, \beta > 0$, this problem exhibits a sharp information-theoretic threshold: it is possible to achieve exact recovery if $\sqrt{\alpha}-\sqrt{\beta} > \sqrt{K}$ and is impossible if $\sqrt{\alpha} - \sqrt{\beta} < \sqrt{K}$ (Abbe & Sandon, 2015). Then, it is of interest to design computationally tractable methods that can achieve exact recovery under a condition on α and β that meets the information-theoretic limit. In the past years, many algorithms have been proposed to achieve this task, such as spectral clustering (McSherry, 2001; Su et al., 2019; Yun & Proutiere, 2014; 2016), SDP-based approach (Amini et al., 2018; Fei & Chen, 2018; 2020; Li et al., 2018), and likelihood-based approach (Amini et al., 2013; Gao et al., 2017; Zhang & Zhou, 2016; Zhou & Li, 2020). However, most of these algorithms have a time complexity that is at least quadratic in n, which usually does not scale well to large-scale problems.

In the symmetric SBM, the maximum likelihood (ML) estimation problem is formulated as

$$\max \left\{ \langle \boldsymbol{A}\boldsymbol{H}, \boldsymbol{H} \rangle : \boldsymbol{H} \in \mathcal{H} \right\}, \qquad \text{(MLE)}$$

where $\boldsymbol{A}$ is the adjacency matrix of the observed graph,

$$\mathcal{H} = \big\{ \boldsymbol{H} \in \mathbb{R}^{n\times K} : \ \boldsymbol{H}\mathbf{1}_K = \mathbf{1}_n, \ \boldsymbol{H}^T\mathbf{1}_n = m\mathbf{1}_K, \ \boldsymbol{H} \in \{0,1\}^{n\times K} \big\} \quad (1)$$

[1]Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Hong Kong [2]Business School, Imperial College London, London, United Kingdom [3]Huawei Technologies Canada Co., Ltd., Burnaby, Canada. Correspondence to: Huikang Liu <hkliu2014@gmail.com>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

is the discrete feasible set, $\mathbf{1}_n$ is the all-one vector of dimension n, and $m = n/K$. It is known that an ML estimator achieves exact recovery at the information-theoretic limit, but solving Problem (MLE) is NP-hard in the worst-case. Recently, in independent lines of research, many non-convex formulations that arise in a variety of applications have been shown to be solvable, in the sense of average-case performance, by simple and scalable iterative methods. This includes phase retrieval (Bendory et al., 2017; Chen et al., 2019), group synchronization (Ling, 2020; Liu et al., 2017a; 2020; Zhong & Boumal, 2018), low-rank matrix recovery (Chi et al., 2019), and two-block community detection (Wang et al., 2020). It then naturally motivates the question of whether one can apply a similar simple and scalable method to the discrete optimization problem (MLE). In this work, we answer this question in the affirmative by showing that a projected power method provably works for solving Problem (MLE). As a consequence, we obtain a simple and scalable method that achieves exact recovery under the optimal condition on α and β.

1.1. Related Works

In the context of the SBM, *exact recovery*, also named *strong consistency*, requires all the communities to be identified correctly up to a permutation of labels. More precisely, exact recovery is achieved if there exists an algorithm that takes one realization of the graph as input and outputs the true partition with high probability. In the logarithmic degree regime of the binary symmetric SBM, i.e., the symmetric SBM with $K = 2$, Abbe et al. (2016) and Mossel et al. (2014) independently showed that it is possible to achieve exact recovery if $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{2}$ and is not possible if $\sqrt{\alpha} - \sqrt{\beta} < \sqrt{2}$, thereby establishing the information-theoretic limit for exact recovery. Later, Abbe & Sandon (2015) generalized this result to the case of $K \geq 2$ and showed that the information-theoretic limit is $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{K}$. *Almost exact recovery*, also named *weak consistency*, requires the recovery of all but a vanishing fraction of vertices. In *partial recovery*, only a constant fraction of vertices needs to be identified correctly. It is obvious that the requirement of partial recovery is much milder than that of almost exact recovery. We refer the reader to Abbe (2017) for the formal definitions of these recovery tasks and more results on the corresponding fundamental limits in the SBM.

Over the past years, many algorithms have been proposed to tackle the problem of exact recovery in the symmetric SBM. One popular approach is spectral clustering. For example, McSherry (2001) proposed a spectral partition method, which first randomly partitions the vertex set into two parts, then calls the combinatorial projection subroutine, and finally clusters the vertices by distances on the projected points. They showed that in the symmetric SBM, the proposed method achieves exact recovery if $(p-q)/\sqrt{p} \gtrsim \sqrt{\log n/n}$ and $np \gtrsim \log^6 n$. Later, Yun & Proutiere (2014; 2016) also presented a spectral partition method, which proceeds by applying spectral decomposition to a trimmed adjacency matrix for generating an initial partition, followed by an additional procedure for local improvement. In the considered SBM, this method achieves exact recovery down to the information-theoretic threshold in $\mathcal{O}(n\mathbf{poly}\log n)$. Recently, Su et al. (2019) showed that the standard spectral clustering, which first computes the leading K eigenvectors of the graph Laplacian matrix and then applies the *k-means* algorithm to do clustering, achieves exact recovery under some weak conditions. These conditions can be simplified as $\sqrt{\alpha} - \sqrt{\beta} \geq c > \sqrt{K}$ for some positive constant c in the symmetric SBM. In general, these spectral clustering methods run in polynomial time. Another popular approach is convex relaxation of the ML estimation problem. In the setting of $K = 2$, Bandeira (2018) and Hajek et al. (2016a;b) respectively showed that semidefinite programming (SDP) relaxation of the ML formulation of the binary symmetric SBM achieves exact recovery at the information-theoretic limit. In the setting of $K \geq 2$, Guédon & Vershynin (2016) proposed a SDP relaxation of Problem (MLE) and showed a recovery error bound, which decays polynomially in the signal-to-noise ratio, for the solution to their considered SDP. Such an error bound only implies that their proposed SDP achieves almost exact recovery in our considered SBM. Following this work, Fei & Chen (2018; 2020) proposed a new SDP relaxation of Problem (MLE) and established a more refined recovery error bound, which decays exponentially in the signal-to-noise ratio. This error bound implies that their proposed SDP achieves exact recovery provided that $(\alpha-\beta)^2 \geq c\,(\alpha + (K-1)\beta)$ for a positive constant c. Besides, Amini et al. (2018) proposed another SDP relaxation of Problem (MLE) and showed that this SDP exactly recovers the communities with high probability if $(\alpha-\beta)^2 \geq cK(\alpha + K\beta)$ for a positive constant c. Despite the nice property of SDP-based approaches that they do not require any initial estimate of the partition or local refinement, solving the SDP problem is usually computationally prohibitive for large-scale data sets. We refer the reader to a survey by Li et al. (2018) for more results on convex relaxation methods for community detection.

We would also like to mention some algorithms for the considered problem that use other techniques. Abbe & Sandon (2015) developed a two-stage algorithm that consists of the Sphere-comparison sub-routine for detecting communities almost exactly and the Degree-profiling sub-routine for identifying the communities exactly. Moreover, it recovers the communities exactly with high probability all the way down to information-theoretic threshold in $\mathcal{O}(n^{1+1/\log\log n})$ time in the considered SBM. Besides, Gao et al. (2017) proposed a two-stage algorithm that needs a weakly consistent initialization and refines it by optimizing the local penalized maximum likelihood function for each node separately. In

Table 1. Comparison of recovery conditions and time complexities of the surveyed methods for exact recovery in the SBM ($K \geq 2$).

References	Conditions	Complexities
McSherry (2001)	Not optimal	Polynomial
Yun & Proutiere (2016)	Optimal	$\mathcal{O}(n\mathbf{poly}\log n)$
Su et al. (2019)	Not optimal	Polynomial
Amini et al. (2018)	Not optimal	Polynomial
Fei & Chen (2018)	Not optimal	Polynomial
Abbe & Sandon (2015)	Optimal	$\mathcal{O}(n^{1+1/\log\log n})$
Gao et al. (2017)	Optimal	Polynomial
Ours	**Optimal**	$\mathcal{O}\left(\frac{n\log^2 n}{\log\log n}\right)$

the considered SBM, their proposed method achieves exact recovery at the information-theoretic limit in polynomial time. We refer the reader to Amini et al. (2013); Zhang & Zhou (2016); Zhou & Li (2020) for more likelihood-based approach. Recently, Wang et al. (2020) proposed a non-convex approach that involves initializing a generalized power method with a power method for solving a regularized ML formulation of the binary symmetric SBM. Their method runs in nearly-linear time and is among the most efficient in the literature that achieves exact recovery at the information-theoretic limit. There are still many other interesting methods for the considered problem, such as the mean field method in Zhang & Zhou (2020), a variant of Lloyd's algorithm in Lu & Zhou (2016), and the modularity-based method in Cohen-Addad et al. (2020). Due to the limitation of space, we shall not discuss further here.

1.2. Our Contribution

In this work, we propose a simple and scalable method that can achieve the optimal exact recovery threshold in the symmetric SBM. Our strategy is simply to apply the projected power method to tackle Problem (MLE) directly. Specifically, it starts with an initial point that satisfies a certain partial recovery condition and then applies projected power iterations to refine the iterates successively. In the logarithmic degree regime of the symmetric SBM, we prove that the proposed method achieves exact recovery at the information-theoretic limit. Moreover, we show that it takes $\mathcal{O}(\log n/\log\log n)$ projected power iterations to obtain the underlying communities. Besides, we demonstrate that each projected power iteration is equivalent to a minimum-cost assignment problem (MCAP), which can be solved in $\mathcal{O}(n\log n)$ time. These yield that the proposed method runs in $\mathcal{O}\left(n\log^2 n/\log\log n\right)$ time with a qualified initialization. This is competitive with the most efficient algorithms in the literature for the considered problem. It is worth noting that despite the simplicity of the proposed method, it only requires a partial recovery condition for the initial point, which is generally milder than almost exact recovery conditions that are needed for most existing two-stage algorithms; see, e.g., Gao et al. (2017, Algorithm 2), Yun & Proutiere (2014, Algorithm 2), and Abbe & Sandon (2015, Sphere-comparison algorithm).

Our work also contributes to the emerging area of provable non-convex methods. In particular, our result indicates that the ML formulation of the symmetric SBM, albeit non-convex and discrete, can be solved via a carefully designed, yet simple, iterative procedure. Prior to our work, such discrete optimization problem is usually handled either by SDP relaxation (see, e.g., Amini et al. (2018); Fei & Chen (2018)) or by non-convex but continuous relaxation (see, e.g., Bandeira et al. (2016)). We believe that the proposed non-convex approach can be extended to other structured discrete optimization problems; cf. Liu et al. (2017b).

The rest of this paper is organized as follows. In Section 2, we introduce the proposed method for exact community recovery and present the main results of this paper. In Sections 3, we prove the main results. We then report some numerical results in Section 4 and conclude in Section 5.

Notation. Let $\mathbb{R}^n$ be the n-dimensional Euclidean space and $\|\cdot\|$ be the Euclidean norm. We write matrices in capital bold letters like $\boldsymbol{A}$, vectors in bold lower case like $\boldsymbol{a}$, and scalars as plain letters. Given a matrix $\boldsymbol{A}$, we use $\|\boldsymbol{A}\|$ to denote its spectral norm, $\|\boldsymbol{A}\|_F$ its Frobenius norm, and a_{ij} its (i,j)-th element. 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. We use $\mathbf{1}_n$ and $\boldsymbol{E}_n$ to denote the n-dimensional all-one vector and $n\times n$ all-one matrix, respectively. We use Π_K to denote the collections of all $K\times K$ permutation matrices. We use $\mathbf{Bern}(p)$ to denote the Bernoulli random variable with mean p.

2. Preliminaries and Main Results

In this section, we formally set up the considered problem in the SBM, present the proposed algorithm, and give a summary of our main results. To proceed, we introduce clustering matrices for representing community structures and the symmetric stochastic block model (SBM) for generating observed graphs.

Definition 1. *We say that $\boldsymbol{H}\in\mathbb{R}^{n\times K}$ is a clustering matrix if it takes the form of*

$$h_{ik}=\begin{cases}1, & \text{if } i\in\mathcal{I}_k,\\ 0, & \text{otherwise}\end{cases} \tag{2}$$

for some $\mathcal{I}_1,\ldots,\mathcal{I}_K$ such that $\cup_{k=1}^K\mathcal{I}_k=[n]$ and $\mathcal{I}_k\cap\mathcal{I}_\ell=\emptyset$ for all $1\leq k\neq\ell\leq K$. Moreover, we say that $\boldsymbol{H}\in\mathbb{R}^{n\times K}$ is a balanced clustering matrix if it satisfies the above requirement with $|\mathcal{I}_k|=m$ for all $k\in[K]$. For simplicity, we use $\mathbb{M}_{n,K},\mathbb{H}_{n,K}$ to denote the collections

of all such clustering and balanced clustering matrices, respectively.

Intuitively, a family of sets $\mathcal{I}_1, \ldots, \mathcal{I}_K$ represents a partition of n nodes into K communities such that $h_{ik} = 1$ if node i belongs to the community encoded by $\mathcal{I}_k$ and $h_{ik} = 0$ otherwise. Given a fixed $\boldsymbol{H} \in \mathbb{M}_{n,K}$, $\boldsymbol{HQ}$ for any $\boldsymbol{Q} \in \Pi_K$ represents the same community structure as $\boldsymbol{H}$ up to a permutation of the labels.

Definition 2 (Symmetric SBM)**.** *Let $n \geq 2$ be the number of vertices, $K \geq 2$ be the number of communities, and $p, q \in [0, 1]$ be parameters of the connectivity probabilities. Furthermore, let $\boldsymbol{H}^* \in \mathbb{H}_{n,K}$ represent a unknown partition of n vertices into K equal-sized communities. We say that a random graph G is generated according to the symmetric SBM with parameters (n, K, p, q) and $\boldsymbol{H}^*$ if G has a vertex set $V = [n]$ and the elements $\{a_{ij}\}_{1 \leq i \leq j \leq n}$ of its adjacency matrix $\boldsymbol{A}$ are generated independently by*

$$a_{ij} \sim \begin{cases} \mathbf{Bern}(p), & \text{if } \boldsymbol{h}_i^{*T} \boldsymbol{h}_j^* = 1, \\ \mathbf{Bern}(q), & \text{if } \boldsymbol{h}_i^{*T} \boldsymbol{h}_j^* = 0, \end{cases} \tag{3}$$

where $\boldsymbol{h}_i^{*T}$ is the i-th row of $\boldsymbol{H}^*$.

Intuitively, this model states that given a true partition of n vertices into K unknown communities of equal size, a random graph G is generated by independently connecting each pair of vertices with probability p if they belong to the same community and with probability q otherwise.

Given one observation of such G, our goal is to develop a simple and scalable algorithm that outputs the true partition, i.e., $\boldsymbol{H}^*\boldsymbol{Q}$ for some $\boldsymbol{Q} \in \Pi_K$, with high probability. Since exact recovery requires the node degree to be at least logarithmic (see, e.g., Abbe (2017, Section 2.5)), we focus on the logarithmic sparsity regime of the symmetric SBM in this work, i.e.,

$$p = \alpha \frac{\log n}{n} \quad \text{and} \quad q = \beta \frac{\log n}{n}, \tag{4}$$

where α, β are positive constants.

The main ingredient in our approach is to apply the projected power method for solving Problem (MLE). Specifically, the projected power step takes the form of

$$\boldsymbol{H}^{k+1} \in \mathcal{T}(\boldsymbol{AH}^k), \text{ for all } k \geq 1, \tag{5}$$

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

$$\mathcal{T}(\boldsymbol{C}) = \arg\min \left\{ \|\boldsymbol{H} - \boldsymbol{C}\|_F : \ \boldsymbol{H} \in \mathcal{H} \right\}. \tag{6}$$

Note that Problem (MLE) can be interpreted as a principal component analysis (PCA) problem with some structural constraints. This motivates us to propose a variant of the power iteration as in (5) for solving it. Actually, many algorithms of similar flavor for solving PCA problems with other structural constraints have appeared in the literature; see, e.g., Boumal (2016); Chen & Candès (2018); Deshpande et al. (2014); Journée et al. (2010).

One important step towards guaranteeing rapid convergence of the projected power method for solving Problem (MLE) is to identify a proper initial point $\boldsymbol{H}^0$, which constitutes another ingredient in our approach. Specifically, the initial point $\boldsymbol{H}^0$ is required to satisfy the following condition:

$$\boldsymbol{H}^0 \in \mathbb{M}_{n,K} \quad \text{s.t.} \ \min_{\boldsymbol{Q} \in \Pi_K} \|\boldsymbol{H}^0 - \boldsymbol{H}^*\boldsymbol{Q}\|_F \leq \theta\sqrt{n}, \tag{7}$$

where θ is a constant that will be specified later. We remark that the condition (7) is equivalent to that $\boldsymbol{H}^0$ satisfies a partial recovery condition; see, e.g., Abbe (2017, Definition 4).

We now summarize the proposed method for solving Problem (MLE) in Algorithm 1. It starts with an initial point $\boldsymbol{H}^0$ satisfying (7) and projects $\boldsymbol{H}^0$ onto $\mathcal{H}$ to make the partition balanced. Then, it refines the iterates via projected power iterations N times, where N is an input parameter of the algorithm, and outputs $\boldsymbol{H}^{N+1}$.

Algorithm 1 Projected Power Method for Solving Problem (MLE)

1: **Input:** adjacency matrix $\boldsymbol{A}$, positive integer N
2: **Initialize** an $\boldsymbol{H}^0$ satisfying (7)
3: set $\boldsymbol{H}^1 \leftarrow \mathcal{T}(\boldsymbol{H}^0)$
4: **for** $k = 1, 2, \ldots, N$ **do**
5: set $\boldsymbol{H}^{k+1} \leftarrow \mathcal{T}(\boldsymbol{AH}^k)$
6: **end for**
7: **Output** $\boldsymbol{H}^{N+1}$

We next present the main theorem of this paper, which shows that Algorithm 1 achieves exact recovery down to the information-theoretic threshold and also provides its explicit iteration complexity bound.

Theorem 1. *Let $\boldsymbol{A}$ be the adjacency matrix of a realization of the random graph generated according to the symmetric SBM with parameters (n, K, p, q) and a planted partition $\boldsymbol{H}^* \in \mathbb{H}_{n,K}$. Suppose that p, q satisfy (4) with $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{K}$ and n is sufficiently large. Then, there exists a constant $\gamma > 0$, whose value depends only on α, β, and K, such that the following statement holds with probability at least $1 - n^{-\Omega(1)}$: If the initial point satisfies the partial recovery condition in (7) such that*

$$\theta = \frac{1}{4} \min \left\{ \frac{1}{\sqrt{K}}, \frac{\gamma\sqrt{K}}{16(\alpha - \beta)} \right\}, \tag{8}$$

Algorithm 1 outputs a true partition in $\lceil 2 \log\log n \rceil + \left\lceil \frac{2 \log n}{\log\log n} \right\rceil + 2$ projected power iterations.

Before we proceed, some remarks are in order. First, an $\boldsymbol{H}^0$ satisfying (7) can be found by a host of initialization procedures in existing methods. For example, Gao et al. (2017, Algorithm 2) and Yun & Proutiere (2014, Algorithm 2) respectively proposed spectral clustering based initialization procedures that can obtain an $\boldsymbol{H}^0$ satisfying

$$\boldsymbol{H}^0 \in \mathbb{M}_{n,K} \text{ s.t. } \min_{\boldsymbol{Q}\in\Pi_K} \|\boldsymbol{H}^0 - \boldsymbol{H}^*\boldsymbol{Q}\|_F \lesssim \sqrt{\frac{n}{\log n}}. \quad (9)$$

These initializations are cheap to compute and automatically fulfill the partial recovery requirement in (7) when n is sufficiently large. Note that compared to our projected power method, the refinement procedures in Gao et al. (2017, Algorithm 1) and Yun & Proutiere (2014, Algorithm 1) are rather complicated. Besides, we remark that (9) is a condition of almost exact recovery (see, e.g., Abbe (2017, Definition 4)). It is much more stringent than (7), which is merely a condition of partial recovery.

Second, as we show in Proposition 1, the projection in (6) is equivalent to a minimum-cost assignment problem (MCAP), which is a special linear programming (LP) problem and can be solved very efficiently; see Tokuyama & Nakano (1995). We refer the reader to Section A.1 of the appendix for the formal definition of the MCAP.

Proposition 1. *Problem (6) is equivalent to a minimum-cost assignment problem, which can be solved in $\mathcal{O}(K^2 n \log n)$ time.*

This, together with the time complexity of computing the matrix product $\boldsymbol{A}\boldsymbol{H}$ for some $\boldsymbol{H} \in \mathbb{R}^{n\times K}$ and Theorem 1, immediately implies the time complexity of Algorithm 1 with a qualified initialization.

Corollary 1. *Consider the setting of Theorem 1. If Algorithm 1 uses an initial point that satisfies the partial recovery condition in (7) with θ in (8), then it outputs a true partition in*

$$\mathcal{O}\left(\left(K^2 + 3\alpha + 3(K-1)\beta\right)\frac{n\log^2 n}{\log\log n}\right)$$

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

Finally, it is worth noting that the proposed method in Algorithm 1 can be viewed as an extension of that in Wang et al. (2020), both of which are essentially the projected gradient method applied to the corresponding ML formulation. In particular, when $K = 2$, the projection operators in these two works both admit a closed-form solution, which can be done via partial sorting. Moreover, our method can be applied to do community detection in the setting of multiple communities, i.e., $K \geq 2$, while that in Wang et al. (2020) only works when $K = 2$. Besides, the method in Wang et al. (2020) requires a spectral initialization to satisfy a condition of almost exact recovery. By contrast, any point satisfying the partial recovery condition in (7), including some spectral initializations, is a qualified initialization for Algorithm 1.

3. Proofs of Main Results

In this section, we provide the proofs of our main results in Section 2. The complete proofs of the theorem, propositions, and lemmas can be found in Sections B, C of the appendix.

3.1. Analysis of the Projected Power Iteration

In this subsection, we study the convergence behavior of the projected power iterations in Algorithm 1. Our main idea is to show the contraction property of the projection operator $\mathcal{T}$ in the symmetric SBM. Let

$$\mathcal{P} = \{\boldsymbol{H} \in \mathbb{R}^{n\times K} : \boldsymbol{H}\mathbf{1}_K = \mathbf{1}_n, \boldsymbol{H}^T\mathbf{1}_n = m\mathbf{1}_K, \boldsymbol{H} \geq 0\}.$$

To begin, we present a lemma that establishes an equivalence among the set of extreme points of this polytope, the discrete set $\mathcal{H}$, and the collection of all balanced clustering matrices $\mathbb{H}_{n,K}$.

Lemma 1. *The following statements are equivalent:*
(i) $\boldsymbol{H} \in \mathcal{H}$. *(ii) $\boldsymbol{H}$ is an extreme point of $\mathcal{P}$.*
(iii) $\boldsymbol{H} \in \mathbb{H}_{n,K}$.

It is worth noting that the proof this lemma builds on the total unimodularity (see, e.g., Heller & Tompkins (1956); Hoffman & Kruskal (2010)) of the equality constraint matrix of the polytope $\mathcal{P}$. Equipped with this lemma, we can show that Problem (6) is equivalent to an LP.

Proposition 2. *For any $\boldsymbol{C} \in \mathbb{R}^{n\times K}$, Problem (6) is equivalent to the following LP:*

$$\mathcal{T}(\boldsymbol{C}) = \arg\max\{\langle \boldsymbol{C}, \boldsymbol{H}\rangle : \boldsymbol{H} \in \mathcal{P}\}. \quad (10)$$

Next, we characterize the optimal solutions of the LP in (10) explicitly by exploiting the structure of the polytope $\mathcal{P}$.

Lemma 2. *For a matrix $\boldsymbol{C} \in \mathbb{R}^{n\times K}$, it holds that $\boldsymbol{H} \in \mathcal{T}(\boldsymbol{C})$ if and only if*

$$h_{ik} = \begin{cases} 1, & \text{if } i \in \mathcal{I}_k, \\ 0, & \text{otherwise}, \end{cases}$$

where $\mathcal{I}_1, \ldots, \mathcal{I}_K$ satisfies (i) $\cup_{k=1}^K \mathcal{I}_k = [n]$, $\mathcal{I}_k \cap \mathcal{I}_\ell = \emptyset$, and $|\mathcal{I}_k| = m$ for all $1 \leq k \neq \ell \leq K$, and (ii) there exists $\boldsymbol{w} \in \mathbb{R}^K$ such that

$$c_{ik} - c_{i\ell} \geq w_k - w_\ell \geq c_{jk} - c_{j\ell} \quad (11)$$

for all $i \in \mathcal{I}_k$, $j \in \mathcal{I}_\ell$, and $1 \leq k \neq \ell \leq K$.

When $K = 2$, let $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ denote the first and second columns of $\boldsymbol{C} \in \mathbb{R}^{n\times 2}$, respectively. In this scenario, Lemma 2 implies that solving the LP in (10) boils down to

finding the indices that correspond to the $n/2$ largest entries of the vector $\boldsymbol{c}_1 - \boldsymbol{c}_2$, which can be done via median finding efficiently.

Based on the above lemma, we can show that the projection operator $\mathcal{T}$ in (6) possesses a Lipschitz-like property in spite of the fact that $\mathcal{H}$ is a discrete set.

Lemma 3. *Let $\delta > 0$, $\boldsymbol{C} \in \mathbb{R}^{n\times K}$ be arbitrary and $m = n/K$. Suppose that there exists a family of index sets $\mathcal{I}_1, \ldots, \mathcal{I}_K$ 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*

$$c_{ik} - c_{i\ell} \geq \delta \tag{12}$$

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 $\boldsymbol{V}' \in \mathcal{T}(\boldsymbol{C}')$, it holds that

$$\|\boldsymbol{V} - \boldsymbol{V}'\|_F \leq \frac{2\|\boldsymbol{C} - \boldsymbol{C}'\|_F}{\delta}. \tag{13}$$

Next, we show an inequality that is useful in establishing the contraction property of the projected power iterations.

Lemma 4. *Let $\Delta = \boldsymbol{A} - \mathbb{E}[\boldsymbol{A}]$. Suppose that $\varepsilon \in (0, 1/\sqrt{K})$ and $\boldsymbol{H} \in \mathcal{H}$ such that $\|\boldsymbol{H} - \boldsymbol{H}^*\boldsymbol{Q}\|_F \leq \varepsilon\sqrt{n}$ for some $\boldsymbol{Q} \in \Pi_K$. Then, it holds that*

$$\|\boldsymbol{A}(\boldsymbol{H} - \boldsymbol{H}^*\boldsymbol{Q})\|_F \leq \left(\frac{4\varepsilon n}{\sqrt{K}}(p-q) + \|\Delta\|\right)\|\boldsymbol{H} - \boldsymbol{H}^*\boldsymbol{Q}\|_F.$$

Then, we present some probabilistic results that will be used for establishing the contraction property of the projected power iterations.

Lemma 5. *Let $\Delta = \boldsymbol{A} - \mathbb{E}[\boldsymbol{A}]$. There exists a constant $c_1 > 0$, whose value only depends on α and β, such that*

$$\|\Delta\| \leq c_1\sqrt{\log n} \tag{14}$$

holds with probability at least $1 - n^{-3}$.

This lemma provides a spectral bound on the deviation of $\boldsymbol{A}$ from its mean. It is a direct consequence of Lei & Rinaldo (2015, Theorem 5.2) and thus we omit its proof.

Lemma 6. *Let $m = n/K$ and $\alpha > \beta > 0$ be constants. Suppose that $\{W_i\}_{i=1}^m$ are i.i.d. $\mathbf{Bern}(\alpha \log n/n)$ and $\{Z_i\}_{i=1}^m$ are i.i.d. $\mathbf{Bern}(\beta \log n/n)$ that is independent of $\{W_i\}_{i=1}^m$. Then, for any $\gamma \in \mathbb{R}$, it holds that*

$$\mathbb{P}\left(\sum_{i=1}^m W_i - \sum_{i=1}^m Z_i \leq \gamma \log n\right) \leq n^{-\frac{(\sqrt{\alpha}-\sqrt{\beta})^2}{K} + \frac{\gamma\log(\alpha/\beta)}{2}}.$$

This lemma is proved in Abbe et al. (2020, Lemma 8). Based on the this lemma, we can show that the entries of $\boldsymbol{A}\boldsymbol{H}^*$ satisfy the requirement of (12) in Lemma 3 with high probability.

Lemma 7. *Suppose that $\alpha > \beta > 0$ and $\boldsymbol{C} = \boldsymbol{A}\boldsymbol{H}^*$. Let $\mathcal{I}_k = \{i \in [n] : h_{ik}^* = 1\}$ for all $k \in [K]$. If $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{K}$, there exists a constant $\gamma > 0$, whose value depends only on α, β, and K, such that for all $i \in \mathcal{I}_k$ and $1 \leq k \neq \ell \leq K$,*

$$c_{ik} - c_{i\ell} \geq \gamma \log n \tag{15}$$

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

Armed with the above results, we are now ready to show that the projected power iteration possesses a contraction property in a certain neighborhood of $\boldsymbol{H}^*\boldsymbol{Q}$ for some $\boldsymbol{Q} \in \Pi_K$.

Proposition 3. *Suppose that the constants $\alpha, \beta > 0$ satisfy $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{K}$ and $n > \exp(16c_1^2/\gamma^2)$. Then, the following event happens with probability at least $1 - n^{-\Omega(1)}$: For all $\boldsymbol{H} \in \mathcal{H}$ and $\varepsilon \in \left(0, \min\left\{\frac{1}{\sqrt{K}}, \frac{\gamma\sqrt{K}}{16(\alpha-\beta)}\right\}\right)$ such that $\|\boldsymbol{H} - \boldsymbol{H}^*\boldsymbol{Q}\|_F \leq \varepsilon\sqrt{n}$ for some $\boldsymbol{Q} \in \Pi_K$, it holds that*

$$\|\boldsymbol{V} - \boldsymbol{H}^*\boldsymbol{Q}\|_F \leq \kappa\|\boldsymbol{H} - \boldsymbol{H}^*\boldsymbol{Q}\|_F \tag{16}$$

for any $\boldsymbol{V} \in \mathcal{T}(\boldsymbol{A}\boldsymbol{H})$, where

$$\kappa = 4\max\left\{\frac{4\varepsilon(\alpha-\beta)}{\gamma\sqrt{K}}, \frac{c_1}{\gamma\sqrt{\log n}}\right\} \in (0,1) \tag{17}$$

and c_1, γ are the constants in Lemmas 5 and 7, respectively.

Observe that the contraction rate κ is decreasing to a quantity on the order of $1/\sqrt{\log n}$ as the iterates approach a ground truth. This implies that the better the initialization, the less iterations the proposed method requires to find a ground truth.

The following lemma indicates that the projected power iterations exhibit one-step convergence to a ground truth. This would imply the finite termination of the proposed algorithm.

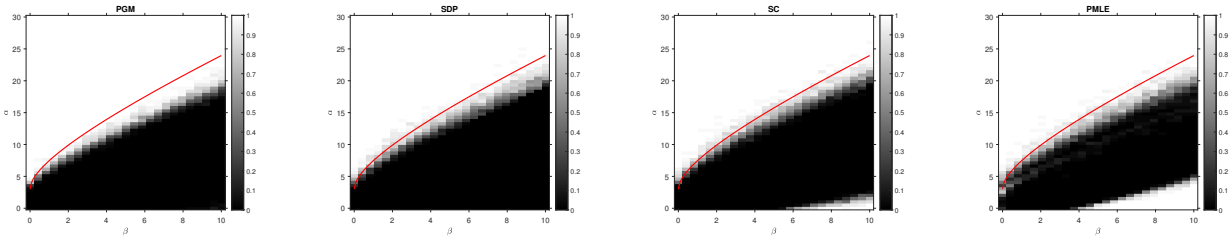
Lemma 8. *Suppose that the constants $\alpha > \beta > 0$ satisfy $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{K}$. Then, the following statement holds with probability at least $1 - n^{-\Omega(1)}$: For all $\boldsymbol{H} \in \mathcal{H}$ such that $\|\boldsymbol{H} - \boldsymbol{H}^*\boldsymbol{Q}\|_F < \sqrt{\gamma \log n}$ for some $\boldsymbol{Q} \in \Pi_K$, it holds that*

$$\mathcal{T}(\boldsymbol{A}\boldsymbol{H}) = \{\boldsymbol{H}^*\boldsymbol{Q}\}, \tag{18}$$

where $\gamma > 0$ is the constant in Lemma 7.

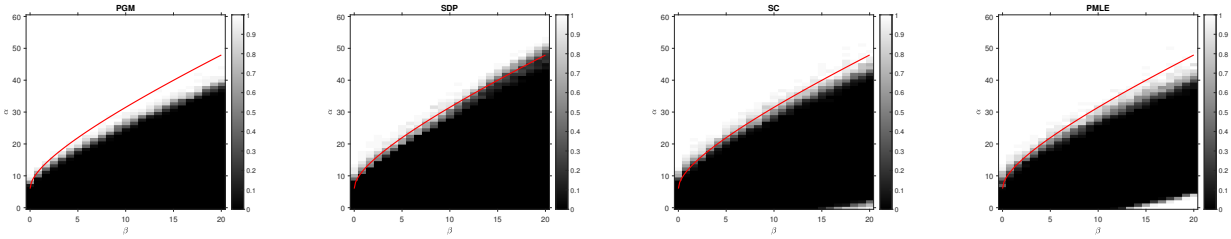
3.2. Proof of Theorem 1

Now, we are ready to derive the iteration complexity bound of Algorithm 1 equipped with the results in Section 3.1. We first provide a formal version of Theorem 1 and then



(a) PPM (b) SDP (c) SC (d) PMLE

Figure 1. Phase transition in the setting of $n = 300$, $K = 3$: 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 $\sqrt{\alpha} - \sqrt{\beta} = \sqrt{3}$.



(a) PPM (b) SDP (c) SC (d) PMLE

Figure 2. Phase transition in the setting of $n = 600$, $K = 6$: The red curve is the information-theoretic threshold $\sqrt{\alpha} - \sqrt{\beta} = \sqrt{6}$.

sketch its proof. The full proof can be found in Section C of the appendix. Recall that θ, c_1, and γ are the constants in Theorem 1, Lemma 5, and Lemma 7, respectively. To simplify the notations in the sequel, let

$$\phi = \frac{c_1\sqrt{K}}{16(\alpha - \beta)}. \tag{19}$$

Theorem 2. *Consider the setting of Theorem 1. Suppose that*

$$n > \exp\left(\max\left\{\frac{64c_1^2}{\gamma^2}, \frac{\gamma^2}{c_1^2}, \frac{4\sqrt{2}\phi}{\sqrt{\gamma}}, \frac{256c_1^4}{\gamma^4}\right\}\right). \tag{20}$$

Then, the following statement holds with probability at least $1 - n^{-\Omega(1)}$: If the initial point $\boldsymbol{H}^0 \in \mathbb{M}_{n,K}$ satisfies

$$\|\boldsymbol{H}^0 - \boldsymbol{H}^*\boldsymbol{Q}\|_F \le \theta\sqrt{n} \tag{21}$$

for some $\boldsymbol{Q} \in \Pi_K$ and θ is defined in (8), Algorithm 1 outputs $\boldsymbol{H}^*\boldsymbol{Q}$ within $\lceil 2\log\log n\rceil + \left\lceil \frac{2\log n}{\log\log n}\right\rceil + 2$ projected power iterations.

Proof. Suppose that the statements in Proposition 3 and Lemma 8 hold, which happens with probability at least $1 - n^{-\Omega(1)}$ by the union bound. We first show that for all $k \ge 2$, $\boldsymbol{H}^k \in \mathbb{H}_{n,K}$ satisfies $\|\boldsymbol{H}^k - \boldsymbol{H}^*\boldsymbol{Q}\|_F \le 2\theta\sqrt{n}$ and

$$\|\boldsymbol{H}^k - \boldsymbol{H}^*\boldsymbol{Q}\|_F \le \frac{1}{2}\|\boldsymbol{H}^{k-1} - \boldsymbol{H}^*\boldsymbol{Q}\|_F,$$

and it holds for $N_1 = \lceil 2\log\log n\rceil + 1$ that

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

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

$$\|\boldsymbol{H}^{N_1+k} - \boldsymbol{H}^*\boldsymbol{Q}\|_F \le \frac{4c_1}{\gamma\sqrt{\log n}}\|\boldsymbol{H}^{N_1+k-1} - \boldsymbol{H}^*\boldsymbol{Q}\|_F,$$

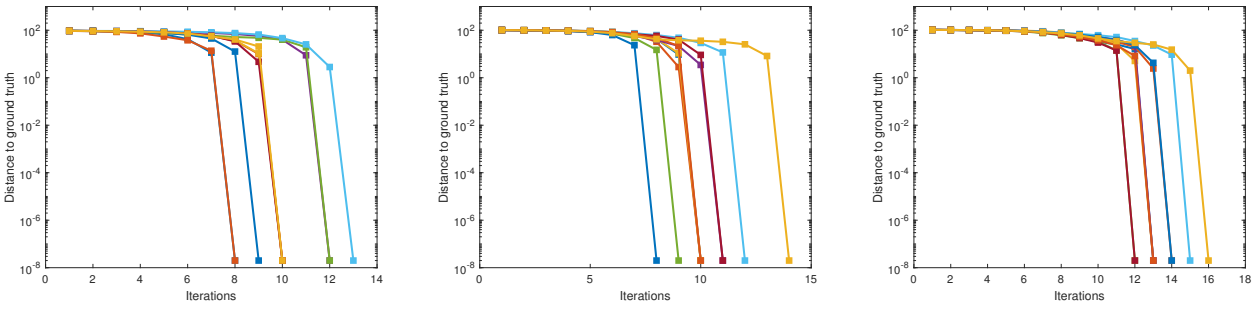
and it holds for $N_2 = \left\lceil \frac{2\log n}{\log\log n}\right\rceil$ that

$$\|\boldsymbol{H}^{N_2+N_1} - \boldsymbol{H}^*\|_F < \sqrt{\gamma\log n}.$$

Once this holds, we have $\boldsymbol{H}^{N_1+N_2+1} = \boldsymbol{H}^*$ by Lemma 8. Then, the desired result is established. □

4. Experimental Results

In this section, we report the recovery performance and numerical efficiency of our proposed method for recovering communities on both synthetic and real data sets. We also compare our method with three existing methods, which are the SDP-based method in Amini et al. (2018), the spectral clustering (SC) method in Su et al. (2019), and the local penalized ML estimation (PMLE) method in Gao et al. (2017). In the implementation, we employ Gao et al. (2017, Algorithm 2) for computing the initial point $\boldsymbol{H}^0$ in Algorithm 1 if we do not specify the initialization method. Moreover, we use alternating direction method of multipliers (ADMM) for



(a) $(\alpha, \beta, K) = (18, 4, 4)$ (b) $(\alpha, \beta, K) = (36, 8, 8)$ (c) $(\alpha, \beta, K) = (54, 12, 12)$

Figure 3. Convergence performance of PPM: The x-axis is number of iterations and the y-axis is the distance from an iterate to a ground truth, i.e., $\min_{\boldsymbol{Q}\in\Pi_K} \|\boldsymbol{H}^k - \boldsymbol{H}^*\boldsymbol{Q}\|_F$, where $\boldsymbol{H}^k$ is the k-th iterate generated by PPM.

solving the SDP as suggested in Amini et al. (2018),[1] the MATLAB function eigs for computing the eigenvectors that are needed in the SC method and the first stage of the PMLE method, and the MATLAB function kmeans for computing the partition in the SC method. For ease of reference, we denote our method simply by PPM. All of our simulations are implemented in MATLAB R2020a on a PC running Windows 10 with 16GB memory and Intel(R) Core(TM) i5-8600 3.10GHz CPU. Our code is available at https://github.com/peng8wang/ICML2021-PPM-SBM.

4.1. Phase Transition and Computational Time

We first conduct the experiments to examine the phase transition property and running time of the aforementioned methods for recovering communities in graphs that are generated by the symmetric SBM in Definition 2. We have two sets of simulations. We choose $n = 300, K = 3$ (resp. $n = 600, K = 6$), and let the parameter α in (4) vary from 0 to 30 (resp. 60) with increments of 0.5 (resp. 1) and the parameter β in (4) vary from 0 to 10 (resp. 20) with increments of 0.4 (resp. 0.8). For every pair of α and β, we generate 40 instances and calculate the ratio of exactly recovering the communities for all the tested methods. The phase transition results are reported in Figures 1 and 2. According to these figures, we can observe that all the methods exhibit a phase transition phenomenon and the recovery performance of PPM is slightly better than the other three methods. Moreover, Figures 1(a) and 2(a) indicate that PPM achieves the optimal recovery threshold, which supports the result in Theorem 1. Besides, we record the total CPU time consumed by each method for completing the phase transition experiments in Table 2. It can be observed that PPM is slightly better than PMLE and substantially faster than SC and SDP.

[1]The code is available at https://github.com/aaamini/SBM-SDP.

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

Time (s)	PPM	SDP	SC	PMLE
$n = 300,\ K = 3$	**401**	25887	1438	572
$n = 600,\ K = 6$	**1824**	82426	3669	2661

4.2. Convergence Performance

We next conduct the experiments to study the convergence performance of PPM for recovering the communities in graphs generated by the symmetric SBM in Definition 2. In the simulations, we choose three different sets of (α, β, K) such that $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{K}$ and generate graphs of dimension $n = 6000$. Moreover, we generate the initial point $\boldsymbol{H}^0$ in Algorithm 1 via $\boldsymbol{H}^0 \in \mathcal{T}(\boldsymbol{G})$, where each entry of $\boldsymbol{G} \in \mathbb{R}^{n\times K}$ is randomly generated by the standard normal distribution. Let $\boldsymbol{H}^k$ denote the k-th iterate of the PPM. In each graph, we run PPM 10 times from different initial points and then plot the distances of the iterates to the ground truth, i.e., $\min_{\boldsymbol{Q}\in\Pi_K} \|\boldsymbol{H}^k - \boldsymbol{H}^*\boldsymbol{Q}\|_F$, against the iteration number in Figure 3. It can be observed that PPM exhibits a finite termination phenomenon and converges to the ground truth within 20 iterations even if it starts from a randomly generated initial point. This also corroborates the one-step convergence result in Lemma 8 and the iteration complexity in Theorem 1.

4.3. Recovery Efficiency and Accuracy

Finally, we conduct the experiments to compare the recovery efficiency and accuracy of our method with SDP, SC, and PMLE on real data sets. We use the data sets *polbooks*, *polblogs*, and *football* downloaded from the SuiteSparse Matrix Collection (Davis & Hu, 2011).[2] For the set *football*, we remove the communities whose sizes are less than 10. To tackle the difficulty that these real networks have unbalanced

[2]https://sparse.tamu.edu/

communities, we modify the second constraint in (1) as $\boldsymbol{H}^T\mathbf{1}_K = \boldsymbol{\pi}$, where π_k denotes the k-th community size for all $k \in [K]$, and then apply PPM for solving the resulting formulation as in Algorithm 1. The stopping criteria for the tested methods are set as follows. For PPM, we terminate it when there exists some iterate $k \geq 6$ such that $\|\boldsymbol{H}^k - \boldsymbol{H}^l\|_F \leq 10^{-3}$ for some $k-5 \leq l \leq k-1$; for ADMM, we terminate it when the norm of difference of two consecutive iterates is less than 10^{-3}. No stopping criterion is needed for SC and PMLE since SC employs the MATLAB function kmeans to do the clustering and PMLE directly assigns each vertex to the corresponding community based on the initialization partition. Besides, we generate an initial point for PPM as in Section 4.2. Then, we run each algorithm 10 times and select the best solution (in terms of function value) as its recovery solution. Moreover, we set the maximum iteration number for PPM and ADMM as 1000. To compare the recovery efficiency and accuracy of the tested methods, we report the total CPU time for all runs and the number of misclassified vertices (MVs) of each method in Table 3. These results, together with those in Table 2, demonstrate that our proposed method is comparable to these state-of-the-art methods in terms of recovery efficiency and accuracy on both synthetic and real data sets.

Table 3. Total CPU times (in seconds) and the number of misclassified vertices (MVs) of the methods on real data sets.

Time (s)	PPM	SDP	SC	PMLE
polbooks	**0.28**	10.26	0.30	19.67
polblogs	**0.02**	2348	0.41	1.39
football	**0.21**	0.83	0.42	0.40
num. of MVs	PPM	SDP	SC	PMLE
polbooks	**18**	24	**18**	19
polblogs	**52**	238	215	279
football	4	**2**	**2**	13

5. Concluding Remarks

In this work, we proposed a projected power method for solving the ML formulation of the symmetric SBM. We showed that provided an initial point satisfying a mild partial recovery condition, this method achieves exact recovery down to the information-theoretic threshold and runs in $\mathcal{O}(n \log n / \log\log n)$ time in the logarithmic degree regime. This is also demonstrated by our numerical results. Moreover, it is observed in the numerical results that the proposed method still works effectively even with a random initialization. Then, one natural future direction is to study the convergence behavior of the proposed method with a random initialization. Another direction is to extend our proposed method to other variants of the basic SBM, such as degree-corrected block models (see, e.g., Gao et al. (2018); Karrer & Newman (2011)), labelled SBMs (see, e.g., Heimlicher et al. (2012); Yun & Proutiere (2016)), and overlapping SBMs (see, e.g., Airoldi et al. (2008); Gopalan & Blei (2013)).

Acknowledgements

This work is supported in part by CUHK Research Sustainability of Major RGC Funding Schemes project 3133236.

References

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

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

Abbe, E., Bandeira, A. S., and Hall, G. Exact recovery in the stochastic block model. *IEEE Transactions on Information Theory*, 62(1):471–487, 2016.

Abbe, E., Fan, J., Wang, K., and Zhong, Y. Entrywise eigenvector analysis of random matrices with low expected rank. *Annals of Statistics*, 48(3):1452–1474, 2020.

Airoldi, E. M., Blei, D. M., Fienberg, S. E., and Xing, E. P. Mixed membership stochastic blockmodels. *Journal of Machine Learning Research*, 2008.

Amini, A. A., Chen, A., Bickel, P. J., and Levina, E. Pseudo-likelihood methods for community detection in large sparse networks. *Annals of Statistics*, 41(4):2097–2122, 2013.

Amini, A. A. and Levina, E. On semidefinite relaxations for the block model. *Annals of Statistics*, 46(1):149–179, 2018.

Bandeira, A. S. Random Laplacian matrices and convex relaxations. *Foundations of Computational Mathematics*, 18(2):345–379, 2018.

Bandeira, A. S., Boumal, N., and Voroninski, V. On the low-rank approach for semidefinite programs arising in synchronization and community detection. In *Conference on Learning Theory*, pp. 361–382, 2016.

Bendory, T., Eldar, Y. C., and Boumal, N. Non-convex phase retrieval from STFT measurements. *IEEE Transactions on Information Theory*, 64(1):467–484, 2017.

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

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(1-2):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.

Cohen-Addad, V., Kosowski, A., Mallmann-Trenn, F., and Saulpic, D. On the power of Louvain in the stochastic block model. *Advances in Neural Information Processing Systems*, 33, 2020.

Davis, T. A. and Hu, Y. The University of Florida sparse matrix collection. *ACM Transactions on Mathematical Software (TOMS)*, 38(1):1–25, 2011.

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

Fei, Y. and Chen, Y. Exponential error rates of SDP for block models: Beyond Grothendieck's inequality. *IEEE Transactions on Information Theory*, 65(1):551–571, 2018.

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

Gao, C., Ma, Z., Zhang, A. Y., and Zhou, H. H. Achieving optimal misclassification proportion in stochastic block models. *Journal of Machine Learning Research*, 18(1): 1980–2024, 2017.

Gao, C., Ma, Z., Zhang, A. Y., and Zhou, H. H. Community detection in degree-corrected block models. *Annals of Statistics*, 46(5):2153–2185, 2018.

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.

Gopalan, P. K. and Blei, D. M. Efficient discovery of overlapping communities in massive networks. *Proceedings of the National Academy of Sciences*, 110(36):14534–14539, 2013.

Guédon, O. and Vershynin, R. Community detection in sparse networks via Grothendieck's inequality. *Probability Theory and Related Fields*, 165(3-4):1025–1049, 2016.

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

Hajek, B., Wu, Y., and Xu, J. Achieving exact cluster recovery threshold via semidefinite programming: Extensions. *IEEE Transactions on Information Theory*, 62(10):5918–5937, 2016b.

Heimlicher, S., Lelarge, M., and Massoulié, L. Community detection in the labelled stochastic block model. *arXiv preprint arXiv:1209.2910*, 2012.

Heller, I. and Tompkins, C. An extension of a theorem of Dantzig's. *Linear Inequalities and Related Systems*, 38: 247–254, 1956.

Hoffman, A. J. and Kruskal, J. B. Integral boundary points of convex polyhedra. In *50 Years of Integer Programming 1958-2008*, pp. 49–76. Springer, 2010.

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.

Karrer, B. and Newman, M. E. Stochastic blockmodels and community structure in networks. *Physical Review E*, 83 (1):016107, 2011.

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

Li, X., Chen, Y., and Xu, J. Convex relaxation methods for community detection. *arXiv preprint arXiv:1810.00315*, 2018.

Ling, S. Improved performance guarantees for orthogonal group synchronization via generalized power method. *arXiv preprint arXiv:2012.00470*, 2020.

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

Liu, H., Yue, M.-C., So, A. M.-C., and Ma, W.-K. A discrete first-order method for large-scale MIMO detection with provable guarantees. In *2017 IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 1–5. IEEE, 2017b.

Liu, H., Yue, M.-C., and So, A. M.-C. A unified approach to synchronization problems over subgroups of the orthogonal group. *arXiv preprint arXiv:2009.07514*, 2020.

Lu, Y. and Zhou, H. H. Statistical and computational guarantees of Lloyd's algorithm and its variants. *arXiv preprint arXiv:1612.02099*, 2016.

McSherry, F. Spectral partitioning of random graphs. In *Proceedings of the 42nd IEEE Symposium on Foundations of Computer Science*, pp. 529–537. IEEE, 2001.

Mossel, E., Neeman, J., and Sly, A. Consistency thresholds for binary symmetric block models. *arXiv preprint arXiv:1407.1591*, 3(5), 2014.

Newman, M. E. and Girvan, M. Finding and evaluating community structure in networks. *Physical Review E*, 69 (2):026113, 2004.

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

Su, L., Wang, W., and Zhang, Y. Strong consistency of spectral clustering for stochastic block models. *IEEE Transactions on Information Theory*, 66(1):324–338, 2019.

Tokuyama, T. and Nakano, J. Geometric algorithms for the minimum cost assignment problem. *Random Structures & Algorithms*, 6(4):393–406, 1995.

Veinott Jr, A. F. and Dantzig, G. B. Integral extreme points. Technical report, Stanford Univ CA Operations Research House, 1967.

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

Yun, S.-Y. and Proutiere, A. Accurate community detection in the stochastic block model via spectral algorithms. *arXiv preprint arXiv:1412.7335*, 2014.

Yun, S.-Y. and Proutiere, A. Optimal cluster recovery in the labeled stochastic block model. In *Advances in Neural Information Processing Systems*, pp. 965–973, 2016.

Zhang, A. Y. and Zhou, H. H. Minimax rates of community detection in stochastic block models. *Annals of Statistics*, 44(5):2252–2280, 2016.

Zhang, A. Y. and Zhou, H. H. Theoretical and computational guarantees of mean field variational inference for community detection. *Annals of Statistics*, 48(5):2575–2598, 2020.

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

Zhou, Z. and Li, P. Rate optimal chernoff bound and application to community detection in the stochastic block models. *Electronic Journal of Statistics*, 14(1):1302–1347, 2020.