



# Non-convex exact community recovery in stochastic block model

Peng Wang<sup>1</sup> · Zirui Zhou<sup>2</sup> · Anthony Man-Cho So<sup>1</sup>

Received: 13 September 2020 / Accepted: 26 September 2021

© Springer-Verlag GmbH Germany, part of Springer Nature and Mathematical Optimization Society 2021

## Abstract

Community detection in graphs that are generated according to stochastic block models (SBMs) has received much attention lately. In this paper, we focus on the binary symmetric SBM—in which a graph of  $n$  vertices is randomly generated by first partitioning the vertices into two equal-sized communities and then connecting each pair of vertices with probability that depends on their community memberships—and study the associated exact community recovery problem. Although the maximum-likelihood formulation of the problem is non-convex and discrete, we propose to tackle it using a popular iterative method called projected power iterations. To ensure fast convergence of the method, we initialize it using a point that is generated by another iterative method called orthogonal iterations, which is a classic method for computing invariant subspaces of a symmetric matrix. We show that in the logarithmic sparsity regime of the problem, with high probability the proposed two-stage method can exactly recover the two communities down to the information-theoretic limit in  $\mathcal{O}(n \log^2 n / \log \log n)$  time, which is competitive with a host of existing state-of-the-art methods that have the same recovery performance. We also conduct numerical experiments on both synthetic

---

A preliminary version of this work has appeared in the Proceedings of the 37th International Conference on Machine Learning (ICML 2020), 2020 [37]. The first and third authors are supported in part by the Hong Kong Research Grants Council (RGC) General Research Fund (GRF) project CUHK 14208117 and in part by the CUHK Research Sustainability of Major RGC Funding Schemes project 3133236. The second author is supported in part by the National Natural Science Foundation of China (NSFC) project 11901490 and in part by a HKBU Start-up Grant.

---

✉ Zirui Zhou  
zirui.zhou@huawei.com

Peng Wang  
wangpeng@se.cuhk.edu.hk

Anthony Man-Cho So  
manchoseo@se.cuhk.edu.hk

<sup>1</sup> Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Shatin, NT, Hong Kong

<sup>2</sup> Huawei Technologies Canada Co., Ltd., Burnaby, Canada

and real data sets to demonstrate the efficacy of our proposed method and complement our theoretical development.

**Keywords** Community detection · Exact recovery · Orthogonal iteration · Projected power iteration · Finite termination · Nearly-linear time

## 1 Introduction

Community detection is a fundamental task in network analysis and has found many applications in diverse fields such as physics [16,30], biology [13], and social science [19], to name a few. In research on community detection, the stochastic block model (SBM), which provides a way to generate graphs with community structure, is widely used as a platform for validating theoretical ideas and comparing numerical algorithms. In particular, substantial advances have been made in the past decade on understanding the fundamental limits of community detection in graphs that are generated by SBMs, and on developing computationally tractable methods that can meet different recovery requirements up to their corresponding fundamental limits; see, e.g., [1] and the references therein.

One problem that has been extensively studied in the literature is the exact recovery of communities in the binary symmetric SBM (also known as the planted bisection model). Specifically, given an  $n$ -vertex graph with two equal-sized hidden communities, and each pair of vertices in the graph is connected by an edge with probability  $p$  if they both belong to the same community and with probability  $q$  otherwise, the goal is to achieve *exact recovery* (i.e., recover the underlying communities exactly with high probability) using only the adjacency matrix of the graph. It is well known that whether exact recovery is achievable depends on the scalings of  $p$ ,  $q$ , and  $p - q$ . When  $p = a/n$  and  $q = b/n$  for some  $a > b > 0$  (the constant sparsity regime), it is impossible to recover the communities because the graph is disconnected with high probability [15]. On the other hand, when  $p = \alpha \log n/n$  and  $q = \beta \log n/n$  for some  $\alpha > \beta > 0$  (the logarithmic sparsity regime), Abbe et al. [2] and Mossel et al. [29] independently showed that exact recovery is impossible if  $\sqrt{\alpha} - \sqrt{\beta} < \sqrt{2}$  but is possible if  $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{2}$ , thereby establishing a sharp threshold for exact recovery. The proof of Abbe et al. [2] takes an information-theoretic approach and obtains the said threshold by analyzing the following maximum-likelihood (ML) formulation of the problem:

$$\max \left\{ \mathbf{x}^T \mathbf{A} \mathbf{x} : \mathbf{1}_n^T \mathbf{x} = 0, x_i \in \{\pm 1\}, i = 1, \dots, n \right\}. \quad (\text{MLE})$$

Here,  $\mathbf{A}$  is the adjacency matrix of the graph,  $\mathbf{1}_n$  is the all-one vector of dimension  $n$ , and  $x_i \in \{\pm 1\}$  encodes the community membership of vertex  $i$  for  $i = 1, \dots, n$ . It is shown in [2] that when  $\sqrt{\alpha} - \sqrt{\beta} < \sqrt{2}$ , the ML estimator (i.e., an optimal solution of (MLE)) fails to recover the communities with probability bounded away from zero for sufficiently large  $n$ , but when  $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{2}$ , the ML estimator can exactly recover the communities with high probability.

From a computational point of view, solving Problem (MLE) amounts to finding a minimum bisection of a graph, which is NP-hard in the worst case [18]. Over the past few decades, many algorithms have been proposed to tackle the problem of exact community recovery in the binary symmetric SBM; see, e.g., [1] for a summary of some of the earlier works. In view of the information-theoretic limit established in [2, 29], a natural task is to design efficient algorithms that can exactly recover the communities down to the limit. This has also been undertaken in [2, 29]. The former presents a two-stage algorithm that combines the partial recovery algorithm of Massoulié [28] with a local improvement procedure, while the latter gives a three-stage algorithm that uses spectral clustering for initialization and then combines a partial recovery step with a local refinement procedure. The former also presents an algorithm based on a semidefinite relaxation (SDR) of Problem (MLE) and poses the conjecture that the algorithm can exactly recover the communities down to the information-theoretic limit. The conjecture was later resolved in the affirmative independently by Hajek et al. [21] and Bandeira [7].

Subsequent to the above development, a variety of efficient algorithms with the same recovery performance in the binary symmetric SBM have appeared in the literature. For instance, Abbe and Sandon [4] developed a two-stage algorithm that is similar in spirit to the one in [2]. Yun and Proutiere [38] presented a spectral partition algorithm, which proceeds by applying spectral decomposition to a trimmed adjacency matrix, followed by some local improvements. Later, Gao et al. [17] proposed a two-stage algorithm that employs spectral clustering for initialization and penalized local maximum likelihood estimation for local refinement. It is worth noting that the aforementioned algorithms apply not only to the binary symmetric SBM but also to more general SBMs. More recently, Abbe et al. [3] showed that the vanilla spectral method, which computes the eigenvector associated with the second largest eigenvalue of the adjacency matrix and uses the signs of the entries to identify the communities, already has the desired recovery performance.

Among the existing algorithms that can achieve exact recovery down to the information-theoretic limit in the binary symmetric SBM, the best complexity bound is nearly linear. This is attained by, e.g., the three-stage algorithm of Mossel et al. [29] and the spectral partition algorithm of Yun and Proutiere [38], both of which have an  $\mathcal{O}(n \log^2 n)$  runtime, and the two-stage algorithm of Abbe and Sandon [4], which has a runtime of  $o(n^{1+\epsilon})$  for any  $\epsilon > 0$ . It should be pointed out that even though the vanilla spectral method is conceptually much simpler than these algorithms, it needs to perform an eigenvector computation, and standard complexity analyses of the commonly used methods for this purpose (such as orthogonal iteration) only yield a quadratic bound at best (see, e.g., [34, Part V]).

## 1.1 Our contributions

In this work, we propose a two-stage iterative algorithm that aims to achieve exact recovery in the binary symmetric SBM by directly tackling the non-convex ML estimation problem (MLE). In the first stage, the algorithm applies the classic method of orthogonal iteration to compute an approximation of the eigenvector  $\mathbf{u}_2$  associated with

the second largest eigenvalue of  $\mathbf{A}$ . Such an approximation is then used as an initialization in the second stage of the algorithm, which applies the method of projected power iteration (cf. [25,26]) to solve Problem (MLE). The first stage is akin to that of a host of existing algorithms, such as those in [3,29,38]. However, we are able to show that a coarse approximation of  $\mathbf{u}_2$  is sufficient for the second stage of our proposed algorithm to find an optimal solution of Problem (MLE), which is key to the efficiency of our algorithm. Specifically, we show that in the logarithmic sparsity regime of the binary symmetric SBM, our proposed algorithm achieves exact recovery all the way down to the information-theoretic limit within  $\mathcal{O}(\log n / \log \log n)$  orthogonal iterations and  $\mathcal{O}(\log n / \log \log n)$  projected power iterations, where each orthogonal iteration and projected power iteration can be implemented in  $\mathcal{O}(n \log n)$  time. This yields an overall complexity bound of  $\mathcal{O}(n \log^2 n / \log \log n)$  for our algorithm, which is competitive with some of the most efficient algorithms in the literature that have the same recovery performance. By combining our techniques with the results in [3], we can further show that the vanilla spectral method can be implemented in  $\mathcal{O}(n \log^2 n / \log \log n)$  time. To the best of our knowledge, this is currently the best complexity bound for the method in the context of exact community recovery in the binary symmetric SBM. We also conduct numerical experiments on synthetic and real data sets to evaluate the performance of our proposed algorithm. The results demonstrate the efficacy of the algorithm and complement our theoretical development.

In recent years, there has been a growing body of literature exploring the design and analysis of fast methods for tackling non-convex formulations that arise in applications. These include deep neural networks [32,33], low-rank matrix recovery [12,23], phase retrieval [27,35], source localization [24,31], and synchronization [25,39]. As these works show, the non-convex formulations in question often possess structures that can be exploited by simple and scalable methods, thereby allowing optimal solutions of those formulations to be found efficiently. Our work contributes to this emerging area by showing that in the logarithmic sparsity regime of the binary symmetric SBM, the ML estimation problem (MLE), albeit non-convex and discrete, can be solved to optimality via a carefully designed, yet simple, iterative procedure. Prior to our work, Bandeira et al. [8] considered another non-convex formulation of the community recovery problem, which is obtained by applying the Burer-Monteiro decomposition [11] to the semidefinite relaxation of Problem (MLE). They showed that all second-order stationary points of the non-convex formulation, which can be computed efficiently by the Riemannian trust-region method [10], correspond to the underlying communities with high probability as long as  $(p - q) / \sqrt{p + q} \geq cn^{-1/6}$  for some constant  $c > 0$ . Despite its low computational complexity, the approach requires a much stronger condition on  $p$  and  $q$  to ensure exact recovery. In particular, it cannot guarantee exact recovery in the logarithmic sparsity regime of the binary symmetric SBM.

Lastly, let us highlight the improvements made in this paper over its preliminary version [37]. First, the method in [37] is designed for a regularized version of Problem (MLE). When applying the method to real data sets, which in general are not generated by the SBM, it is difficult to tune the regularization parameter. In the current work, we circumvent this difficulty by handling Problem (MLE) directly, which makes our proposed method simpler and more practical. Second, compared to its regularized

version in [37], Problem (MLE) is more challenging as it contains an additional linear constraint. Nevertheless, we show that a suitably initialized projected power method can solve it efficiently and enjoys the same recovery guarantee as that in [37]. Third, although both the method in [37] and the one proposed in this paper have the property that an iterate will converge in one step to an optimal solution of Problem (MLE) once the former is in a suitable neighborhood of the latter, we show in this paper that the size of the neighborhood can be as large as  $\mathcal{O}(\sqrt{\log n})$ , which improves upon the  $\mathcal{O}(1)$  bound established in [37].

## 1.2 Organization

The rest of this paper is organized as follows. In Sect. 2, we introduce the proposed two-stage algorithm for exact community recovery and present the main result of this paper. In Sect. 3, we prove the main result and discuss its consequences. We then report some numerical results in Sect. 4 and conclude in Sect. 5.

*Notation.* Let  $\mathbb{R}^n$  be the  $n$ -dimensional Euclidean space. We write matrices in bold capital letters like  $\mathbf{A}$ , vectors in bold lower-case letters like  $\mathbf{a}$ , and scalars in plain letters. Given a matrix  $\mathbf{A} \in \mathbb{R}^{m \times n}$ , we use  $\sigma_{\max}(\mathbf{A})$  or  $\|\mathbf{A}\|$  to denote its largest singular value (i.e., spectral norm),  $\sigma_{\min}(\mathbf{A})$  its smallest singular value, and  $a_{ij}$  its  $(i, j)$ -th element. If  $\mathbf{A}$  is symmetric, then we use  $\lambda_{\min}(\mathbf{A})$  to denote its smallest eigenvalue. Given a vector  $\mathbf{x} \in \mathbb{R}^n$ , we use  $\|\mathbf{x}\|_2$  to denote its Euclidean norm,  $x_i$  its  $i$ -th element, and  $\text{diag}(\mathbf{x})$  the diagonal matrix with  $\mathbf{x}$  on its diagonal. We use  $\mathbf{1}_n$  and  $\mathbf{E}_n$  to denote the  $n$ -dimensional all-one vector and  $n \times n$  all-one matrix, respectively, and simply write  $\mathbf{1}$  and  $\mathbf{E}$  when their dimensions can be inferred from the context. Given a positive integer  $n$ , we denote by  $[n]$  the set  $\{1, \dots, n\}$ . Given a discrete set  $T$ , we denote by  $|T|$  the number of elements in  $T$ . We use  $\text{sgn}$  to denote the element-wise sign function; i.e., for any  $\mathbf{x} \in \mathbb{R}^n$ ,

$$[\text{sgn}(\mathbf{x})]_i = \begin{cases} 1, & \text{if } x_i > 0, \\ 0, & \text{if } x_i = 0, \\ -1, & \text{if } x_i < 0, \end{cases} \quad i \in [n].$$

We use  $\text{Bern}(p)$  to denote the Bernoulli random variable with mean  $p$ . Given two random variables  $X$  and  $Y$ , we write  $X \stackrel{d}{=} Y$  if  $X$  and  $Y$  are equal in distribution.

## 2 Preliminaries and main results

In this section, we formally state the considered problem, present the proposed algorithm, and give a summary of our main theoretical results.

To begin, let us introduce a central object in our study—the binary symmetric SBM.

**Definition 1** (Binary Symmetric SBM) Let  $n \geq 2$  be an even integer and  $p, q \in [0, 1]$  be parameters with  $p > q$ . Furthermore, let  $\mathbf{x}^* \in \{-1, +1\}^n$  be a label vector representing a partition of  $[n]$  into two equal-sized subsets (in particular,  $\mathbf{1}^T \mathbf{x}^* = 0$ ). We say that a random graph  $G$  is generated according to the binary symmetric SBM

with parameters  $(n, p, q)$  and label  $\mathbf{x}^*$  if  $G$  has vertex set  $V = [n]$  and the elements  $\{a_{ij}\}_{1 \leq i \leq j \leq n}$  of its adjacency matrix  $\mathbf{A}$  are generated independently by

$$a_{ij} \sim \begin{cases} \text{Bern}(p), & \text{if } x_i^* x_j^* = 1, \\ \text{Bern}(q), & \text{if } x_i^* x_j^* = -1. \end{cases} \quad (1)$$

Intuitively, the label vector  $\mathbf{x}^*$  induces two equal-sized communities in the graph  $G$ . Note that we allow self-loops in  $G$ , though our analysis also applies to the case where no self-loop is allowed (i.e.,  $a_{ii} = 0$  for all  $i \in [n]$ ); see Sect. 3.4.1.

Now, given a realization of  $G$  that is generated according to the binary symmetric SBM, the problem of interest is to recover the two communities. Since  $-\mathbf{x}^*$  represents the same community structure as  $\mathbf{x}^*$ , this is equivalent to identifying  $\mathbf{x}^*$  or  $-\mathbf{x}^*$  from the adjacency matrix  $\mathbf{A}$ . As in [2], we say that an estimator achieves *exact recovery*<sup>1</sup> if it yields  $\mathbf{x}^*$  or  $-\mathbf{x}^*$  with probability tending to one as  $n \rightarrow \infty$ , where the probability is taken with respect to the distribution in (1).

In this paper, we focus on the logarithmic sparsity regime of the binary symmetric SBM—i.e.,

$$p = \frac{\alpha \log n}{n} \quad \text{and} \quad q = \frac{\beta \log n}{n} \quad (2)$$

for some constants  $\alpha > \beta > 0$ —and propose to solve the community recovery problem by directly handling the non-convex ML estimation problem (MLE), even though it is NP-hard in the worst case. The first ingredient in our approach is a simple iterative procedure called the *method of projected power iteration*, which is essentially the projected gradient method applied to Problem (MLE). Specifically, let

$$\mathcal{F} := \left\{ \mathbf{x} \in \mathbb{R}^n : \mathbf{1}^T \mathbf{x} = 0, x_i = \pm 1, i \in [n] \right\} \quad (3)$$

denote the feasible set of Problem (MLE). Furthermore, let  $\mathcal{P} : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$  be the projection operator onto  $\mathcal{F}$ ; i.e., for any  $\mathbf{c} \in \mathbb{R}^n$ ,

$$\mathcal{P}(\mathbf{c}) := \underset{\mathbf{u} \in \mathbb{R}^n}{\text{Argmin}} \left\{ \|\mathbf{u} - \mathbf{c}\|_2^2 : \mathbf{u} \in \mathcal{F} \right\}.^2 \quad (4)$$

Then, the projected power iterations take the form

$$\mathbf{x}^{(k)} \in \mathcal{P}(\mathbf{A}\mathbf{x}^{(k-1)}), \quad k = 1, 2, \dots \quad (5)$$

As the following proposition shows, for every  $k \geq 1$ , the problem of computing  $\mathbf{x}^{(k)}$  in (5) boils down to that of finding the indices that correspond to the  $n/2$  largest entries of  $\mathbf{A}\mathbf{x}^{(k-1)}$ , which can be done efficiently.

<sup>1</sup> This is also termed *strong consistency* in the literature; see [29].

<sup>2</sup> By convention, we use the symbol “Argmin” to denote the solution set of the associated minimization problem. When the solution set is known to be a singleton, we use the symbol “argmin” to denote the unique solution in the set.

**Proposition 1** For any  $\mathbf{c} \in \mathbb{R}^n$ , it holds that  $\mathbf{v} \in \mathcal{P}(\mathbf{c})$  if and only if

$$v_\ell = \begin{cases} 1, & \ell \in \mathcal{I}, \\ -1, & \ell \in [n] \setminus \mathcal{I}, \end{cases} \quad (6)$$

where  $\mathcal{I} \subset [n]$  satisfies  $|\mathcal{I}| = n/2$  and  $c_i \geq c_j$  for all  $i \in \mathcal{I}$  and  $j \in [n] \setminus \mathcal{I}$ .

**Proof** By (4), we have

$$\mathcal{P}(\mathbf{c}) = \underset{\mathbf{u} \in \mathbb{R}^n}{\text{Argmin}} \left\{ \|\mathbf{u} - \mathbf{c}\|_2^2 : \mathbf{u} \in \mathcal{F} \right\} = \underset{\mathbf{u} \in \mathbb{R}^n}{\text{Argmax}} \left\{ \mathbf{c}^T \mathbf{u} : \mathbf{u} \in \mathcal{F} \right\},$$

where the second equality is due to the fact that  $\|\mathbf{u}\|_2^2 = n$  for all  $\mathbf{u} \in \mathcal{F}$ . The desired formula (6) follows immediately.  $\square$

Despite its simplicity, the method of projected power iteration may not be effective for solving Problem (MLE) unless a proper initial point  $\mathbf{x}^{(0)}$  is available. Thus, we need an additional iterative procedure, which is usually referred to as the *method of orthogonal iteration* (see, e.g., [20]) and constitutes the second ingredient in our approach, to obtain a good initial point for the projected power iterations. The method of orthogonal iteration starts with a matrix  $\mathbf{Q}^{(0)} \in \mathbb{R}^{n \times 2}$  with orthonormal columns. In iteration  $k \geq 1$ , it computes the QR decomposition of  $\mathbf{A} \mathbf{Q}^{(k-1)}$ ; i.e.,

$$\mathbf{A} \mathbf{Q}^{(k-1)} = \mathbf{Q}^{(k)} \mathbf{R}^{(k)},$$

where  $\mathbf{Q}^{(k)} \in \mathbb{R}^{n \times 2}$  has orthonormal columns and  $\mathbf{R}^{(k)} \in \mathbb{R}^{2 \times 2}$  is upper triangular. It is known that the distance between the subspace spanned by the columns of  $\mathbf{Q}^{(k)}$  and the invariant subspace of  $\mathbf{A}$  that corresponds to its first two dominant eigenvalues converges linearly to 0 as  $k \rightarrow \infty$ ; see, e.g., [20, Theorem 8.2.2]. For our purpose, we only perform  $N$  orthogonal iterations, where  $N$  is an input parameter of the algorithm. Then, we apply Ritz acceleration (see, e.g., [20, Chapter 8.3.7]) to the last iterate, which amounts to computing the eigenvalue decomposition of  $\mathbf{Q}^{(N)T} \mathbf{A} \mathbf{Q}^{(N)}$ ; i.e.,

$$\mathbf{Q}^{(N)T} \mathbf{A} \mathbf{Q}^{(N)} = \mathbf{H}^{(N)} \mathbf{D}^{(N)} \mathbf{H}^{(N)T},$$

where  $\mathbf{H}^{(N)} \in \mathbb{R}^{2 \times 2}$  is orthogonal and  $\mathbf{D}^{(N)} = \text{diag}(d_1^{(N)}, d_2^{(N)})$  is diagonal with  $|d_1^{(N)}| \geq |d_2^{(N)}|$ . Finally, we extract the column of  $\tilde{\mathbf{Q}}^{(N)} = \mathbf{Q}^{(N)} \mathbf{H}^{(N)}$  that corresponds to the smaller eigenvalue of  $\mathbf{Q}^{(N)T} \mathbf{A} \mathbf{Q}^{(N)}$  to construct a suitable initial point for the projected power iterations.

We now summarize our proposed method for solving Problem (MLE) in Algorithm 1. It starts with a matrix  $\mathbf{Y} \in \mathbb{R}^{n \times 2}$ , whose entries are generated independently and identically from the standard normal distribution. In the first stage (lines 2–9 of Algorithm 1), the algorithm performs  $N$  orthogonal iterations with the initial iterate  $\mathbf{Q}^{(0)}$ , which is obtained by orthonormalizing the columns of  $\mathbf{Y}$  via  $\mathbf{Q}^{(0)} = \mathbf{Y}(\mathbf{Y}^T \mathbf{Y})^{-1/2}$ , and then applies the Ritz acceleration. In the second stage (lines 12–19 of Algorithm 1), the algorithm employs projected power iterations to refine the initial iterate

**Algorithm 1:** A Two-Stage Algorithm for Solving Problem (MLE)

---

**Input:** adjacency matrix  $A$ , positive integer  $N$   
**Output:** label vector  $\hat{x}$

- 1 choose a matrix  $Y \in \mathbb{R}^{n \times 2}$ , whose entries are generated independently and identically from the standard normal distribution  
 /\* stage 1: method of orthogonal iteration with Ritz acceleration \*/
- 2 set  $Q^{(0)} \leftarrow Y(Y^T Y)^{-1/2}$
- 3 **for**  $k = 1, 2, \dots, N$  **do**
- 4   set  $Z^{(k)} \leftarrow A Q^{(k-1)}$
- 5   compute the QR decomposition  $Z^{(k)} = Q^{(k)} R^{(k)}$
- 6 **end**
- 7 set  $S^{(N)} \leftarrow Q^{(N)T} A Q^{(N)}$
- 8 compute the eigen-decomposition  $S^{(N)} = H^{(N)} D^{(N)} H^{(N)T}$ , where  $D^{(N)} = \text{diag}(d_1^{(N)}, d_2^{(N)})$   
 such that  $|d_1^{(N)}| \geq |d_2^{(N)}|$
- 9 set  $\bar{Q}^{(N)} \leftarrow Q^{(N)} H^{(N)}$
- 10 set  $\bar{y}$  to be the  $i^*$ -th column of  $\bar{Q}^{(N)}$ , where  $i^* = \text{argmin}_{i \in \{1, 2\}} d_i^{(N)}$
- 11 set  $y \leftarrow \bar{y} - (1^T \bar{y} / n) \mathbf{1}$   
 /\* stage 2: method of projected power iteration \*/
- 12 set  $x^{(0)} \leftarrow \sqrt{n} y / \|y\|_2$
- 13 **for**  $k = 1, 2, \dots$  **do**
- 14   set  $z^{(k)} \leftarrow A x^{(k-1)}$
- 15   set  $x^{(k)} \leftarrow \mathcal{P}(z^{(k)})$
- 16   **if**  $x^{(k)} = x^{(k-1)}$  **then**
- 17     terminate and output  $\hat{x} = x^{(k)}$
- 18   **end**
- 19 **end**

---

$x^{(0)}$ , which is constructed from a suitable column of  $\bar{Q}^{(N)}$ . The algorithm terminates when  $x^{(k)} = x^{(k-1)}$  for some  $k$  in the second stage, at which point it outputs  $x^{(k)}$ .

We next present the main result of this paper, which shows that Algorithm 1 achieves exact recovery at the information-theoretic limit and also provides explicit iteration complexity bounds for Algorithm 1.

**Theorem 1** *Let  $A$  be the adjacency matrix of a random graph generated according to the binary symmetric SBM with parameters  $(n, p, q)$  and label  $x^*$ , where  $p, q$  satisfy (2) for some constants  $\alpha > \beta > 0$ . Set  $N = \Theta(\log n / \log \log n)$ . If  $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{2}$ , then for all sufficiently large  $n$ , the following statement holds with probability at least  $1 - n^{-\Omega(1)}$ : Algorithm 1 takes  $\mathcal{O}(\log n / \log \log n)$  orthogonal iterations and  $\mathcal{O}(\log n / \log \log n)$  projected power iterations to output  $x^*$  or  $-x^*$ . Here, the probability is taken with respect to the random choices in  $A$  and in Algorithm 1.*

We remark that the value of  $N$  in Theorem 1 can be explicitly given; see (69). Equipped with Theorem 1, it is not hard to derive the total computational cost of Algorithm 1. Indeed, since  $Z^{(k)} \in \mathbb{R}^{n \times 2}$  and  $S^{(N)} \in \mathbb{R}^{2 \times 2}$ , the QR decomposition in line 5 can be found in  $\mathcal{O}(n)$  time, while the eigen-decomposition in line 8 can be found in  $\mathcal{O}(1)$  time [34]. Moreover, by Proposition 1, the projection  $\mathcal{P}(z^{(k)})$  in line 15



can be found by first identifying the  $(n/2)$ -th largest element  $\bar{z}^{(k)}$  of  $\mathbf{z}^{(k)}$ , which can be done in  $\mathcal{O}(n)$  time [9], and then comparing each element of  $\mathbf{z}^{(k)}$  with  $\bar{z}^{(k)}$ , which can be trivially done in  $\mathcal{O}(n)$  time. Now, the remaining dominant computational cost is that of computing matrix-vector products of the form  $\mathbf{A}\mathbf{v}$ , where  $\mathbf{A} \in \mathbb{R}^{n \times n}$  is generated according to the setting of Theorem 1 and  $\mathbf{v} \in \mathbb{R}^n$  is arbitrary. Using a simple concentration argument, one can show that the number of non-zero entries in  $\mathbf{A}$  is  $\mathcal{O}(n \log n)$  with high probability; see Sect. 3. Hence, with high probability, the cost of computing  $\mathbf{A}\mathbf{v}$  is  $\mathcal{O}(n \log n)$  for any  $\mathbf{v}$ . Putting the above time bounds together and using the iteration bounds established in Theorem 1, we obtain the following corollary.

**Corollary 1** *Consider the setting of Theorem 1. If  $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{2}$ , then for all sufficiently large  $n$ , the probability that Algorithm 1 outputs  $\mathbf{x}^*$  or  $-\mathbf{x}^*$  in  $\mathcal{O}(n \log^2 n / \log \log n)$  time is at least  $1 - n^{-\Omega(1)}$ .*

To put the above results in perspective, let us make the following remarks:

- (a) While Problem (MLE) is known to be NP-hard in the worst case, the assumption that the adjacency matrix  $\mathbf{A}$  arises from the binary symmetric SBM in Definition 1 allows us to conduct an average-case analysis of Algorithm 1. In particular, if the constants  $\alpha, \beta > 0$  satisfy  $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{2}$ , which is known to be the information-theoretic limit for exact recovery [2,29], then with high probability,  $\mathbf{x}^*$  and  $-\mathbf{x}^*$  are the only optimal solutions of Problem (MLE) [2]. Moreover, Corollary 1 shows that with high probability, Algorithm 1 computes an optimal solution of Problem (MLE) in nearly-linear time. As such, Algorithm 1 is more efficient than SDP-based methods (see, e.g., [7,21]). The time bound for Algorithm 1 is also competitive with those for some of the most efficient methods in the literature (see, e.g., [4,17,29,38]) under the setting of Theorem 1.
- (b) In the recent work [3], Abbe et al. showed that the vanilla spectral method, which first computes an exact eigenvector  $\mathbf{u}_2$  associated with the second largest eigenvalue of  $\mathbf{A}$  and then returns  $\hat{\mathbf{x}} = \text{sgn}(\mathbf{u}_2)$  as the label vector, achieves exact recovery under the setting of Theorem 1. Conceptually, the method can be implemented in the framework of Algorithm 1 as follows. First, by performing  $N \rightarrow \infty$  orthogonal iterations, we obtain a limit point  $\tilde{\mathbf{Q}}^{(\infty)}$ , which can be used to construct  $\mathbf{u}_2$ ; see [20, Theorem 8.2.2] and compare with lines 10–11 of Algorithm 1. Then, we return the label  $\hat{\mathbf{x}} = \text{sgn}(\mathbf{u}_2)$ . Incidentally, observe that if the label  $\hat{\mathbf{x}}$  coincides (up to sign) with the ground-truth label  $\mathbf{x}^*$ , then  $\mathbf{u}_2$  has exactly  $n/2$  positive entries and  $n/2$  negative entries. This, together with Proposition 1, implies that  $\hat{\mathbf{x}}$  can also be computed by projecting  $\mathbf{u}_2$  onto  $\mathcal{F}$ .

In actual implementation, however, we need to know when to terminate the orthogonal iterations, so that the vanilla spectral method can proceed to the sign-taking step. Unfortunately, the results in [3] do not provide the required termination criterion. By contrast, Theorem 1 shows that the underlying communities can be exactly recovered by first performing  $\mathcal{O}(\log n / \log \log n)$  orthogonal iterations to obtain a coarse approximation  $\mathbf{y}$  of  $\mathbf{u}_2$  and then applying  $\mathcal{O}(\log n / \log \log n)$  projected power iterations to a suitably scaled  $\mathbf{y}$ . As it turns out, by combining our results in Sects. 3.1 and 3.2 with the arguments in [3], we can show that the vanilla

spectral method only needs  $\mathcal{O}(\log n / \log \log n)$  orthogonal iterations before the sign-taking step in order to exactly recover the underlying communities with high probability; see Sect. 3.4.2. It is worth noting that this gives the best complexity bound known to date for the vanilla spectral method in the context of exact community recovery in the binary symmetric SBM. This further demonstrates the power of our approach.

### 3 Proof of the main result

In this section, we prove our main result (i.e., Theorem 1) concerning the recovery performance and iteration complexity of Algorithm 1. This involves establishing some key properties of the orthogonal iterations and projected power iterations, which will be accomplished in Sects. 3.1 and 3.2, respectively.

#### 3.1 Analysis of the method of orthogonal iteration

Our main goal in this sub-section is to provide a probabilistic analysis of the convergence behavior of the orthogonal iterations deployed in the first stage of Algorithm 1. Such an analysis is not only useful for proving Theorem 1 but may also be of independent interest. To proceed, let us introduce some further notation that will be used in the sequel. Let  $A$  be as in Theorem 1 and consider its eigenvalue decomposition  $A = U \Lambda U^T$ , where  $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$  with  $\lambda_1 \geq \dots \geq \lambda_n$  and  $U = [u_1, \dots, u_n]$ . We write

$$\begin{aligned} \Lambda_\alpha &:= \text{diag}(\lambda_1, \lambda_2), & \Lambda_\beta &:= \text{diag}(\lambda_3, \dots, \lambda_n), \\ U_\alpha &:= [u_1, u_2], & U_\beta &:= [u_3, \dots, u_n]. \end{aligned} \quad (7)$$

Moreover, we define, for every  $k \geq 0$ ,

$$P^{(k)} := U^T Q^{(k)}, \quad V^{(k)} := U_\alpha^T Q^{(k)}, \quad W^{(k)} := U_\beta^T Q^{(k)}, \quad (8)$$

where the sequence  $\{Q^{(k)}\}_{k \geq 1}$  is generated by Algorithm 1. Note that for all  $k \geq 0$ ,  $P^{(k)} \in \mathbb{R}^{n \times 2}$ ,  $V^{(k)} \in \mathbb{R}^{2 \times 2}$ ,  $W^{(k)} \in \mathbb{R}^{(n-2) \times 2}$ , and  $P^{(k)}$  has orthonormal columns. Besides, by the CS decomposition (see, e.g., [20, Theorem 2.5.2]), we have

$$\sigma_{\min}^2(V^{(k)}) + \sigma_{\max}^2(W^{(k)}) = 1, \quad k = 0, 1, \dots \quad (9)$$

It is known that the quantity  $\sqrt{1 - \sigma_{\min}^2(V^{(0)})}$  measures the distance between the subspaces spanned by the columns of  $U_\alpha$  and  $Q^{(0)}$ ; see, e.g., [20, Theorem 2.5.1]. To bound this distance, we prove the following result.

**Lemma 1** *For all  $n \geq 6$ , it holds with probability at least  $1 - 4\sqrt{\log n/n}$  that*

$$\sigma_{\min}(V^{(0)}) \geq \frac{1}{\sqrt{n^2 + 1}}. \quad (10)$$

**Proof** Recall from Algorithm 1 that  $\mathbf{Q}^{(0)} = \mathbf{Y}(\mathbf{Y}^T \mathbf{Y})^{-1/2}$ , where  $\mathbf{Y} \in \mathbb{R}^{n \times 2}$  is a random matrix whose entries are i.i.d. standard normal random variables. By the definition of  $\mathbf{U}_\alpha$ , we have  $\mathbf{U}_\alpha = \mathbf{U}\mathbf{E}$ , where  $\mathbf{E} := (\mathbf{e}_1, \mathbf{e}_2)$ . This, together with  $\mathbf{V}^{(0)} = \mathbf{U}_\alpha^T \mathbf{Q}^{(0)}$ ,  $\mathbf{Q}^{(0)} = \mathbf{Y}(\mathbf{Y}^T \mathbf{Y})^{-1/2}$ ,  $\mathbf{U}\mathbf{U}^T = \mathbf{I}$ , and the orthogonal invariance of the normal distribution, yields

$$\mathbf{V}^{(0)} = \mathbf{E}^T (\mathbf{U}^T \mathbf{Y}) \left( (\mathbf{U}^T \mathbf{Y})^T (\mathbf{U}^T \mathbf{Y}) \right)^{-1/2} \stackrel{d}{=} \mathbf{E}^T \mathbf{Y} (\mathbf{Y}^T \mathbf{Y})^{-1/2} = \mathbf{E}^T \mathbf{Q}^{(0)}.$$

It then follows that

$$\begin{aligned} \mathbb{P} \left( \sigma_{\min} \left( \mathbf{V}^{(0)} \right) \leq a \right) &= \mathbb{P} \left( \sigma_{\min} \left( \mathbf{E}^T \mathbf{Q}^{(0)} \right) \leq a \right) \\ &= \mathbb{P} \left( \lambda_{\min} \left( \mathbf{Q}^{(0)T} \mathbf{E} \mathbf{E}^T \mathbf{Q}^{(0)} \right) \leq a^2 \right) = \int_0^{a^2} p(x) dx, \end{aligned} \quad (11)$$

where  $p(\cdot)$  is the probability density function of  $\lambda_{\min} \left( \mathbf{Q}^{(0)T} \mathbf{E} \mathbf{E}^T \mathbf{Q}^{(0)} \right)$ . By [5, eq. (6)],  $p(\cdot)$  takes the form

$$p(x) = \frac{(n-2)(n-3)}{4} \cdot \frac{(1-x)^{n-3}}{x} \int_0^1 y \left( 1 + \frac{1-x}{x} y \right)^{-\frac{1}{2}} (1-y)^{\frac{n-5}{2}} dy \quad (12)$$

for any  $x \in (0, 1]$ . This allows us to derive an upper bound on the integral in (11). Indeed, using the Cauchy-Schwarz inequality, we get

$$\begin{aligned} &\int_0^1 y \left( 1 + \frac{1-x}{x} y \right)^{-\frac{1}{2}} (1-y)^{\frac{n-5}{2}} dy \\ &\leq \left( \int_0^1 \left( 1 + \frac{1-x}{x} y \right)^{-1} dy \right)^{\frac{1}{2}} \cdot \left( \int_0^1 y^2 (1-y)^{n-5} dy \right)^{\frac{1}{2}} \\ &= \left( \frac{x}{1-x} \cdot \log \left( \frac{1}{x} \right) \right)^{\frac{1}{2}} \cdot \left( \frac{2}{(n-2)(n-3)(n-4)} \right)^{\frac{1}{2}}. \end{aligned}$$

Together with (12), this implies that for all  $n \geq 6$ ,

$$\begin{aligned} \int_0^{a^2} p(x) dx &\leq \frac{\sqrt{2}}{4} \cdot \frac{\sqrt{(n-2)(n-3)}}{\sqrt{n-4}} \cdot \int_0^{a^2} \sqrt{\frac{1}{x} \log \left( \frac{1}{x} \right)} (1-x)^{n-\frac{7}{2}} dx \\ &\leq \frac{\sqrt{2n}}{4} \cdot \int_0^{a^2} \sqrt{\frac{1}{x} \log \left( \frac{1}{x} \right)} dx, \end{aligned} \quad (13)$$

where the second inequality follows from the fact that for all  $n \geq 6$ , we have  $(n-2)(n-3) \leq n(n-4)$  and  $0 \leq (1-x)^{n-7/2} \leq 1$  for every  $0 \leq x \leq 1$ . By letting

$x = e^{-2z^2}$ , we obtain

$$\int_0^{a^2} \sqrt{\frac{1}{x} \log\left(\frac{1}{x}\right)} dx = 4\sqrt{2} \int_{\sqrt{\log\left(\frac{1}{a}\right)}}^{\infty} z^2 e^{-z^2} dz. \quad (14)$$

Moreover, using integration by parts, we compute

$$\int_{\sqrt{\log\left(\frac{1}{a}\right)}}^{\infty} z^2 e^{-z^2} dz = \frac{a}{2} \sqrt{\log\left(\frac{1}{a}\right)} + \frac{1}{2} \int_{\sqrt{\log\left(\frac{1}{a}\right)}}^{\infty} e^{-z^2} dz. \quad (15)$$

Notice that for any  $a \leq 1/e$ , we have  $\sqrt{\log(1/a)} \geq 1$ , which leads to

$$\int_{\sqrt{\log\left(\frac{1}{a}\right)}}^{\infty} e^{-z^2} dz \leq \int_{\sqrt{\log\left(\frac{1}{a}\right)}}^{\infty} z^2 e^{-z^2} dz. \quad (16)$$

Combining (15) and (16), we have that for any  $a \leq 1/e$ ,

$$\int_{\sqrt{\log\left(\frac{1}{a}\right)}}^{\infty} z^2 e^{-z^2} dz \leq a \sqrt{\log\left(\frac{1}{a}\right)}.$$

This, together with (11), (13), and (14), yields

$$\mathbb{P}\left(\sigma_{\min}\left(\mathbf{V}^{(0)}\right) \leq a\right) \leq 2\sqrt{na} \sqrt{\log\left(\frac{1}{a}\right)}$$

for all  $n \geq 6$  and  $a \leq 1/e$ . By letting  $a = 1/\sqrt{n^2 + 1}$  (which satisfies  $a \leq 1/e$  for any  $n \geq 6$ ) in the above inequality, we obtain

$$\mathbb{P}\left(\sigma_{\min}\left(\mathbf{V}^{(0)}\right) \leq \frac{1}{\sqrt{n^2 + 1}}\right) \leq \frac{2\sqrt{n}}{\sqrt{n^2 + 1}} \sqrt{\log\left(\sqrt{n^2 + 1}\right)} \leq 4\sqrt{\frac{\log n}{n}},$$

which implies Lemma 1 as desired.  $\square$

Next, we present a spectral bound on the deviation of  $\mathbf{A}$  from its mean. It is a direct consequence of [22, Theorem 5.2] and thus we omit its proof.

**Lemma 2** *There exist constants  $c_1 \geq 1$  and  $c_2 > 0$ , whose values depend only on  $\alpha$  and  $\beta$ , such that*

$$\|\mathbf{A} - \mathbb{E}[\mathbf{A}]\| \leq c_1 \sqrt{\log n} \quad (17)$$

*holds with probability at least  $1 - c_2 n^{-3}$ .*

Based on Lemma 2, we can establish the following corollary, which provides estimates on the eigenvalues and eigenvectors of  $\mathbf{A}$ .

**Corollary 2** *With probability at least  $1 - c_2 n^{-3}$ , the following statements hold:*

$$\frac{\alpha + \beta}{2} \log n - c_1 \sqrt{\log n} \leq \lambda_1 \leq \frac{\alpha + \beta}{2} \log n + c_1 \sqrt{\log n}, \quad (18)$$

$$\frac{\alpha - \beta}{2} \log n - c_1 \sqrt{\log n} \leq \lambda_2 \leq \frac{\alpha - \beta}{2} \log n + c_1 \sqrt{\log n}, \quad (19)$$

$$|\lambda_i| \leq c_1 \sqrt{\log n}, \quad i = 3, \dots, n, \quad (20)$$

$$\min_{\theta \in \{\pm 1\}} \left\| \theta \mathbf{u}_1 - \frac{\mathbf{1}}{\sqrt{n}} \right\|_2 \leq \frac{c_3}{\sqrt{\log n}}, \quad (21)$$

$$\min_{\theta \in \{\pm 1\}} \left\| \theta \mathbf{u}_2 - \frac{\mathbf{x}^*}{\sqrt{n}} \right\|_2 \leq \frac{c_3}{\sqrt{\log n}}, \quad (22)$$

where  $c_1, c_2$  are the constants in Lemma 2 and  $c_3 := 2\sqrt{2}c_1 / \min\{\beta, (\alpha - \beta)/2\}$ .

**Proof** Suppose that the statement in Lemma 2 holds, which happens with probability at least  $1 - c_2 n^{-3}$ . Let  $v_1 \geq v_2 \geq \dots \geq v_n$  be the eigenvalues of  $\mathbb{E}[\mathbf{A}]$ . It follows from Weyl's inequality (see, e.g., [36, Theorem 4.5.3]) that

$$|\lambda_i - v_i| \leq \|\mathbf{A} - \mathbb{E}[\mathbf{A}]\|, \quad i = 1, 2, \dots, n. \quad (23)$$

According to the binary symmetric SBM in Definition 1, we have

$$\mathbb{E}[\mathbf{A}] = \frac{p+q}{2} \mathbf{1}\mathbf{1}^T + \frac{p-q}{2} \mathbf{x}^* \mathbf{x}^{*T}. \quad (24)$$

Since  $\mathbf{1}^T \mathbf{x}^* = 0$  and  $\|\mathbf{1}\|_2^2 = \|\mathbf{x}^*\|_2^2 = n$ , we see that  $\mathbb{E}[\mathbf{A}]$  is a rank-2 matrix with  $\mathbf{1}$  and  $\mathbf{x}^*$  being the eigenvectors associated with the largest and second largest eigenvalues, respectively. Using (2), we can compute

$$v_1 = \frac{\alpha + \beta}{2} \log n, \quad v_2 = \frac{\alpha - \beta}{2} \log n, \quad v_i = 0, \quad i = 3, \dots, n. \quad (25)$$

By (23), (25), and Lemma 2, the desired results (18)–(20) are immediate. Moreover, it follows from (25) that

$$\delta_1 := \min_{i \neq 1} |v_1 - v_i| = \beta \log n, \quad \delta_2 := \min_{i \neq 2} |v_2 - v_i| = \min \left\{ \beta, \frac{\alpha - \beta}{2} \right\} \log n.$$

This, together with the Davis-Kahan theorem (see, e.g., [36, Theorem 4.5.5]) and Lemma 2, yields

$$\min_{\theta \in \{\pm 1\}} \left\| \theta \mathbf{u}_1 - \frac{\mathbf{1}}{\sqrt{n}} \right\|_2 \leq \frac{2\sqrt{2} \|\mathbf{A} - \mathbb{E}[\mathbf{A}]\|}{\delta_1} \leq \frac{2\sqrt{2}c_1}{\beta \sqrt{\log n}} \leq \frac{c_3}{\sqrt{\log n}}$$

and

$$\min_{\theta \in \{\pm 1\}} \left\| \theta \mathbf{u}_2 - \frac{\mathbf{x}^*}{\sqrt{n}} \right\|_2 \leq \frac{2\sqrt{2}\|\mathbf{A} - \mathbb{E}[\mathbf{A}]\|}{\delta_2} \leq \frac{2\sqrt{2}c_1}{\min\left\{\beta, \frac{\alpha-\beta}{2}\right\}\sqrt{\log n}} = \frac{c_3}{\sqrt{\log n}}.$$

The proof is then completed.  $\square$

Now, we are ready to analyze the convergence of the orthogonal iterations. From Algorithm 1, one can verify by induction that for every  $k \geq 1$ ,

$$\mathbf{A}^k \mathbf{Q}^{(0)} = \mathbf{Q}^{(k)} \tilde{\mathbf{R}}^{(k)}, \quad (26)$$

where  $\tilde{\mathbf{R}}^{(k)} := \mathbf{R}^{(k)} \mathbf{R}^{(k-1)} \dots \mathbf{R}^{(1)}$  with  $\tilde{\mathbf{R}}^{(1)} = \mathbf{R}^{(1)}$ . This, together with (7)–(8) and  $\mathbf{A} = \mathbf{U} \mathbf{A} \mathbf{U}^T$ , yields

$$\mathbf{A}_\alpha^k \mathbf{V}^{(0)} = \mathbf{V}^{(k)} \tilde{\mathbf{R}}^{(k)}, \quad \mathbf{A}_\beta^k \mathbf{W}^{(0)} = \mathbf{W}^{(k)} \tilde{\mathbf{R}}^{(k)}. \quad (27)$$

Suppose that (10) and  $\lambda_1 \geq \lambda_2 > 0$  hold. Then,  $\mathbf{A}_\alpha^k \mathbf{V}^{(0)}$  is non-singular, which implies that the square matrix  $\mathbf{V}^{(k)}$  is invertible and  $\tilde{\mathbf{R}}^{(k)} = (\mathbf{V}^{(k)})^{-1} \mathbf{A}_\alpha^k \mathbf{V}^{(0)}$ . Together with (27), this leads to

$$\mathbf{K}^{(k)} = \mathbf{A}_\beta^k \mathbf{K}^{(0)} \mathbf{A}_\alpha^{-k}, \quad (28)$$

where we define  $\mathbf{K}^{(k)} := \mathbf{W}^{(k)} (\mathbf{V}^{(k)})^{-1}$  for all  $k \geq 0$ . Let  $\mathbf{k}_1^{(k)}$  and  $\mathbf{k}_2^{(k)}$  be the first and second column of  $\mathbf{K}^{(k)}$ , respectively. The following result characterizes the convergence rates of  $\mathbf{k}_1^{(k)}$  and  $\mathbf{k}_2^{(k)}$ .

**Proposition 2** Suppose that  $n \geq \exp(16c_1^2/(\alpha - \beta)^2)$  and that (10), (18)–(20), and  $\lambda_2 > 0$  hold. Then, for every  $k \geq 0$ , it holds that

$$\|\mathbf{k}_1^{(k)}\|_2 \leq n \left( \frac{4c_1}{(\alpha + \beta)\sqrt{\log n}} \right)^k \quad \text{and} \quad \|\mathbf{k}_2^{(k)}\|_2 \leq n \left( \frac{4c_1}{(\alpha - \beta)\sqrt{\log n}} \right)^k. \quad (29)$$

**Proof** It follows from (28) and  $\lambda_1 \geq \lambda_2 > 0$  that

$$\|\mathbf{k}_1^{(k)}\|_2 \leq \left( \frac{\bar{\lambda}}{\lambda_1} \right)^k \|\mathbf{K}^{(0)}\|, \quad \|\mathbf{k}_2^{(k)}\|_2 \leq \left( \frac{\bar{\lambda}}{\lambda_2} \right)^k \|\mathbf{K}^{(0)}\|, \quad (30)$$

where  $\bar{\lambda} = \max\{|\lambda_3|, \dots, |\lambda_n|\}$ . By (20), we have  $\bar{\lambda} \leq c_1 \sqrt{\log n}$ . Moreover,

$$\|\mathbf{K}^{(0)}\| = \|\mathbf{W}^{(0)} (\mathbf{V}^{(0)})^{-1}\| \leq \frac{\sigma_{\max}(\mathbf{W}^{(0)})}{\sigma_{\min}(\mathbf{V}^{(0)})} = \sqrt{\frac{1}{\sigma_{\min}^2(\mathbf{V}^{(0)})} - 1} \leq n,$$

where the second equality follows from (9) and the last inequality is due to (10). These, together with (18) and (19), yield

$$\begin{aligned}\|\mathbf{k}_1^{(k)}\|_2 &\leq n \left( \frac{2c_1\sqrt{\log n}}{(\alpha + \beta) \log n - 2c_1\sqrt{\log n}} \right)^k, \\ \|\mathbf{k}_2^{(k)}\|_2 &\leq n \left( \frac{2c_1\sqrt{\log n}}{(\alpha - \beta) \log n - 2c_1\sqrt{\log n}} \right)^k.\end{aligned}$$

In particular, the desired result (29) holds for all  $n$  satisfying  $4c_1\sqrt{\log n} \leq (\alpha - \beta) \log n$ , or equivalently,  $n \geq \exp(16c_1^2/(\alpha - \beta)^2)$ .  $\square$

Next, we study the effect of Ritz acceleration. Let  $\{\bar{\mathbf{Q}}^{(k)}\}_{k \geq 1}^{N-1}$  be an auxiliary sequence constructed via  $\bar{\mathbf{Q}}^{(k)} = \mathbf{Q}^{(k)} \mathbf{H}^{(k)}$ , where  $\mathbf{H}^{(k)}$  is obtained from the eigen-decomposition of  $\mathbf{Q}^{(k)T} \mathbf{A} \mathbf{Q}^{(k)}$ —i.e.,  $\mathbf{Q}^{(k)T} \mathbf{A} \mathbf{Q}^{(k)} = \mathbf{H}^{(k)} \mathbf{D}^{(k)} \mathbf{H}^{(k)T}$  with  $\mathbf{D}^{(k)} = \text{diag}(d_1^{(k)}, d_2^{(k)})$  and  $|d_1^{(k)}| \geq |d_2^{(k)}|$ . We are interested in relating the Ritz eigenvalues  $d_1^{(k)}, d_2^{(k)}$  to the eigenvalues  $\lambda_1, \lambda_2$  of  $\mathbf{A}$ . Towards that end, we define

$$\bar{d}^{(k)} := \max\{d_1^{(k)}, d_2^{(k)}\}, \quad \underline{d}^{(k)} := \min\{d_1^{(k)}, d_2^{(k)}\}.$$

Then, we have the following estimates.

**Proposition 3** *For every  $k \geq 0$ , it holds that*

$$\lambda_1 \geq \bar{d}^{(k)} \geq \lambda_1 - 2\|\mathbf{A}\| \cdot \|\mathbf{k}_1^{(k)}\|_2^2, \quad \lambda_2 \geq \underline{d}^{(k)}. \quad (31)$$

**Proof** For ease of exposition, we shall omit the superscript  $(k)$  in  $d_1^{(k)}, d_2^{(k)}, \bar{d}^{(k)}, \underline{d}^{(k)}, \mathbf{Q}^{(k)}, \mathbf{P}^{(k)}, \mathbf{V}^{(k)}, \mathbf{W}^{(k)}, \mathbf{k}_1^{(k)}$ , and  $\mathbf{K}^{(k)}$  throughout the proof. Since  $\bar{d} \geq \underline{d}$  are the eigenvalues of  $\mathbf{Q}^T \mathbf{A} \mathbf{Q}$  and  $\lambda_1 \geq \lambda_2$  are the largest two eigenvalues of  $\mathbf{A}$ , we have  $\lambda_1 \geq \bar{d}$  and  $\lambda_2 \geq \underline{d}$  from the Courant-Fischer minimax theorem (see, e.g., [20, Theorem 8.1.2]). Moreover, using (7)–(8) and the definition of  $\mathbf{K}$  and letting  $\hat{\mathbf{e}}_1 = (1, 0)$ , we have

$$(\mathbf{V}^{-1} \hat{\mathbf{e}}_1)^T \mathbf{Q}^T \mathbf{A} \mathbf{Q} (\mathbf{V}^{-1} \hat{\mathbf{e}}_1) = \hat{\mathbf{e}}_1^T (\mathbf{A}_\alpha + \mathbf{K}^T \mathbf{A}_\beta \mathbf{K}) \hat{\mathbf{e}}_1 = \lambda_1 + \mathbf{k}_1^T \mathbf{A}_\beta \mathbf{k}_1$$

and

$$\|\mathbf{V}^{-1} \hat{\mathbf{e}}_1\|_2^2 = (\mathbf{V}^{-1} \hat{\mathbf{e}}_1)^T \mathbf{P}^T \mathbf{P} \mathbf{V}^{-1} \hat{\mathbf{e}}_1 = \hat{\mathbf{e}}_1^T (\mathbf{I} + \mathbf{K}^T \mathbf{K}) \hat{\mathbf{e}}_1 = 1 + \|\mathbf{k}_1\|_2^2, \quad (32)$$

where the first equality in (32) is due to the fact that  $\mathbf{P}$  has orthonormal columns. Now, since  $\bar{d}$  is the largest eigenvalue of  $\mathbf{Q}^T \mathbf{A} \mathbf{Q}$ , we have

$$\bar{d} \geq \frac{(\mathbf{V}^{-1} \hat{\mathbf{e}}_1)^T \mathbf{Q}^T \mathbf{A} \mathbf{Q} (\mathbf{V}^{-1} \hat{\mathbf{e}}_1)}{\|\mathbf{V}^{-1} \hat{\mathbf{e}}_1\|_2^2} = \frac{\lambda_1 + \mathbf{k}_1^T \mathbf{A}_\beta \mathbf{k}_1}{1 + \|\mathbf{k}_1\|_2^2},$$

which leads to

$$\bar{d} - \lambda_1 \geq \frac{-\lambda_1 \|\mathbf{k}_1\|_2^2 + \mathbf{k}_1^T \mathbf{A}_\beta \mathbf{k}_1}{1 + \|\mathbf{k}_1\|_2^2} \geq -\frac{2\|\mathbf{A}\| \cdot \|\mathbf{k}_1\|_2^2}{1 + \|\mathbf{k}_1\|_2^2} \geq -2\|\mathbf{A}\| \cdot \|\mathbf{k}_1\|_2^2.$$

The proof is then completed.  $\square$

Now, let  $\bar{\mathbf{h}}^{(k)}, \underline{\mathbf{h}}^{(k)}$  denote the eigenvectors associated with the eigenvalues  $\bar{d}^{(k)}, \underline{d}^{(k)}$ , respectively. Furthermore, we define  $\bar{\mathbf{q}}^{(k)} := \mathbf{Q}^{(k)} \bar{\mathbf{h}}^{(k)}$  and  $\underline{\mathbf{q}}^{(k)} := \mathbf{Q}^{(k)} \underline{\mathbf{h}}^{(k)}$ . Equipped with Propositions 2 and 3, we can establish the convergence rate of the orthogonal iterations.

**Theorem 2** Let  $c_1 \geq 1$  and  $c_2 > 0$  be the constants in Lemma 2. Suppose that

$$n \geq \max \left\{ \exp \left( \frac{16c_1^2}{\beta^2} \right), 6 \right\}. \quad (33)$$

Then, it holds with probability at least  $1 - c_2 n^{-3} - 4\sqrt{\log n/n}$  that

$$\min_{\theta \in \{\pm 1\}} \|\bar{\mathbf{q}}^{(k)} - \theta \mathbf{u}_1\|_2 \leq n \left( \frac{8\alpha}{\beta} + 10 \right) \left( \frac{4c_1}{(\alpha + \beta)\sqrt{\log n}} \right)^k, \quad \forall k \geq 0, \quad (34)$$

$$\min_{\theta \in \{\pm 1\}} \|\underline{\mathbf{q}}^{(k)} - \theta \mathbf{u}_2\|_2 \leq n \left( \frac{16\alpha}{\beta} + 18 \right) \left( \frac{4c_1}{(\alpha - \beta)\sqrt{\log n}} \right)^k, \quad \forall k \geq \bar{K}, \quad (35)$$

where

$$\bar{K} = \left\lceil \frac{2 \log n + \log \left( \frac{8(\alpha + \beta)}{\beta} \right)}{\log \log n + 2 \log \left( \frac{\alpha + \beta}{4c_1} \right)} \right\rceil. \quad (36)$$

**Proof** Again, we shall omit the superscript  $(k)$  throughout the proof for ease of exposition. By definition, we have

$$\mathbf{Q}^T \mathbf{A} \mathbf{Q} \bar{\mathbf{h}} = \bar{d} \bar{\mathbf{h}}, \quad \mathbf{Q}^T \mathbf{A} \mathbf{Q} \underline{\mathbf{h}} = \underline{d} \underline{\mathbf{h}}. \quad (37)$$

Observe that for any  $\mathbf{v} \in \mathbb{R}^2$  with  $\|\mathbf{v}\|_2 = 1$ , we have

$$\begin{aligned} \|\bar{\mathbf{h}} - \text{sgn}(\bar{\mathbf{h}}^T \mathbf{v}) \mathbf{v}\|_2^2 &= 2 - 2\text{sgn}(\bar{\mathbf{h}}^T \mathbf{v}) \bar{\mathbf{h}}^T \mathbf{v} = 2 - 2|\bar{\mathbf{h}}^T \mathbf{v}| \\ &\leq 2 - 2(\bar{\mathbf{h}}^T \mathbf{v})^2 = 2(\underline{\mathbf{h}}^T \mathbf{v})^2, \end{aligned}$$

where the first equality follows from  $\|\bar{\mathbf{h}}\|_2 = 1$ , the second equality uses  $|a| = \text{sgn}(a) \cdot a$  for any  $a \in \mathbb{R}$ , the inequality follows from  $|\bar{\mathbf{h}}^T \mathbf{v}| \leq \|\bar{\mathbf{h}}\|_2 \|\mathbf{v}\|_2 = 1$ , and the last equality is due to  $\|\mathbf{v}\|_2 = 1$  and  $\mathbf{H}$  being orthogonal. This gives

$$\|\bar{\mathbf{h}} - \text{sgn}(\bar{\mathbf{h}}^T \mathbf{v}) \mathbf{v}\|_2 \leq \sqrt{2} |\underline{\mathbf{h}}^T \mathbf{v}| \quad (38)$$



for any  $\mathbf{v} \in \mathbb{R}^2$  with  $\|\mathbf{v}\|_2 = 1$ . Moreover, we obtain from (37) that

$$\begin{aligned} |(\lambda_1 - d)\underline{\mathbf{h}}^T \mathbf{V}^{-1} \hat{\mathbf{e}}_1| &= |\lambda_1 \underline{\mathbf{h}}^T \mathbf{V}^{-1} \hat{\mathbf{e}}_1 - \underline{\mathbf{h}}^T \mathbf{Q}^T \mathbf{A} \mathbf{Q} \mathbf{V}^{-1} \hat{\mathbf{e}}_1| \\ &\leq \|\lambda_1 \mathbf{V}^{-1} \hat{\mathbf{e}}_1 - \mathbf{P}^T \mathbf{A} \mathbf{P} \mathbf{V}^{-1} \hat{\mathbf{e}}_1\|_2 \leq \|\lambda_1 \mathbf{P} \mathbf{V}^{-1} \hat{\mathbf{e}}_1 - \mathbf{A} \mathbf{P} \mathbf{V}^{-1} \hat{\mathbf{e}}_1\|_2 \\ &= \left\| \lambda_1 \begin{bmatrix} \mathbf{I} \\ \mathbf{K} \end{bmatrix} \hat{\mathbf{e}}_1 - \begin{bmatrix} \mathbf{A}_\alpha & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_\beta \end{bmatrix} \begin{bmatrix} \mathbf{I} \\ \mathbf{K} \end{bmatrix} \hat{\mathbf{e}}_1 \right\|_2 = \|\lambda_1 \mathbf{k}_1 - \mathbf{A}_\beta \mathbf{k}_1\|_2 \\ &\leq 2\|\mathbf{A}\| \cdot \|\mathbf{k}_1\|_2, \end{aligned} \quad (39)$$

where the first inequality is due to  $\|\underline{\mathbf{h}}\|_2 = 1$  and  $\mathbf{Q}^T \mathbf{A} \mathbf{Q} = \mathbf{P}^T \mathbf{A} \mathbf{P}$ , the second inequality follows from  $\mathbf{P}^T \mathbf{P} = \mathbf{I}$  and  $\|\mathbf{P}^T \mathbf{v}\|_2 \leq \|\mathbf{v}\|_2$ , and the second equality uses (8) and  $\mathbf{K} = \mathbf{W} \mathbf{V}^{-1}$ . Then, by letting  $\bar{\theta} = \text{sgn}(\bar{\mathbf{h}}^T \mathbf{V}^{-1} \hat{\mathbf{e}}_1)$ , we have

$$\left\| \bar{\mathbf{h}} - \bar{\theta} \frac{\mathbf{V}^{-1} \hat{\mathbf{e}}_1}{\|\mathbf{V}^{-1} \hat{\mathbf{e}}_1\|_2} \right\|_2 \leq \frac{\sqrt{2} |\underline{\mathbf{h}}^T \mathbf{V}^{-1} \hat{\mathbf{e}}_1|}{\|\mathbf{V}^{-1} \hat{\mathbf{e}}_1\|_2} \leq \frac{2\sqrt{2} \|\mathbf{A}\| \cdot \|\mathbf{k}_1\|_2}{|\lambda_1 - d|}, \quad (40)$$

where the first inequality follows from (38) and the second one is due to (39) and  $\|\mathbf{V}^{-1} \hat{\mathbf{e}}_1\|_2 \geq 1$  (implied by (32)). Also, by (8) and  $\mathbf{U}^T \mathbf{u}_1 = \mathbf{e}_1$ , we obtain

$$\|\mathbf{Q} \mathbf{V}^{-1} \hat{\mathbf{e}}_1 - \mathbf{u}_1\|_2 = \|\mathbf{P} \mathbf{V}^{-1} \hat{\mathbf{e}}_1 - \mathbf{e}_1\|_2 = \left\| \begin{bmatrix} \mathbf{I} \\ \mathbf{K} \end{bmatrix} \hat{\mathbf{e}}_1 - \mathbf{e}_1 \right\|_2 = \|\mathbf{k}_1\|_2. \quad (41)$$

It then follows that

$$\begin{aligned} \min_{\theta \in \{\pm 1\}} \|\bar{\mathbf{q}} - \theta \mathbf{u}_1\|_2 &\leq \|\bar{\mathbf{q}} - \bar{\theta} \mathbf{u}_1\|_2 = \|\mathbf{Q} \bar{\mathbf{h}} - \bar{\theta} \mathbf{u}_1\|_2 \\ &\leq \left\| \mathbf{Q} \bar{\mathbf{h}} - \bar{\theta} \frac{\mathbf{Q} \mathbf{V}^{-1} \hat{\mathbf{e}}_1}{\|\mathbf{V}^{-1} \hat{\mathbf{e}}_1\|_2} \right\|_2 + \left\| \bar{\theta} \frac{\mathbf{Q} \mathbf{V}^{-1} \hat{\mathbf{e}}_1}{\|\mathbf{V}^{-1} \hat{\mathbf{e}}_1\|_2} - \bar{\theta} \frac{\mathbf{u}_1}{\|\mathbf{V}^{-1} \hat{\mathbf{e}}_1\|_2} \right\|_2 \\ &\quad + \left\| \bar{\theta} \frac{\mathbf{u}_1}{\|\mathbf{V}^{-1} \hat{\mathbf{e}}_1\|_2} - \bar{\theta} \mathbf{u}_1 \right\|_2 \\ &\leq \left\| \bar{\mathbf{h}} - \bar{\theta} \frac{\mathbf{V}^{-1} \hat{\mathbf{e}}_1}{\|\mathbf{V}^{-1} \hat{\mathbf{e}}_1\|_2} \right\|_2 + \frac{\|\mathbf{Q} \mathbf{V}^{-1} \hat{\mathbf{e}}_1 - \mathbf{u}_1\|_2}{\|\mathbf{V}^{-1} \hat{\mathbf{e}}_1\|_2} + \left| \frac{1}{\|\mathbf{V}^{-1} \hat{\mathbf{e}}_1\|_2} - 1 \right| \\ &\leq \frac{2\sqrt{2} \|\mathbf{A}\| \cdot \|\mathbf{k}_1\|_2}{|\lambda_1 - d|} + \frac{\|\mathbf{k}_1\|_2}{\sqrt{1 + \|\mathbf{k}_1\|_2^2}} + \frac{\sqrt{1 + \|\mathbf{k}_1\|_2^2} - 1}{\sqrt{1 + \|\mathbf{k}_1\|_2^2}} \\ &\leq \left( \frac{2\sqrt{2} \|\mathbf{A}\|}{|\lambda_1 - d|} + 2 \right) \|\mathbf{k}_1\|_2, \end{aligned} \quad (42)$$

where the third inequality uses the fact that  $\mathbf{Q}$  has orthonormal columns,  $|\bar{\theta}| = 1$ , and  $\|\mathbf{u}_1\|_2 = 1$ ; the fourth one follows from (32), (40), and (41); the last one uses  $\sqrt{1 + \|\mathbf{k}_1\|_2^2} \geq 1$  and  $\sqrt{1 + \|\mathbf{k}_1\|_2^2} - 1 \leq \|\mathbf{k}_1\|_2$ .

By a similar argument, one can show that

$$\min_{\theta \in \{\pm 1\}} \|\underline{q} - \theta \mathbf{u}_2\|_2 \leq \left( \frac{2\sqrt{2}\|\mathbf{A}\|}{|\lambda_2 - \bar{d}|} + 2 \right) \|\mathbf{k}_2\|_2. \quad (43)$$

Now, suppose that (10), (18)–(20), and  $\lambda_2 > 0$  hold, which happens with probability at least  $1 - c_2 n^{-3} - 4\sqrt{\log n/n}$  for all  $n$  satisfying (33) due to Lemma 1, Corollary 2, and the union bound. Then, by (18), (19), Proposition 3, and  $n \geq \exp(16c_1^2/\beta^2)$ , we obtain

$$|\lambda_1 - \bar{d}| \geq \lambda_1 - \lambda_2 \geq \beta \log n - 2c_1 \sqrt{\log n} \geq \frac{\beta}{2} \log n. \quad (44)$$

Moreover, it follows from (18)–(20) and  $n \geq \exp(4c_1^2/(\alpha + \beta)^2)$  that

$$\|\mathbf{A}\| = \lambda_1 \leq \frac{\alpha + \beta}{2} \log n + c_1 \sqrt{\log n} \leq (\alpha + \beta) \log n. \quad (45)$$

These, together with the first inequality in (29), imply (34) as desired. Besides, using  $n > \exp(16c_1^2/(\alpha + \beta)^2)$  and (29), one can verify that for every  $k \geq \bar{K}$ ,

$$\|\mathbf{k}_1^{(k)}\|_2 \leq n \left( \frac{4c_1}{(\alpha + \beta)\sqrt{\log n}} \right)^{\bar{K}} \leq \sqrt{\frac{\beta}{8(\alpha + \beta)}} \leq \sqrt{\frac{\lambda_1 - \lambda_2}{4\|\mathbf{A}\|}},$$

where the last inequality follows from (44) and (45). It then follows from Proposition 3 that  $\lambda_1 - \bar{d}^{(k)} \leq 2\|\mathbf{A}\| \cdot \|\mathbf{k}_1\|_2^2 \leq (\lambda_1 - \lambda_2)/2$  for all  $k \geq \bar{K}$ . Together with (44), this implies that for all  $k \geq \bar{K}$ ,

$$\bar{d}^{(k)} - \lambda_2 = \bar{d}^{(k)} - \lambda_1 + \lambda_1 - \lambda_2 \geq \frac{\lambda_1 - \lambda_2}{2} \geq \frac{\beta}{4} \log n. \quad (46)$$

The desired result (35) then follows from (29), (43), (45), and (46).  $\square$

### 3.2 Analysis of the method of projected power iteration

Now, let us turn to study the convergence behavior of the projected power iterations employed in the second stage of Algorithm 1. Our goal is to show that if  $\mathbf{x}^{(0)}$  is properly chosen, then the projected power iterations will terminate in a finite number of iterations and output the ground-truth label vector  $\mathbf{x}^*$  with high probability. To begin, let  $\mathbf{A}$  be as in Theorem 1. In particular, Lemma 2 and Corollary 2 can be applied here.

Recall that the projected power iterations take the form

$$\mathbf{x}^{(k)} \in \mathcal{P}(\mathbf{A}\mathbf{x}^{(k-1)}), \quad k = 1, 2, \dots,$$

where  $\mathcal{P}$  is the projection operator onto  $\mathcal{F}$ ; see (3)–(5). The following result shows that  $\mathcal{P}$  possesses a Lipschitz-like property, despite the fact that  $\mathcal{F}$  is a discrete set. Such a property plays an important role in the analysis of the projected power iterations; cf. [25, 26].

**Lemma 3** *Suppose that  $\mathbf{c} \in \mathbb{R}^n$  is arbitrary and  $\varepsilon > 0$  is constant such that*

$$c_i \begin{cases} \geq \varepsilon, & i \in \mathcal{I}, \\ \leq -\varepsilon, & i \in [n] \setminus \mathcal{I} \end{cases} \quad (47)$$

for some  $\mathcal{I} \subset [n]$  with  $|\mathcal{I}| = n/2$ . Then, for any  $\mathbf{v} \in \mathcal{P}(\mathbf{c})$ ,  $\mathbf{c}' \in \mathbb{R}^n$ , and  $\mathbf{v}' \in \mathcal{P}(\mathbf{c}')$ , it holds that

$$\|\mathbf{v} - \mathbf{v}'\|_2 \leq \frac{2\|\mathbf{c} - \mathbf{c}'\|_2}{\varepsilon}. \quad (48)$$

**Proof** By (47) and Proposition 1, we see that  $\mathcal{P}(\mathbf{c})$  is a singleton and  $\mathbf{v} \in \mathcal{P}(\mathbf{c})$  satisfies

$$v_i = \begin{cases} 1, & i \in \mathcal{I}, \\ -1, & i \in [n] \setminus \mathcal{I}. \end{cases} \quad (49)$$

Let  $\mathbf{c}' \in \mathbb{R}^n$  be arbitrary and  $\mathbf{v}' \in \mathcal{P}(\mathbf{c}')$ . It then follows from Proposition 1 that

$$v'_i = \begin{cases} 1, & i \in \mathcal{J}, \\ -1, & i \in [n] \setminus \mathcal{J} \end{cases} \quad (50)$$

for some  $\mathcal{J} \subset [n]$  with  $|\mathcal{J}| = n/2$  such that  $c'_i \geq c'_j$  for all  $i \in \mathcal{J}$  and  $j \in [n] \setminus \mathcal{J}$ . For ease of exposition, we write  $\mathcal{I}^c := [n] \setminus \mathcal{I}$  and  $\mathcal{J}^c := [n] \setminus \mathcal{J}$ . Since

$$|\mathcal{I} \cap \mathcal{J}| + |\mathcal{I} \cap \mathcal{J}^c| = |\mathcal{I}| = \frac{n}{2}, \quad |\mathcal{I} \cap \mathcal{J}| + |\mathcal{I}^c \cap \mathcal{J}| = |\mathcal{J}| = \frac{n}{2},$$

we deduce that  $|\mathcal{I} \cap \mathcal{J}^c| = |\mathcal{I}^c \cap \mathcal{J}| = s$  for some  $0 \leq s \leq n/2$ . In addition, by (49) and (50), we have

$$v_i - v'_i = \begin{cases} 0, & i \in (\mathcal{I} \cap \mathcal{J}) \cup (\mathcal{I}^c \cap \mathcal{J}^c), \\ 2, & i \in \mathcal{I} \cap \mathcal{J}^c, \\ -2, & i \in \mathcal{I}^c \cap \mathcal{J}. \end{cases}$$

Since  $|\mathcal{I} \cap \mathcal{J}^c| = |\mathcal{I}^c \cap \mathcal{J}| = s$ , this yields

$$\|\mathbf{v} - \mathbf{v}'\|_2 = 2\sqrt{2s}. \quad (51)$$

On the other hand, we have

$$\|\mathbf{c} - \mathbf{c}'\|_2^2 = \sum_{i=1}^n (c_i - c'_i)^2 \geq \sum_{i \in \mathcal{I} \cap \mathcal{J}^c} (c_i - c'_i)^2 + \sum_{j \in \mathcal{I}^c \cap \mathcal{J}} (c_j - c'_j)^2. \quad (52)$$

It follows from (47) that  $c_i - c_j \geq 2\varepsilon$  for any  $i \in \mathcal{I}$  and  $j \in \mathcal{I}^c$ . Besides, recall from (50) that  $c'_j \geq c'_i$  for any  $i \in \mathcal{J}^c$  and  $j \in \mathcal{J}$ . Thus, for every  $i \in \mathcal{I} \cap \mathcal{J}^c$  and  $j \in \mathcal{I}^c \cap \mathcal{J}$ , it holds that

$$(c_i - c'_i)^2 + (c_j - c'_j)^2 \geq \frac{1}{2} \underbrace{(c_i - c_j)}_{\geq 2\varepsilon} \underbrace{(c'_j - c'_i)}_{\geq 0} \geq 2\varepsilon^2,$$

where the first inequality is due to  $a^2 + b^2 \geq (a - b)^2/2$  for any  $a, b \in \mathbb{R}$ . Using (52) and  $|\mathcal{I} \cap \mathcal{J}^c| = |\mathcal{I}^c \cap \mathcal{J}| = s$ , we obtain

$$\|\mathbf{c} - \mathbf{c}'\|_2^2 \geq 2s\varepsilon^2. \quad (53)$$

The desired result (48) then follows from (51) and (53).  $\square$

Next, we recall the following result, which is established in [2] and pertains to the difference of two binomial random variables; see also [3, Lemma 8].

**Lemma 4** *Let  $\alpha > \beta > 0$  be given constants. Suppose that  $\{W_i\}_{i=1}^{n/2}$  are i.i.d.  $\text{Bern}(\alpha \log n/n)$  and  $\{Z_i\}_{i=1}^{n/2}$  are i.i.d.  $\text{Bern}(\beta \log n/n)$ , with  $\{Z_i\}_{i=1}^{n/2}$  being independent of  $\{W_i\}_{i=1}^{n/2}$ . Then, for any  $\gamma \in \mathbb{R}$ , it holds that*

$$\mathbb{P} \left( \sum_{i=1}^{n/2} W_i - \sum_{i=1}^{n/2} Z_i \leq \gamma \log n \right) \leq n^{-\frac{(\sqrt{\alpha} - \sqrt{\beta})^2}{2} + \frac{\gamma(\log \alpha - \log \beta)}{2}}. \quad (54)$$

Equipped with Lemmas 3 and 4, we can show that the set-valued map  $\mathbf{x} \mapsto \mathcal{P}(\mathbf{A}\mathbf{x})$  possesses a contraction property in a certain neighborhood of  $\mathbf{x}^*$ . This would then imply the linear convergence of the projected power iterations.

**Proposition 4** *Suppose that the constants  $\alpha > \beta > 0$  satisfy  $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{2}$ . Then, there exists a constant  $\gamma > 0$ , whose value depends only on  $\alpha$  and  $\beta$ , such that the following statement holds with probability at least  $1 - n^{-\Omega(1)}$ : For all  $\mathbf{x} \in \mathbb{R}^n$  such that  $\mathbf{1}^T \mathbf{x} = 0$ ,  $\|\mathbf{x}\|_2 = \sqrt{n}$ , and*

$$\|\mathbf{x} - \mathbf{x}^*\|_2 \leq 5c_3 \sqrt{\frac{n}{\log n}}, \quad (55)$$

*one has*

$$\|\mathbf{v} - \mathbf{x}^*\|_2 \leq \frac{c_4}{\gamma \sqrt{\log n}} \|\mathbf{x} - \mathbf{x}^*\|_2 \quad (56)$$

*for any  $\mathbf{v} \in \mathcal{P}(\mathbf{A}\mathbf{x})$ , where  $c_4 := 3c_3(\alpha - \beta) + 2c_1$  and  $c_1, c_3$  are the constants in Lemma 2 and Corollary 2, respectively.*

**Proof** Since  $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{2}$ , there exists a  $\gamma > 0$ , whose value depends only on  $\alpha$  and  $\beta$ , such that

$$c_5 := \frac{(\sqrt{\alpha} - \sqrt{\beta})^2}{2} - \frac{\gamma(\log \alpha - \log \beta)}{2} - 1 > 0. \quad (57)$$

According to the binary symmetric SBM in Definition 1, we have

$$x_i^*(\mathbf{A}\mathbf{x}^*)_i = \sum_{j=1}^n a_{ij}x_i^*x_j^* \stackrel{d}{=} \sum_{i=1}^{n/2} W_i - \sum_{i=1}^{n/2} Z_i$$

for every  $i \in [n]$ , where  $\{W_i\}_{i=1}^{n/2}$  are i.i.d.  $\text{Bern}(\alpha \log n/n)$ ,  $\{Z_i\}_{i=1}^{n/2}$  are i.i.d.  $\text{Bern}(\beta \log n/n)$ , and  $\{Z_i\}_{i=1}^{n/2}$  are independent of  $\{W_i\}_{i=1}^{n/2}$ . It then follows from Lemma 4, (57), and the union bound that

$$\min_{i \in [n]} x_i^*(\mathbf{A}\mathbf{x}^*)_i \geq \gamma \log n \quad (58)$$

holds with probability at least  $1 - n^{-c_5}$ .

In the rest of the proof, we suppose that both (17) and (58) hold, which happens with probability at least  $1 - c_2 n^{-3} - n^{-c_5}$  due to Lemma 2 and the union bound. Let  $\mathcal{I} = \{i \in [n] : x_i^* = 1\}$  and  $\mathcal{I}^c = [n] \setminus \mathcal{I}$ . Since  $\mathbf{x}^* \in \mathcal{F}$ , we have  $|\mathcal{I}| = |\mathcal{I}^c| = n/2$  and  $x_i^* = -1$  for all  $i \in \mathcal{I}^c$ . This, together with (58), implies that

$$(\mathbf{A}\mathbf{x}^*)_i \begin{cases} \geq \gamma \log n, & i \in \mathcal{I}, \\ \leq -\gamma \log n, & i \in \mathcal{I}^c. \end{cases} \quad (59)$$

It then follows from Proposition 1 that  $\mathcal{P}(\mathbf{A}\mathbf{x}^*) = \{\mathbf{x}^*\}$ . Now, let  $\mathbf{x} \in \mathbb{R}^n$  be such that  $\mathbf{1}^T \mathbf{x} = 0$ ,  $\|\mathbf{x}\|_2 = \sqrt{n}$ , and (55) holds. By (59),  $\mathcal{P}(\mathbf{A}\mathbf{x}^*) = \{\mathbf{x}^*\}$ , and Lemma 3, we obtain

$$\|\mathbf{v} - \mathbf{x}^*\|_2 \leq \frac{2\|\mathbf{A}\mathbf{x} - \mathbf{A}\mathbf{x}^*\|_2}{\gamma \log n} \quad (60)$$

for any  $\mathbf{v} \in \mathcal{P}(\mathbf{A}\mathbf{x})$ . In addition, using  $\|\mathbf{x}^*\|_2 = \|\mathbf{x}\|_2$ , we compute

$$\|\mathbf{x} - \mathbf{x}^*\|_2^2 = \|\mathbf{x}\|_2^2 + \|\mathbf{x}^*\|_2^2 - 2\mathbf{x}^T \mathbf{x}^* = 2\|\mathbf{x}^*\|_2^2 - 2\mathbf{x}^T \mathbf{x}^* = 2(\mathbf{x}^{*T} \mathbf{x}^* - \mathbf{x}^T \mathbf{x}^*).$$

This, together with (17), (24),  $\mathbf{1}^T \mathbf{x} = \mathbf{1}^T \mathbf{x}^* = 0$ , and  $\|\mathbf{x}^*\|_2 = \sqrt{n}$ , yields

$$\begin{aligned} \|\mathbf{A}\mathbf{x} - \mathbf{A}\mathbf{x}^*\|_2 &= \|\mathbb{E}[\mathbf{A}](\mathbf{x} - \mathbf{x}^*) + (\mathbf{A} - \mathbb{E}[\mathbf{A}])(\mathbf{x} - \mathbf{x}^*)\|_2 \\ &\leq \left\| \left( \frac{p+q}{2} \mathbf{1}\mathbf{1}^T + \frac{p-q}{2} \mathbf{x}^* \mathbf{x}^{*T} \right) (\mathbf{x} - \mathbf{x}^*) \right\|_2 \\ &\quad + \|\mathbf{A} - \mathbb{E}[\mathbf{A}]\| \cdot \|\mathbf{x} - \mathbf{x}^*\|_2 \end{aligned}$$

$$\begin{aligned}
&= \left\| \frac{p-q}{2} (\mathbf{x}^T \mathbf{x}^* - \mathbf{x}^{*T} \mathbf{x}^*) \mathbf{x}^* \right\|_2 + \|\mathbf{A} - \mathbb{E}[\mathbf{A}]\| \cdot \|\mathbf{x} - \mathbf{x}^*\|_2 \\
&\leq \frac{p-q}{4} \sqrt{n} \|\mathbf{x} - \mathbf{x}^*\|_2^2 + c_1 \sqrt{\log n} \|\mathbf{x} - \mathbf{x}^*\|_2.
\end{aligned} \tag{61}$$

Then, by (2), (55), (60), and (61), we obtain for any  $\mathbf{v} \in \mathcal{P}(\mathbf{A}\mathbf{x})$  that

$$\begin{aligned}
\|\mathbf{v} - \mathbf{x}^*\|_2 &\leq \frac{2\|\mathbf{A}\mathbf{x} - \mathbf{A}\mathbf{x}^*\|_2}{\gamma \log n} \\
&\leq \frac{2}{\gamma \log n} \left( \frac{(\alpha - \beta) \log n}{4\sqrt{n}} \|\mathbf{x} - \mathbf{x}^*\|_2 + c_1 \sqrt{\log n} \right) \|\mathbf{x} - \mathbf{x}^*\|_2 \\
&\leq \frac{3c_3(\alpha - \beta) + 2c_1}{\gamma \sqrt{\log n}} \|\mathbf{x} - \mathbf{x}^*\|_2 \\
&= \frac{c_4}{\gamma \sqrt{\log n}} \|\mathbf{x} - \mathbf{x}^*\|_2.
\end{aligned}$$

This completes the proof.  $\square$

Since the feasible set  $\mathcal{F}$  of Problem (MLE) is discrete, the contraction property (56) suggests that if an iterate is sufficiently close to  $\mathbf{x}^*$ , then the projected power iterations will exhibit one-step convergence to  $\mathbf{x}^*$ ; i.e., all subsequent iterates will stay at  $\mathbf{x}^*$ . This would then imply the finite termination of stage 2 of Algorithm 1. Let us now formalize the above observation.

**Proposition 5** Suppose that the constants  $\alpha > \beta > 0$  satisfy  $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{2}$  and let  $\gamma > 0$  be such that (57) holds. Then, the following statement holds with probability at least  $1 - n^{-\Omega(1)}$ : For all  $\mathbf{x} \in \mathcal{F}$  such that  $\|\mathbf{x} - \mathbf{x}^*\|_2 \leq \sqrt{2\gamma \log n}$ , one has

$$\mathcal{P}(\mathbf{A}\mathbf{x}) = \{\mathbf{x}^*\}. \tag{62}$$

**Proof** Suppose that (58) holds, which happens with probability at least  $1 - n^{-c_5}$ , where  $c_5 > 0$  is given in (57). Let  $\mathcal{I} = \{i \in [n] : x_i^* = 1\}$  and  $\mathcal{I}^c = [n] \setminus \mathcal{I}$ . Since  $\mathbf{x}^* \in \mathcal{F}$ , we have  $|\mathcal{I}| = |\mathcal{I}^c| = n/2$  and  $x_i^* = -1$  for all  $i \in \mathcal{I}^c$ . Let  $\mathbf{x} \in \mathcal{F}$  be such that  $\|\mathbf{x} - \mathbf{x}^*\|_2 \leq \sqrt{2\gamma \log n}$ . Since  $\mathbf{x}, \mathbf{x}^* \in \mathcal{F}$ , there exist an  $S \subset \mathcal{I}$  and an  $S' \subset \mathcal{I}^c$  with  $|S| = |S'| \leq n/2$  such that

$$\mathbf{x} = \mathbf{x}^* - 2\mathbf{e}_S + 2\mathbf{e}_{S'},$$

where  $\mathbf{e}_S$  (resp.  $\mathbf{e}_{S'}$ ) is an  $n$ -dimensional vector with  $(\mathbf{e}_S)_i = 1$  if  $i \in S$  (resp.  $S'$ ) and 0 otherwise. Observe that

$$\sqrt{2\gamma \log n} \geq \|\mathbf{x} - \mathbf{x}^*\|_2 = 2\|\mathbf{e}_{S'} - \mathbf{e}_S\|_2 = 2\sqrt{|S| + |S'|}.$$

This, together with  $|S| = |S'|$ , implies that  $|S| = |S'| \leq \gamma \log n/4$ . Now, for all  $i \in \mathcal{I}$ , we get

$$(\mathbf{A}\mathbf{x})_i = (\mathbf{A}\mathbf{x}^*)_i - 2(\mathbf{A}\mathbf{e}_S)_i + 2(\mathbf{A}\mathbf{e}_{S'})_i$$

$$\begin{aligned}
&\geq \gamma \log n - 2 \sum_{j \in S} a_{ij} + 2 \sum_{j \in S'} a_{ij} \\
&\geq \gamma \log n - 2|S| \\
&\geq \frac{1}{2} \gamma \log n,
\end{aligned}$$

where the first inequality follows from (58), the second inequality uses  $0 \leq a_{ij} \leq 1$ , and the last one is due to  $|S| \leq \gamma \log n / 4$ . By the same argument, we can show that for all  $i \in \mathcal{I}^c$ ,

$$(\mathbf{A}\mathbf{x})_i \leq -\frac{1}{2} \gamma \log n.$$

These, together with Proposition 1, imply (62) as desired.  $\square$

The next result establishes the iteration complexity of the projected power iterations for finding the ground-truth label  $\mathbf{x}^*$ . Recall that  $c_1$ ,  $c_3$ , and  $c_4$  are the constants defined in Lemma 2, Corollary 2, and Proposition 4, respectively, whose values depend only on  $\alpha$  and  $\beta$ .

**Theorem 3** *Suppose that the constants  $\alpha > \beta > 0$  satisfy  $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{2}$  and let  $\gamma > 0$  be such that (57) holds. Suppose in addition that*

$$n > \max \left\{ \exp \left( \frac{c_4^2}{\gamma^2} \right), \exp \left( \frac{5c_3}{\sqrt{2}\gamma} \right) \right\}. \quad (63)$$

*Let  $\{\mathbf{x}^{(k)}\}_{k \geq 0}$  be the sequence generated by Algorithm 1. If  $\mathbf{x}^{(0)}$  satisfies*

$$\mathbf{1}^T \mathbf{x}^{(0)} = 0, \quad \|\mathbf{x}^{(0)}\|_2 = \sqrt{n}, \quad \|\mathbf{x}^{(0)} - \mathbf{x}^*\|_2 \leq 5c_3 \sqrt{\frac{n}{\log n}}, \quad (64)$$

*then the following statements hold with probability at least  $1 - n^{-\Omega(1)}$ :*

(i) *For all  $k \geq 1$ , it holds that*

$$\|\mathbf{x}^{(k)} - \mathbf{x}^*\|_2 \leq \frac{c_4}{\gamma \sqrt{\log n}} \|\mathbf{x}^{(k-1)} - \mathbf{x}^*\|_2 \quad (65)$$

*and*

$$\|\mathbf{x}^{(k)} - \mathbf{x}^*\|_2 \leq 5c_3 \sqrt{\frac{n}{\log n}}. \quad (66)$$

(ii) *There exists some  $k \leq \tilde{K}$  such that  $\mathbf{x}^{(k)} = \mathbf{x}^{(k-1)} = \mathbf{x}^*$ , where*

$$\tilde{K} = \left\lceil \frac{\log n}{\log \log n + 2 \log \left( \frac{\gamma}{c_4} \right)} \right\rceil + 2. \quad (67)$$

**Proof** Suppose that the statements in Propositions 4 and 5 hold simultaneously, which happens with probability at least  $1 - n^{-\Omega(1)}$  by the union bound. We first prove (i). It follows from (64) that (66) holds for  $k = 0$ . Moreover, by (64), Proposition 4, and  $\mathbf{x}^{(1)} \in \mathcal{P}(\mathbf{A}\mathbf{x}^{(0)})$ , one can observe that (65) holds for  $k = 1$ . These, together with (63), yield

$$\|\mathbf{x}^{(1)} - \mathbf{x}^*\|_2 \leq \frac{c_4}{\gamma\sqrt{\log n}} \|\mathbf{x}^{(0)} - \mathbf{x}^*\|_2 \leq 5c_3 \sqrt{\frac{n}{\log n}}$$

and thus (66) holds for  $k = 1$ . Then, (i) can be established by a simple inductive argument. Next, we prove (ii). By (63), we have  $c_4/(\gamma\sqrt{\log n}) < 1$  and  $5c_3/\sqrt{\log n} < \sqrt{2\gamma \log n}$ . This, together with (64), (65), and (67), yields

$$\begin{aligned} \|\mathbf{x}^{(\tilde{K}-2)} - \mathbf{x}^*\|_2 &\leq \|\mathbf{x}^{(0)} - \mathbf{x}^*\|_2 \left( \frac{c_4}{\gamma\sqrt{\log n}} \right)^{\tilde{K}-2} \\ &\leq 5c_3 \sqrt{\frac{n}{\log n}} \left( \frac{c_4}{\gamma\sqrt{\log n}} \right)^{\frac{\log n}{\log \log n + 2 \log(\gamma/c_4)}} = \frac{5c_3}{\sqrt{\log n}} \leq \sqrt{2\gamma \log n}. \end{aligned}$$

By Proposition 5 and the projected power iterations (5), we have  $\mathbf{x}^{(\tilde{K}-1)} = \mathbf{x}^*$ . By applying Proposition 5 to  $\mathbf{x}^{(\tilde{K}-1)}$ , we further have  $\mathbf{x}^{(\tilde{K})} = \mathbf{x}^{(\tilde{K}-1)} = \mathbf{x}^*$ . This, together with the stopping criterion in Algorithm 1, completes the proof of (ii).  $\square$

Let us make two remarks before we leave this sub-section. First, due to the symmetry of Problem (MLE), Theorem 3 also holds if we replace all the  $\mathbf{x}^*$  in its statement by  $-\mathbf{x}^*$ . Second, Theorem 3(ii) implies that the second stage of Algorithm 1 terminates in a finite number of iterations at an optimal solution of Problem (MLE), provided that  $\mathbf{x}^{(0)}$  is properly chosen. In particular, for any  $\mathbf{x}^{(0)}$  satisfying (64), it terminates in roughly  $\mathcal{O}(\log n / \log \log n)$  iterations for all sufficiently large  $n$ .

### 3.3 Proofs of Theorem 1 and Corollary 1

With the preparations in Sects. 3.1 and 3.2, we are ready to establish the main results stated in Sect. 2. We first provide a formal version of Theorem 1 and its proof. Recall that the constants  $c_1, c_3, c_4$  are given in Lemma 2, Corollary 2, and Proposition 4, respectively, and their values depend only on  $\alpha$  and  $\beta$ .

**Theorem 4** Consider the setting of Theorem 1. Suppose that  $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{2}$  and let  $\gamma > 0$  be such that (57) holds. In addition, suppose that

$$n > \max \left\{ \exp \left( \frac{16c_1^2}{\min\{(\alpha - \beta)^2, \beta^2\}} \right), \exp \left( \frac{5c_3}{\sqrt{2\gamma}} \right), \exp \left( \frac{c_4^2}{\gamma^2} \right), 6 \right\}. \quad (68)$$



Then, with probability at least  $1 - n^{-\Omega(1)}$ , Algorithm 1 takes at most  $N_1$  orthogonal iterations and  $N_2$  projected power iterations to find  $\mathbf{x}^*$  or  $-\mathbf{x}^*$ , where

$$N_1 = \left\lceil \frac{2 \log n + \log \log n + 2 \log \left( \frac{16\alpha + 18\beta}{\beta \cdot \min\{1, c_3\}} \right)}{\log \log n + 2 \log \left( \frac{\alpha - \beta}{4c_1} \right)} \right\rceil, \quad (69)$$

$$N_2 = \left\lceil \frac{\log n}{\log \log n + 2 \log \left( \frac{\gamma}{c_4} \right)} \right\rceil + 2. \quad (70)$$

**Proof** Suppose that the statements in Lemma 2, Theorem 2, and Theorem 3 hold, which happens with probability at least  $1 - n^{-\Omega(1)}$  due to the union bound. Recall from line 9 of Algorithm 1 that  $\bar{\mathbf{Q}}^{(N_1)} = \mathbf{Q}^{(N_1)} \mathbf{H}^{(N_1)}$ . Moreover, the columns of  $\bar{\mathbf{Q}}^{(N_1)}$  are denoted by  $\bar{\mathbf{q}}^{(N_1)} = \mathbf{Q}^{(N_1)} \bar{\mathbf{h}}^{(N_1)}$  and  $\underline{\mathbf{q}}^{(N_1)} = \mathbf{Q}^{(N_1)} \underline{\mathbf{h}}^{(N_1)}$ , where  $\bar{\mathbf{h}}^{(N_1)}, \underline{\mathbf{h}}^{(N_1)}$  are the eigenvectors of  $\mathbf{Q}^{(N_1)T} \mathbf{A} \mathbf{Q}^{(N_1)}$  associated with the eigenvalues  $\bar{d}^{(N_1)}, \underline{d}^{(N_1)}$ , respectively, and  $\bar{d}^{(N_1)} \geq \underline{d}^{(N_1)}$ .

By  $\alpha > \beta > 0$ , (36), and (69), one can verify that  $N_1 \geq \bar{K}$ . It then follows from Theorem 2 and (69) that there exist  $\theta_1, \theta_2 \in \{\pm 1\}$  satisfying

$$\|\bar{\mathbf{q}}^{(N_1)} - \theta_1 \mathbf{u}_1\|_2 \leq n \left( \frac{8\alpha}{\beta} + 10 \right) \left( \frac{4c_1}{(\alpha + \beta)\sqrt{\log n}} \right)^{N_1} \leq \frac{c_3}{\sqrt{\log n}}, \quad (71)$$

$$\|\underline{\mathbf{q}}^{(N_1)} - \theta_2 \mathbf{u}_2\|_2 \leq n \left( \frac{16\alpha}{\beta} + 18 \right) \left( \frac{4c_1}{(\alpha - \beta)\sqrt{\log n}} \right)^{N_1} \leq \frac{c_3}{\sqrt{\log n}}. \quad (72)$$

After obtaining  $\bar{\mathbf{Q}}^{(N_1)}$ , Algorithm 1 generates an initial point  $\mathbf{x}^{(0)}$  for the projected power iterations by setting  $\mathbf{x}^{(0)} = \sqrt{n} \mathbf{y}^{(0)} / \|\mathbf{y}^{(0)}\|_2$ , where  $\mathbf{y}^{(0)} = \underline{\mathbf{q}}^{(N_1)} - (\mathbf{1}^T \underline{\mathbf{q}}^{(N_1)} / n) \mathbf{1}$ . Since  $\|\mathbf{1}\|_2 = \sqrt{n}$  and  $\|\underline{\mathbf{q}}^{(N_1)}\|_2 = 1$ , we see that  $\|\mathbf{y}^{(0)}\|_2 \leq 1$  and

$$\mathbf{y}^{(0)T} \underline{\mathbf{q}}^{(N_1)} = 1 - \frac{(\mathbf{1}^T \underline{\mathbf{q}}^{(N_1)})^2}{n} \geq 0.$$

These, together with  $\|\mathbf{x}^{(0)}\|_2 = \sqrt{n}$ , yield

$$\begin{aligned} \|\mathbf{x}^{(0)} - \sqrt{n} \underline{\mathbf{q}}^{(N_1)}\|_2^2 &= 2n - 2n \frac{(\mathbf{y}^{(0)})^T \underline{\mathbf{q}}^{(N_1)}}{\|\mathbf{y}^{(0)}\|_2} \leq 2n - 2n (\mathbf{y}^{(0)})^T \underline{\mathbf{q}}^{(N_1)} \\ &= 2n - 2n \left( 1 - \frac{(\mathbf{1}^T \underline{\mathbf{q}}^{(N_1)})^2}{n} \right) = 2(\mathbf{1}^T \underline{\mathbf{q}}^{(N_1)})^2, \end{aligned}$$

which implies that

$$\|\mathbf{x}^{(0)} - \sqrt{n} \underline{\mathbf{q}}^{(N_1)}\|_2 \leq \sqrt{2} |\mathbf{1}^T \underline{\mathbf{q}}^{(N_1)}|. \quad (73)$$

Besides, by (21), there exist  $\tilde{\theta}_1, \tilde{\theta}_2 \in \{\pm 1\}$  such that

$$\left\| \tilde{\theta}_1 \mathbf{u}_1 - \frac{\mathbf{1}}{\sqrt{n}} \right\|_2 \leq \frac{c_3}{\sqrt{\log n}}, \quad \left\| \tilde{\theta}_2 \mathbf{u}_2 - \frac{\mathbf{x}^*}{\sqrt{n}} \right\|_2 \leq \frac{c_3}{\sqrt{\log n}}. \quad (74)$$

This, together with (71), (73),  $(\bar{\mathbf{q}}^{(N_1)})^T \underline{\mathbf{q}}^{(N_1)} = 0$ , and  $\|\underline{\mathbf{q}}^{(N_1)}\|_2 = 1$ , yields

$$\begin{aligned} \|\mathbf{x}^{(0)} - \sqrt{n} \underline{\mathbf{q}}^{(N_1)}\|_2 &\leq \sqrt{2n} \left\| \left( \frac{\mathbf{1}}{\sqrt{n}} - \tilde{\theta}_1 \mathbf{u}_1 + \tilde{\theta}_1 \mathbf{u}_1 - \frac{\tilde{\theta}_1 \bar{\mathbf{q}}^{(N_1)}}{\theta_1} \right)^T \underline{\mathbf{q}}^{(N_1)} \right\| \\ &\leq \sqrt{2n} \left( \left\| \tilde{\theta}_1 \mathbf{u}_1 - \frac{\mathbf{1}}{\sqrt{n}} \right\|_2 + \left\| \theta_1 \mathbf{u}_1 - \bar{\mathbf{q}}^{(N_1)} \right\|_2 \right) \\ &\leq 2\sqrt{2}c_3 \sqrt{\frac{n}{\log n}}. \end{aligned} \quad (75)$$

Then, by (72), (74), (75), and  $\theta_2, \tilde{\theta}_2 \in \{\pm 1\}$ , we obtain

$$\begin{aligned} \min_{\theta \in \{\pm 1\}} \|\mathbf{x}^{(0)} - \theta \mathbf{x}^*\|_2 &\leq \left\| \mathbf{x}^{(0)} - \frac{\theta_2}{\tilde{\theta}_2} \mathbf{x}^* \right\|_2 \\ &\leq \|\mathbf{x}^{(0)} - \sqrt{n} \underline{\mathbf{q}}^{(N_1)}\|_2 + \sqrt{n} \|\underline{\mathbf{q}}^{(N_1)} - \theta_2 \mathbf{u}_2\|_2 + \sqrt{n} \left\| \theta_2 \mathbf{u}_2 - \frac{\theta_2}{\tilde{\theta}_2} \frac{\mathbf{x}^*}{\sqrt{n}} \right\|_2 \\ &= \|\mathbf{x}^{(0)} - \sqrt{n} \underline{\mathbf{q}}^{(N_1)}\|_2 + \sqrt{n} \|\underline{\mathbf{q}}^{(N_1)} - \theta_2 \mathbf{u}_2\|_2 + \sqrt{n} \left\| \tilde{\theta}_2 \mathbf{u}_2 - \frac{\mathbf{x}^*}{\sqrt{n}} \right\|_2 \\ &\leq (2 + 2\sqrt{2})c_3 \sqrt{\frac{n}{\log n}} \leq 5c_3 \sqrt{\frac{n}{\log n}}. \end{aligned}$$

This, together with  $\|\mathbf{x}^{(0)}\|_2 = \sqrt{n}$ ,  $\mathbf{1}^T \mathbf{x}^{(0)} = 0$ , and Theorem 3, implies that if  $N_2 = \tilde{K}$ , where  $\tilde{K}$  is defined in (67), then Algorithm 1 can find  $\mathbf{x}^*$  or  $-\mathbf{x}^*$  with probability at least  $1 - n^{-\Omega(1)}$ .  $\square$

Armed with the results in Theorem 4, we can now provide a proof of Corollary 1.

**Proof (of Corollary 1)** From the discussion following Theorem 1, it remains to bound the number of non-zero entries in  $\mathbf{A}$ . Towards that end, let us first estimate the number of non-zero entries in an arbitrary column  $\mathbf{a} \in \mathbb{R}^n$  of  $\mathbf{A}$ , which we denote by  $\|\mathbf{a}\|_0$ . According to the binary symmetric SBM in Definition 1, we have

$$\|\mathbf{a}\|_0 \stackrel{d}{=} \sum_{i=1}^{n/2} W_i + \sum_{i=1}^{n/2} Z_i,$$

where  $\{W_i\}_{i=1}^{n/2}$  are i.i.d.  $\mathbf{Bern}(p)$ ,  $\{Z_i\}_{i=1}^{n/2}$  are i.i.d.  $\mathbf{Bern}(q)$ , and  $\{Z_i\}_{i=1}^{n/2}$  are independent of  $\{W_i\}_{i=1}^{n/2}$ . It follows that

$$\mathbb{E}[\|\mathbf{a}\|_0] = \frac{n}{2}(p+q), \quad \text{Var}(\|\mathbf{a}\|_0) \leq \frac{n}{2}(p+q).$$

Upon applying Bernstein's inequality for bounded distributions (see, e.g., [36, Theorem 2.8.4]), we get

$$\begin{aligned} \mathbb{P}\left(\left|\|\mathbf{a}\|_0 - \frac{n}{2}(p+q)\right| \geq \frac{3}{2}n(p+q)\right) &\leq 2 \exp\left(-\frac{\frac{9}{8}n^2(p+q)^2}{\frac{1}{2}n(p+q) + \frac{1}{2}n(p+q)}\right) \\ &= 2 \exp\left(-\frac{9}{8}n(p+q)\right) = 2 \exp\left(-\frac{9}{8}(\alpha+\beta)\log n\right) = 2n^{-\frac{9}{8}(\alpha+\beta)}. \end{aligned}$$

This gives

$$\mathbb{P}(\|\mathbf{a}\|_0 < 2n(p+q)) \geq 1 - 2n^{-\frac{9}{8}(\alpha+\beta)} \geq 1 - 2n^{-\frac{9}{4}},$$

where the second inequality is due to  $\alpha + \beta > 2$ , which follows from  $\alpha > \beta > 0$  and  $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{2}$ .

Now, upon applying the union bound, we conclude that with probability at least  $1 - 2n^{-5/4}$ , the number of non-zero entries in  $\mathbf{A}$  is less than  $2n^2(p+q) = 2(\alpha+\beta)n \log n$ . Thus, the per-iteration cost of Algorithm 1 is  $\mathcal{O}(n \log n)$ . Together with Theorem 1, the desired bound on the total computational cost of Algorithm 1 follows.  $\square$

### 3.4 Discussion

Before we leave this section, let us discuss two further consequences of our technical development.

#### 3.4.1 Binary symmetric SBM without self-loops

Consider a variant of the binary symmetric SBM in which no self-loop is allowed in the graph. Specifically, the entries of the adjacency matrix  $\mathbf{A}$  are still generated independently according to (1), except that  $a_{ii} = 0$  for all  $i \in [n]$ . Our results are still valid under this setting. Indeed, it is easy to verify that

$$\mathbb{E}[\mathbf{A}] = \frac{p+q}{2} \mathbf{1}\mathbf{1}^T + \frac{p-q}{2} \mathbf{x}^* \mathbf{x}^{*T} - p\mathbf{I}.$$

In particular, the eigenvalues of  $\mathbb{E}[\mathbf{A}]$  change by  $p = \mathcal{O}(\log n/n)$ , which is dominated by other quantities that appear Lemma 2 and Corollary 2. Thus, the results in Sect. 3.1

still hold. Moreover, observe that

$$x_i^*(\mathbf{A}\mathbf{x}^*)_i = \sum_{j \neq i} a_{ij} x_i^* x_j^* \stackrel{d}{=} \sum_{i=1}^{n/2-1} W_i - \sum_{i=1}^{n/2} Z_i$$

for every  $i \in [n]$ , where, as before,  $\{W_i\}_{i=1}^{n/2-1}$  are i.i.d.  $\text{Bern}(\alpha \log n/n)$ ,  $\{Z_i\}_{i=1}^{n/2}$  are i.i.d.  $\text{Bern}(\beta \log n/n)$ , and  $\{Z_i\}_{i=1}^{n/2}$  are independent of  $\{W_i\}_{i=1}^{n/2-1}$ . Since

$$\mathbb{P} \left( \sum_{i=1}^{n/2-1} W_i - \sum_{i=1}^{n/2} Z_i \leq \gamma \log n \right) \leq \mathbb{P} \left( \sum_{i=1}^{n/2-1} W_i - \sum_{i=1}^{n/2-1} Z_i \leq \gamma \log n + 1 \right)$$

for any  $\gamma \in \mathbb{R}$ , we can apply Lemma 4 and obtain the results in Sect. 3.2. Hence, we can conclude that Theorem 4 holds. Lastly, the number of non-zero entries in  $\mathbf{A}$  can only decrease if there is no self-loop, which implies that Corollary 1 also holds.

### 3.4.2 Implementation of the vanilla spectral method and its complexity analysis

Recently, Abbe et al. [3] have shown that the vanilla spectral method, which is presented in Algorithm 2, can also achieve exact recovery down to the information-theoretic limit in the binary symmetric SBM. A popular and practically efficient way of implementing this method is to employ orthogonal iterations to find the eigenvector  $\mathbf{u}_2$  of the adjacency matrix  $\mathbf{A}$ . However, the results in [3] do not establish the number of orthogonal iterations needed to obtain a sufficiently accurate approximation of  $\mathbf{u}_2$  that can provably recover the communities in the graph.

---

#### Algorithm 2: Vanilla Spectral Method

---

**Input:** adjacency matrix  $\mathbf{A}$

**Output:** label vector  $\hat{\mathbf{x}}$

- 1 compute  $\mathbf{u}_2$ , the eigenvector of  $\mathbf{A}$  associated with the second largest eigenvalue of  $\mathbf{A}$
  - 2 output  $\hat{\mathbf{x}} = \text{sgn}(\mathbf{u}_2)$
- 

By combining the results in [3] with those in Sects. 3.1 and 3.2, we now show that if the vanilla spectral method is implemented as  $\mathcal{O}(\log n / \log \log n)$  orthogonal iterations followed by a single projection onto  $\mathcal{F}$ , then it can exactly recover the underlying communities with high probability. In particular, such an implementation runs in  $\mathcal{O}(n \log^2 n / \log \log n)$  time, which, to the best of our knowledge, is currently the best complexity bound for the vanilla spectral method in the context of exact community recovery in the binary symmetric SBM. To begin, we recall [3, Corollary 3.1], which states that with probability  $1 - \mathcal{O}(n^{-3})$ , one has

$$\min_{\theta \in \{\pm 1\}} \left\| \mathbf{u}_2 - \theta \frac{\mathbf{A}\mathbf{x}^*}{v_2 \sqrt{n}} \right\|_{\infty} \leq \frac{C}{\sqrt{n} \log \log n}, \quad (76)$$

where  $C > 0$  is a constant whose value depends only on  $\alpha$  and  $\beta$ , and  $v_2$  is the second largest eigenvalue of  $\mathbb{E}[\mathbf{A}]$  (see (25)). By Theorem 2, after

$$N = \left\lceil \frac{3 \log n + 2 \log \log \log n + 2 \log \left( \frac{16\alpha + 18\beta}{C\beta} \right)}{\log \log n + 2 \log \left( \frac{\alpha - \beta}{4c_1} \right)} \right\rceil \geq \bar{K}$$

orthogonal iterations, where  $\bar{K}$  is given in (36), we obtain a vector  $\underline{\mathbf{q}}^{(N)}$  satisfying

$$\min_{\theta \in \{\pm 1\}} \|\underline{\mathbf{q}}^{(N)} - \theta \mathbf{u}_2\|_2 \leq \frac{C}{\sqrt{n} \log \log n}. \quad (77)$$

From (76) and (77), we deduce the existence of  $\tilde{\theta} \in \{\pm 1\}$  such that

$$\left\| \underline{\mathbf{q}}^{(N)} - \tilde{\theta} \frac{\mathbf{A}\mathbf{x}^*}{v_2 \sqrt{n}} \right\|_\infty \leq \frac{2C}{\sqrt{n} \log \log n}. \quad (78)$$

Now, by (58), it holds with probability at least  $1 - n^{-c_5}$  that

$$\min_{i \in [n]} \left| \frac{(\mathbf{A}\mathbf{x}^*)_i}{v_2 \sqrt{n}} \right| \geq \frac{2\gamma}{\sqrt{n}(\alpha - \beta)}. \quad (79)$$

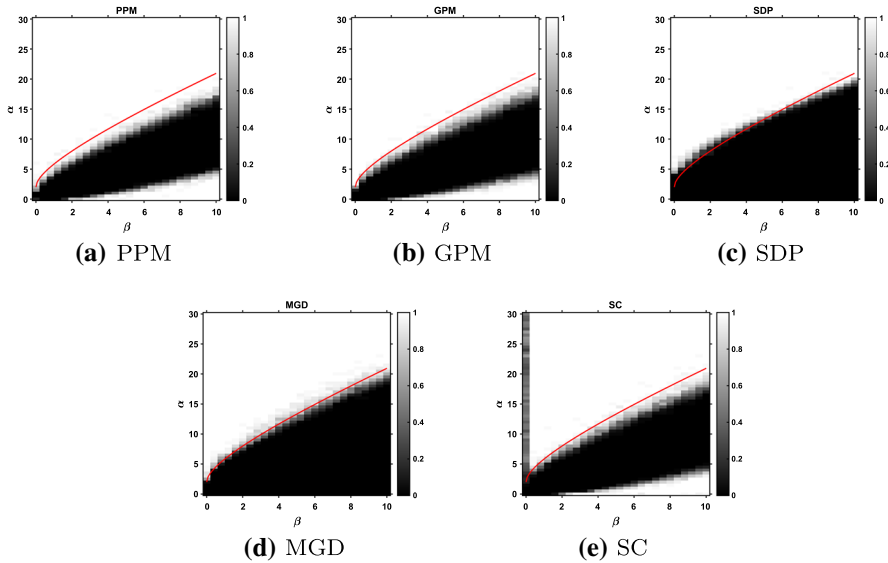
It follows that

$$\mathbf{x}^* = \text{sgn}(\mathbf{x}^*) = \text{sgn}(\mathbf{A}\mathbf{x}^*) = \text{sgn} \left( \frac{\mathbf{A}\mathbf{x}^*}{v_2 \sqrt{n}} \right) = \tilde{\theta} \cdot \text{sgn}(\underline{\mathbf{q}}^{(N)}),$$

where the second equality is due to (58) and the fourth equality uses (78) and (79). The above implies that  $\underline{\mathbf{q}}^{(N)}$  has  $n/2$  positive entries and  $n/2$  negative entries. Thus, by Proposition 1, we have  $\mathcal{P}(\underline{\mathbf{q}}^{(N)}) \in \{\pm \mathbf{x}^*\}$  as desired.

## 4 Numerical results

In this section, we report the recovery performance and numerical efficiency of our proposed two-stage method (which we denote by PPM in this section for ease of reference) for community recovery on both synthetic and real data sets. We also compare our approach with four existing approaches, which are the SDP-based approach in [6], the manifold optimization (MFO)-based approach in [8], the spectral clustering (SC) approach in [3], and the two-stage approach based on the generalized power method (GPM) in [37]. In the implementation, we use alternating direction method of multipliers (ADMM) to solve the SDP as suggested in [6], manifold gradient descent (MGD) method to solve the MFO, and the MATLAB function `eigs` for computing the eigenvector that is needed in the SC approach. Our codes are implemented in MATLAB R2020a and can be downloaded at <https://github.com/peng8wang/MP-Exact->

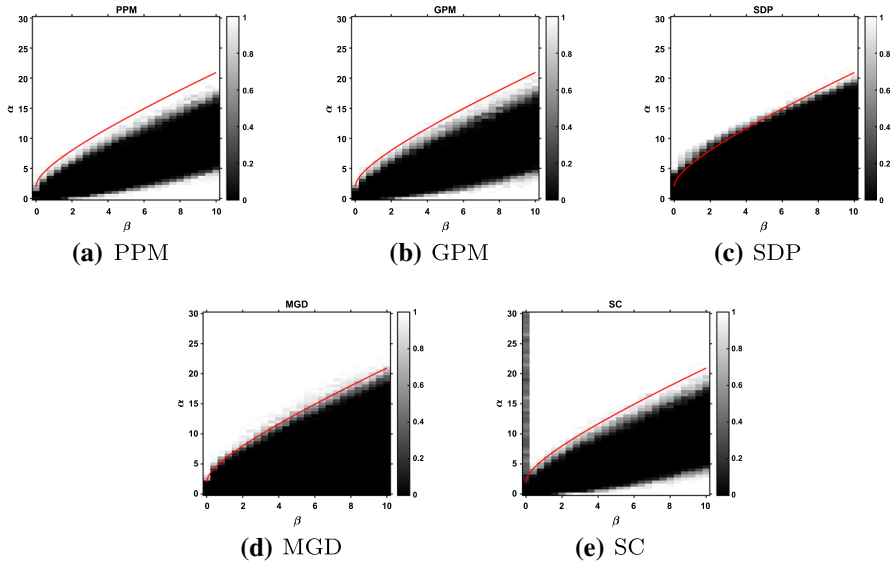


**Fig. 1** Phase transition in graphs generated by the SBM with self-loops: The  $x$ -axis is  $\beta$ , which ranges from 0 to 10 with an increment of 2; the  $y$ -axis is  $\alpha$ , which ranges from 0 to 30 with an increment of 5. Darker pixels represent lower empirical probability of success. The red curve is the information-theoretic threshold  $\sqrt{\alpha} - \sqrt{\beta} = \sqrt{2}$

**Recovery-in-SBM.** All the experiments are conducted on a PC with 16GB memory and Intel(R) Core(TM) i5-8600 3.10GHz CPU.

#### 4.1 Phase transition and computation time

We first examine the phase transition property and runtime of the aforementioned methods for recovering communities in graphs that are randomly generated according to the binary symmetric SBM, both with and without self-loops (see Definition 1 and Sect. 3.4.1). We choose  $n = 300$  in the experiments and let the parameters  $\alpha$  and  $\beta$  in (2) vary from 0 to 30 and 0 to 10 with increments of 0.5 and 0.4, respectively. For every pair of  $\alpha$  and  $\beta$ , we generate 40 instances and calculate, for all the methods, the ratio of exact recovery. The simulation results are presented in Figs. 1, 2 and Table 1. It can be seen that all methods exhibit a phase transition phenomenon and the recovery performance of PPM is slightly better than the other three methods. Moreover, Figs. 1a and 2a suggest that PPM can achieve the optimal recovery threshold on graphs with and without self-loops, respectively. This supports the results in Theorem 1 and Sect. 3.4.1. In Table 1, we record the total CPU time consumed by each approach for completing the phase transition experiment. It can be observed from the table that PPM is comparable to GPM, slightly better than SC, and substantially faster than SDP and MGD.



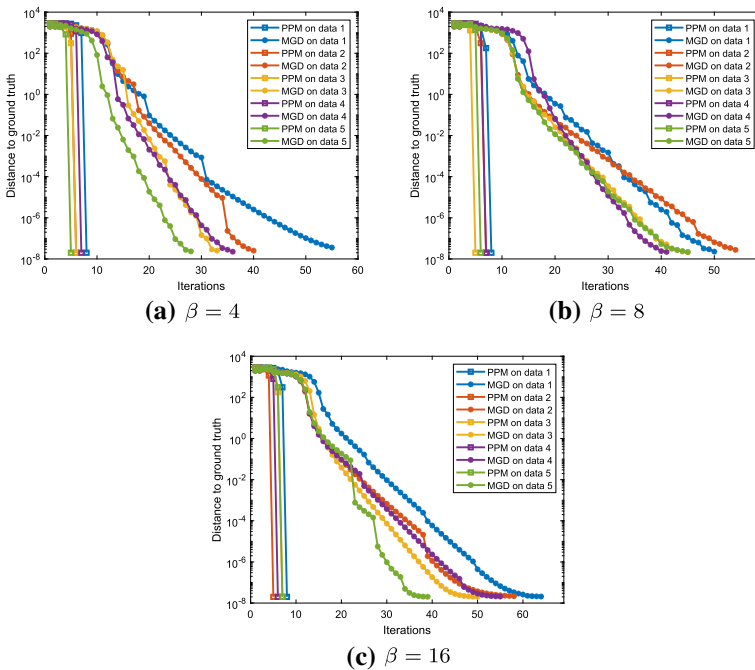
**Fig. 2** Phase transition in graphs generated by the SBM without self-loops: The x-axis is  $\beta$ , which ranges from 0 to 10 with an increment of 2; the y-axis is  $\alpha$ , which ranges from 0 to 30 with an increment of 5. Darker pixels represent lower empirical probability of success. The red curve is the information-theoretic threshold  $\sqrt{\alpha} - \sqrt{\beta} = \sqrt{2}$

**Table 1** Phase transition: Total CPU time (in seconds) of the different approaches

Methods	PPM	GPM	SDP	MGD	SC
Self-loops	18	<b>14</b>	8195	922	104
No self-loop	19	<b>14</b>	8356	932	103

## 4.2 Convergence performance

Next, we study the convergence performance of PPM and MGD, both of which have similar per-iteration cost, and report the number of iterations needed to exactly identify the two communities in graphs generated according to the binary symmetric SBM. We do not report the performance of GPM, SDP, and SC, as GPM and PPM have very similar performance, SDP cannot be solved to high accuracy using ADMM, and SC can be directly solved using the MATLAB function `eigs`. We conduct 3 sets of numerical tests each on graphs with and without self-loops, which correspond to  $\beta \in \{4, 8, 16\}$ . In each set, we generate 5 graphs of dimension  $n = 2000$ , which correspond to  $\alpha = (\sqrt{\beta} + \sqrt{2})^2 + i$  for  $i \in \{1, 2, 3, 4, 5\}$ . Such a setting ensures that the information-theoretic threshold for exact recovery is met. Let  $\mathbf{x}^k$  and  $\mathbf{Q}^k$  denote  $k$ -th iterate of PPM and MGD, respectively. In Figs. 3 and 4, we plot the distances of the iterates to the ground truth  $\|\mathbf{x}^k \mathbf{x}^{kT} - \mathbf{x}^* \mathbf{x}^{*T}\|_F$  and  $\|\mathbf{Q}^k \mathbf{Q}^{kT} - \mathbf{x}^* \mathbf{x}^{*T}\|_F$  against the iteration number for PPM and MGD, respectively. It can be observed that PPM exhibits a finite termination phenomenon and converges to the ground truth much



**Fig. 3** Convergence performance on graphs generated by the SBM with self-loops: The  $x$ -axis is number of iterations, the  $y$ -axis is distance to ground truth, which is given by  $\|\mathbf{x}^k \mathbf{x}^{kT} - \mathbf{x}^* \mathbf{x}^{*T}\|_F$  for PPM and  $\|\mathbf{Q}^k \mathbf{Q}^{kT} - \mathbf{x}^* \mathbf{x}^{*T}\|_F$  for MGD. Here,  $\mathbf{x}^k$  and  $\mathbf{Q}^k$  are the  $k$ -th iterates generated by the PPM and the MGD, respectively

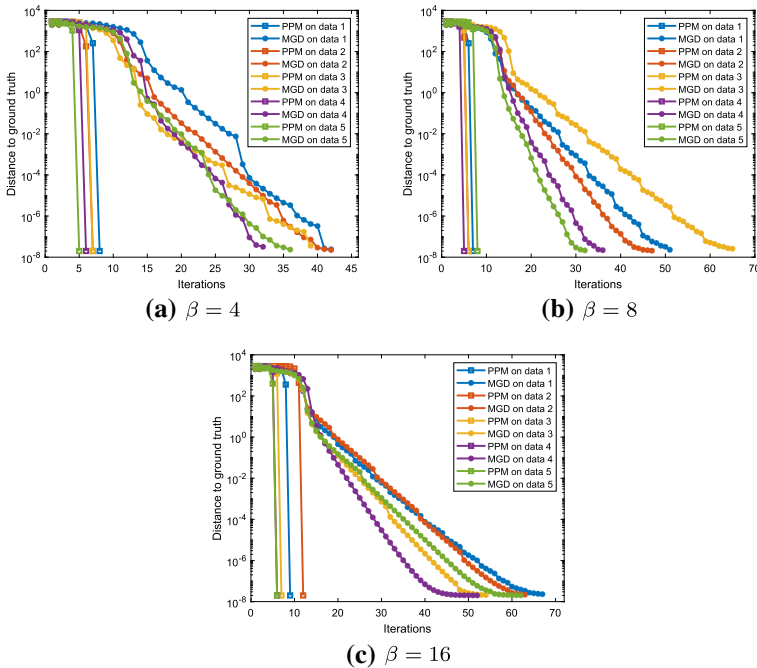
faster than MGD in graphs both with and without self-loops. This also corroborates the iteration complexity established in Theorem 1 and Sect. 3.4.1.

### 4.3 Computational efficiency

In this sub-section, we compare the computational efficiency of our proposed method with GPM, MGD, SDP, and SC on both synthetic and real data sets. For the synthetic data sets, we fix  $\beta = 16$ ,  $\alpha = (\sqrt{\beta} + \sqrt{2})^2 + 1$ , and generate three graphs of dimensions  $n = 2000$ ,  $10000$ , and  $20000$ , respectively. For the real ones, we use the data sets *polbooks* and *polblogs* downloaded from UF Sparse Matrix Collection [14].<sup>3</sup> Since these real-world networks have unbalanced or multiple communities, we extract 2 balanced communities from them. The sizes of each community extracted from *polbooks* and *polblogs* are 43 and 732, respectively. The stopping criteria for the tested algorithms are set as follows. For PPM, we terminate it when  $\|\mathbf{x}^k - \mathbf{y}^k\|_2 < 10^{-3}$  for some  $\mathbf{y}^k \in \mathcal{P}(\mathbf{x}^k + \mathbf{A}\mathbf{x}^k)$ ; for GPM, we terminate it when  $\|\mathbf{x}^k - \mathbf{y}^k\|_2 < 10^{-3}$  for some  $\mathbf{y}^k \in \text{sgn}(\mathbf{x}^k + \mathbf{A}\mathbf{x}^k - \mathbf{1}_n^T \mathbf{A} \mathbf{1}_n / n^2 \cdot \mathbf{1}_n^T \mathbf{x}^k \mathbf{1}_n)$ ; for MGD, we terminate it when the norm of the manifold gradient is less than  $10^{-3}$ ; for ADMM, we terminate it when

<sup>3</sup> <https://sparse.tamu.edu/>.

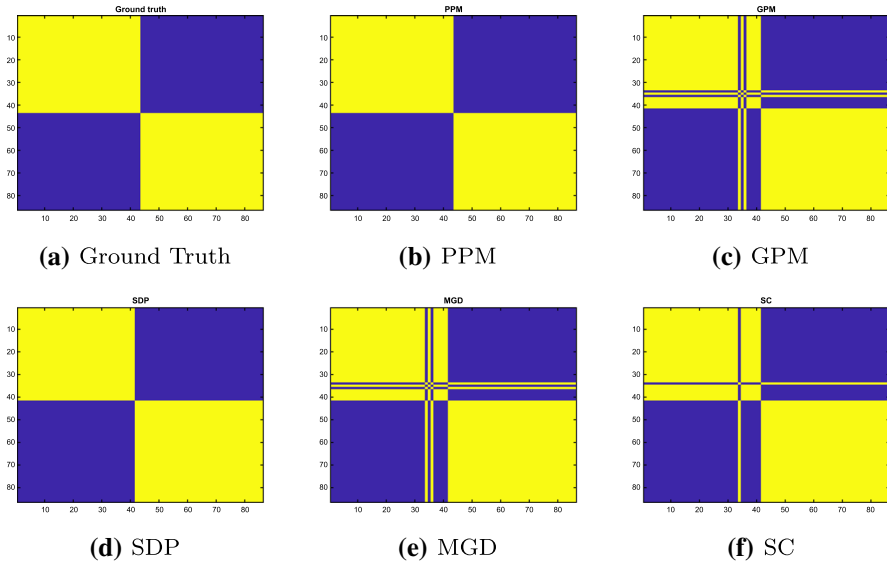




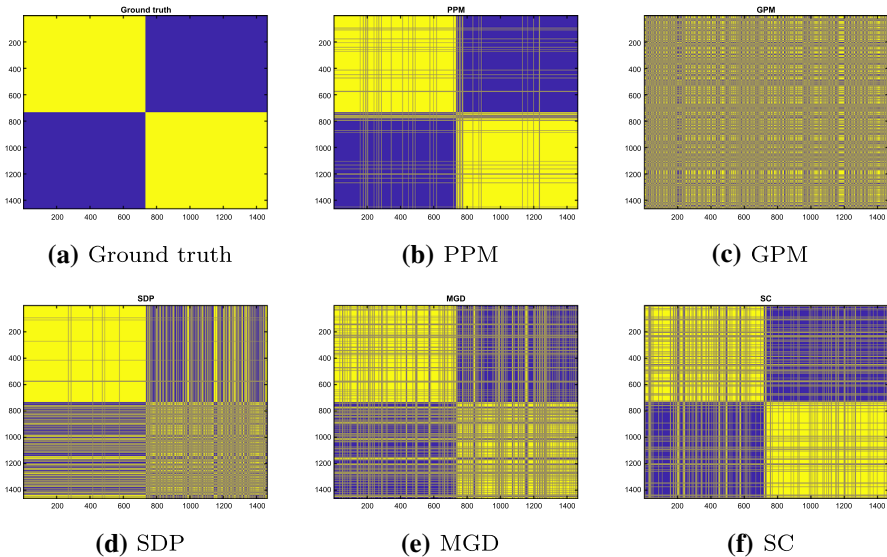
**Fig. 4** Convergence performance on graphs generated by the SBM without self-loops: The  $x$ -axis is number of iterations, the  $y$ -axis is distance to ground truth, which is given by  $\|\mathbf{x}^k \mathbf{x}^{kT} - \mathbf{x}^* \mathbf{x}^{*T}\|_F$  for PPM and  $\|\mathbf{Q}^k \mathbf{Q}^{kT} - \mathbf{x}^* \mathbf{x}^{*T}\|_F$  for MGD. Here,  $\mathbf{x}^k$  and  $\mathbf{Q}^k$  are the  $k$ -th iterates generated by the PPM and the MGD, respectively

the norm of the difference of two consecutive iterates is less than  $10^{-3}$ . No stopping criterion is needed for SC as it simply employs the MATLAB `eigs` function with some post-processing. We run each algorithm 10 times from randomly generated initial points and select the best solution (in terms of function value) as its recovery solution. Moreover, we set the maximum iteration number as 2000 for every algorithm. To compare the computational efficiency of the tested algorithms, we record their CPU time, averaged over 10 runs, and present the results in Table 2. It can be observed that our proposed method is nearly as fast as GPM, slightly better than MGD and SC, and substantially faster than SDP.

All the tested methods can achieve exact recovery on synthetic data sets. Their recovery performance on the two real data sets *polbooks* and *polblogs* are presented in Figs. 5 and 6, respectively. According to the ground truth of *polbooks*, the number of misclassified vertices by PPM, GPM, SDP, MGD, and SC are 0, 4, 1, 4, and 3, respectively. As for *polblogs*, the number of misclassified vertices by PPM, GPM, SDP, MGD, and SC are 64, 698, 289, 294, and 194, respectively. These results demonstrate that our proposed method is comparable to SDP, MGD, SC and is better than GPM in terms of recovery performance on the two real data sets.



**Fig. 5** Recovery performance on the *polbooks* network: The *x*-axis and *y*-axis give the labels of the vertices (from vertex 1 to vertex 86) in the network *polbooks*



**Fig. 6** Recovery performance on the *polblogs* network: The *x*-axis and *y*-axis give the labels of the vertices (from vertex 1 to vertex 1464) in the network *polblogs*

**Table 2** CPU times (in seconds) of the algorithms on synthetic and real data sets

	PPM	GPM	SDP	MGD	SC
$n = 2000$	0.005	<b>0.004</b>	79.47	0.313	0.014
$n = 10000$	0.031	<b>0.028</b>	—*	2.562	0.113
$n = 20000$	<b>0.074</b>	<b>0.074</b>	—	4.687	0.268
<i>polbooks</i>	0.003	<b>0.002</b>	0.172	0.059	0.098
<i>polblogs</i>	<b>0.019</b>	0.102	492.1	6.118	0.021

\* “—” denotes out of memory

## 5 Conclusions

In this work, we proposed a two-stage iterative algorithm that provably achieves exact recovery down to the information-theoretic limit in the binary symmetric SBM and has a runtime of  $\mathcal{O}(n \log^2 n / \log \log n)$ . The complexity bound is among the best for algorithms that have the same recovery performance. In the process of establishing our main results, we developed new analyses of the orthogonal iterations and projected power iterations used in our algorithm, which could be of independent interest. Our numerical results on synthetic and real data sets demonstrate the strong recovery performance and high computational efficiency of the proposed algorithm. A natural future direction is to extend the proposed approach to tackle recovery tasks in more general SBMs.

## References

1. Abbe, E.: Community detection and stochastic block models. *Foundations and Trends® in Communications and Information Theory* **14**(1–2), 1–162 (2018)
2. Abbe, E., Bandeira, A.S., Hall, G.: Exact recovery in the stochastic block model. *IEEE Trans. Inf. Theory* **62**(1), 471–487 (2016)
3. Abbe, E., Fan, J., Wang, K., Zhong, Y.: Entrywise eigenvector analysis of random matrices with low expected rank. *Annals Stat.* **48**(3), 1452–1474 (2020)
4. Abbe, E., Sandon, C.: Community detection in general stochastic block models: Fundamental limits and efficient algorithms for recovery. In: *Proceedings of the 2015 IEEE 56th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 670–688 (2015)
5. Absil, P.A., Edelman, A., Koev, P.: On the largest principal angle between random subspaces. *Linear Algebra Appl.* **414**(1), 288–294 (2006)
6. Amini, A.A., Levina, E.: On semidefinite relaxations for the block model. *Annals Stat.* **46**(1), 149–179 (2018)
7. Bandeira, A.S.: Random Laplacian matrices and convex relaxations. *Found. Comput. Math.* **18**(2), 345–379 (2018)
8. Bandeira, A.S., Boumal, N., Voroninski, V.: On the low-rank approach for semidefinite programs arising in synchronization and community detection. In: *Proceedings of the 29th Annual Conference on Learning Theory (COLT 2016)*, pp. 361–382 (2016)
9. Blum, M., Floyd, R.W., Pratt, V., Rivest, R.L., Tarjan, R.E.: Time bounds for selection. *J. Comput. Syst. Sci.* **7**(4), 448–461 (1973)
10. Boumal, N., Absil, P.A., Cartis, C.: Global rates of convergence for nonconvex optimization on manifolds. *IMA J. Numer. Anal.* **39**(1), 1–33 (2018)
11. Burer, S., Monteiro, R.D.: A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. *Math. Program.* **95**(2), 329–357 (2003)

12. Chi, Y., Lu, Y.M., Chen, Y.: Nonconvex optimization meets low-rank matrix factorization: an overview. *IEEE Trans. Signal Process.* **67**(20), 5239–5269 (2019)
13. Cline, M.S., Smoot, M., Cerami, E., Kuchinsky, A., Landys, N., Workman, C., Christmas, R., Avila-Campilo, I., Creech, M., Gross, B., et al.: Integration of biological networks and gene expression data using cytoscape. *Nat. Protocols* **2**(10), 2366 (2007)
14. Davis, T.A., Hu, Y.: The University of Florida sparse matrix collection. *ACM Trans. Math. Softw.* **38**(1), (2011)
15. Decelle, A., Krzakala, F., Moore, C., Zdeborová, L.: Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Phys. Rev. E* **84**(6), 066106 (2011)
16. Fortunato, S.: Community detection in graphs. *Phys. Reports* **486**(3–5), 75–174 (2010)
17. Gao, C., Ma, Z., Zhang, A.Y., Zhou, H.H.: Achieving optimal misclassification proportion in stochastic block models. *J. Mach. Learn. Res.* **18**(1), 1980–2024 (2017)
18. Garey, M.R., Johnson, D.S., Stockmeyer, L.: Some simplified NP-complete problems. In: *Proceedings of the 6th Annual ACM Symposium on Theory of Computing*, pp. 47–63 (1974)
19. Girvan, M., Newman, M.E.: Community structure in social and biological networks. *Proc. Nat. Acad. Sci.* **99**(12), 7821–7826 (2002)
20. Golub, G.H., Van Loan, C.F.: *Matrix Comput*, 4th edn. The Johns Hopkins University Press, Baltimore (2013)
21. Hajek, B., Wu, Y., Xu, J.: Achieving exact cluster recovery threshold via semidefinite programming. *IEEE Trans. Inf. Theory* **62**(5), 2788–2797 (2016)
22. Lei, J., Rinaldo, A.: Consistency of spectral clustering in stochastic block models. *Annals Stat.* **43**(1), 215–237 (2015)
23. Li, X., Zhu, Z., So, A.M.C., Vidal, R.: Nonconvex robust low-rank matrix recovery. *SIAM J. Optim.* **30**(1), 660–686 (2020)
24. Liu, H., Pun, Y.M., So, A.M.C.: Local strong convexity of maximum-likelihood TDOA-based source localization and its algorithmic implications. In: *Proceedings of the IEEE 7th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP 2017)*, pp. 1–5 (2017)
25. Liu, H., Yue, M.C., So, A.M.C.: On the estimation performance and convergence rate of the generalized power method for phase synchronization. *SIAM J. Optim.* **27**(4), 2426–2446 (2017)
26. Liu, H., Yue, M.C., So, A.M.C., Ma, W.K.: A discrete first-order method for large-scale MIMO detection with provable guarantees. In: *Proceedings of the 18th IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC 2017)*, pp. 669–673 (2017)
27. Ma, C., Wang, K., Chi, Y., Chen, Y.: Implicit regularization in nonconvex statistical estimation: gradient descent converges linearly for phase retrieval, matrix completion, and blind deconvolution. *Found. Comput. Math.* **20**, 451–632 (2020)
28. Massoulié, L.: Community detection thresholds and the weak Ramanujan property. In: *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, pp. 694–703 (2014)
29. Mossel, E., Neeman, J., Sly, A.: Consistency thresholds for the planted bisection model. *Electron. J. Probability* **21**(21), 1–24 (2016)
30. Newman, M.E., Girvan, M.: Finding and evaluating community structure in networks. *Phys. Rev. E* **69**(2), 026113 (2004)
31. Pun, Y.M., So, A.M.C.: Dynamic regret bound for moving target tracking based on online time-of-arrival measurements. In: *Proceedings of the 59th IEEE Conference on Decision and Control (CDC 2020)*, pp. 5968–5973 (2020)
32. Sun, R., Li, D., Liang, S., Ding, T., Srikant, R.: The global landscape of neural networks: an overview. *IEEE Signal Process. Magazine* **37**(5), 95–108 (2020)
33. Sun, R.Y.: Optimization for deep learning: an overview. *J. Oper. Res. Soc. China* **8**(2), 249–294 (2020)
34. Trefethen, L.N., Bau, D., III.: *Numerical Linear Algebra*. SIAM, New Delhi (1997)
35. Vaswani, N.: Non-convex structured phase retrieval. *IEEE Signal Process. Magazine* **37**(5), 67–77 (2020)
36. Vershynin, R.: *High-Dimensional Probability: an Introduction with Applications in Data Science*, Cambridge Series in Statistical and Probabilistic Mathematics, vol. 47. Cambridge University Press, Cambridge (2018)
37. Wang, P., Zhou, Z., 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 (ICML 2020)*, pp. 10126–10135 (2020)

38. Yun, S.Y., Proutiere, A.: Optimal cluster recovery in the labeled stochastic block model. In: D.D. Lee, M. Sugiyama, U.V. Luxburg, I. Guyon, R. Garnett (eds.) *Advances in Neural Information Processing Systems 29: Proceedings of the 2016 Conference*, pp. 965–973 (2016)
39. Zhong, Y., Boumal, N.: Near-optimal bounds for phase synchronization. *SIAM J. Optim.* **28**(2), 989–1016 (2018)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.