The Annals of Statistics
2020, Vol. 48, No. 6, 3488–3512
https://doi.org/10.1214/19-AOS1939

CLUSTERING IN BLOCK MARKOV CHAINS

BY JARON SANDERS[1,2,*], ALEXANDRE PROUTIÈRE[1,†] AND SE-YOUNG YUN[3]

[1]*Department of Automatic Control, School of Electrical Engineering, KTH Royal Institute of Technology,* [*]*jaron.sanders@tue.nl;* [†]*alepro@kth.se*

[2]*Department of Quantum & Computer Engineering, Faculty of Electrical Engineering, Mathematics & Computer Science, Delft University of Technology, j.sanders@tudelft.nl*

[3]*Department of Industrial & Systems Engineering, Graduate School of AI, Korea Advanced Institute of Science and Technology, yunseyoung@kaist.ac.kr*

This paper considers cluster detection in Block Markov Chains (BMCs). These Markov chains are characterized by a block structure in their transition matrix. More precisely, the n possible states are divided into a finite number of K groups or clusters, such that states in the same cluster exhibit the same transition rates to other states. One observes a trajectory of the Markov chain, and the objective is to recover, from this observation only, the (initially unknown) clusters. In this paper, we devise a clustering procedure that accurately, efficiently and provably detects the clusters. We first derive a fundamental information-theoretical lower bound on the detection error rate satisfied under any clustering algorithm. This bound identifies the parameters of the BMC, and trajectory lengths, for which it is possible to accurately detect the clusters. We next develop two clustering algorithms that can together accurately recover the cluster structure from the shortest possible trajectories, whenever the parameters allow detection. These algorithms thus reach the fundamental detectability limit, and are optimal in that sense.

1. Introduction. The ability to accurately discover all hidden relations between items that share similarities is of paramount importance to a wide range of disciplines. Clustering algorithms in particular are employed throughout social sciences, biology, computer science, economics and physics. The reason these techniques have become prevalent is that once clusters of similar items have been identified, any subsequent analysis or optimization procedure benefits from a powerful reduction in dimensionality.

The canonical Stochastic Block Model (SBM), originally introduced in [13], has become the benchmark to investigate the performance of cluster detection algorithms. This model generates random graphs that contain groups of similar vertices. Vertices within the same group are similar in that they share the same average edge densities to the other vertices. More precisely, if the set of n vertices $\mathcal{V}$ is, for example, partitioned into two groups $\mathcal{V}_1$ and $\mathcal{V}_2$, an edge is drawn between two vertices $x, y \in \mathcal{V}$ with probability $p \in (0, 1)$ if they belong to the same group, and with probability $q \in (0, 1)$, $p \neq q$, if they belong to different groups. Edges are drawn independently of all other edges. Within the context of the SBM and its generalizations, the problem of cluster detection is to infer the clusters from observations of a realization of the random graph with the aforementioned structure.

This paper deviates by considering the problem of cluster detection when the observation is instead the sample path of a Markov chain over the set of vertices. Specifically, we introduce the Block Markov Chain (BMC), which is a Markov chain characterized by a block structure in its transition matrix. States that are in the same cluster are similar in the sense that they have the same transition rates. The goal is to detect the clusters from an observed

Received February 2018; revised July 2019.
MSC2020 subject classifications. 62H30, 60J10, 60J20.
Key words and phrases. Clustering, Markov chains, mixing times, community detection, change of measure, asymptotic analysis, information theory.

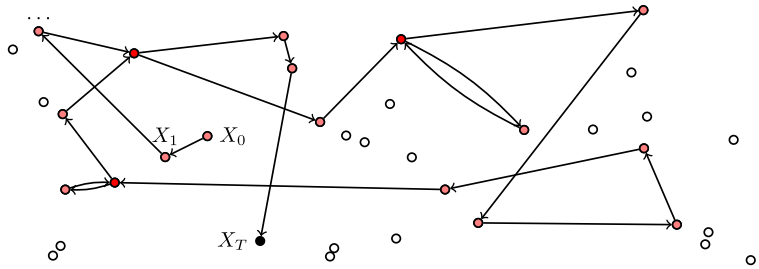


FIG. 1. *The goal of this paper is to infer the hidden cluster structure underlying a Markov chain* $\{X_t\}_{t\geq 0}$, *from one observation of a sample path* $X_0, X_1, \ldots, X_T$ *of length* T.

sample path $X_0, X_1, \ldots, X_T$ of the Markov chain (Figure 1). This new clustering problem is mathematically more challenging because consecutive samples of the random walk are *not* independent: besides noise, there is bias in a sample path. Intuitively though there is hope for accurate cluster detection if the Markov chain can get close to stationarity within T steps. Indeed, as we will show, the mixing time [18] of the BMC plays a crucial role in the detectability of the clusters.

Clustering in BMCs is motivated by Reinforcement Learning (RL) problems [27] with large state spaces. These problems have received substantial attention recently due to the wide spectrum of their applications in robotics, games, medicine, finance, etc. RL problems are concerned with the control of dynamical systems modeled as Markov chains whose transition kernels are initially unknown. The objective is to identify an optimal control policy as early as possible by observing the trajectory of a Markov chain generated under some known policy. The time it takes to learn efficient policies using standard algorithms such as Q-learning dramatically increases with the number of possible states, so that these algorithms become useless when the state space is prohibitively large. In most practical problems, however, different states may yield similar reward and exhibit similar transition probabilities to other states, that is, states can be grouped into clusters. In this scenario, it becomes critical to learn and leverage this structure in order to speed up the learning process. In this paper, we consider *uncontrolled* Markov chains, and we aim to identify clusters of states as quickly as possible. In the future, we hope to extend the techniques developed here for an uncontrolled BMC to the more general case of controlled Markov chains, and hence to devise reinforcement learning algorithms that will efficiently exploit an underlying cluster structure. The idea of clustering states in reinforcement learning to speed up the learning process has been investigated in [26] and [23], but no theoretical guarantees were provided in these early papers.

This paper answers two important questions for the problem of cluster detection on BMCs. First, we derive a fundamental information-theoretical clustering error lower bound. The latter allows us to identify the parameters of the BMC and the sample path lengths T for which it is theoretically impossible to accurately detect the underlying cluster structure. Second, we develop two clustering algorithms that when combined, are able to accurately detect the underlying cluster structure from the shortest possible sample paths, whenever the parameters of the BMC allow detection, and that provably work as $n \to \infty$. These algorithms thus reach the fundamental detectability limit, and are optimal in that sense.

1.1. *Related work.* Clustering in the SBM and the BMCs may be seen as similar problems: the objective in both cases is to infer the cluster structure from random observations made on the relationships between pairs of vertices. However, the way these observations are gathered differ significantly in the SBM and the BMCs. In the SBM, these observations are independent random variables, which allows the use of theoretical developments in random

matrices with independent entries. In the BMCs on the contrary, observations are successive states of a Markov chain, and hence are not independent. Furthermore, observed edges in the SBM are scattered and undirected, whereas in a BMC, the observed path is a concatenation of directed edges. Generally, the probabilities to move from state x to state y and from y to x are different. Finally, the sparsity of the observations in the BMC is controlled by the length T of the observed sample path, while it is hard-coded in the SBM. For all these reasons, it is difficult to quantitatively compare or relate the recovery rates in the two models. Nevertheless, techniques as those used in the SBM can be exploited in the analysis of the BMC if they are properly extended to handle the differences between the two models. For this reason, we now provide a brief survey of the techniques and results available for the SBM.

Significant advances have been made on cluster recovery within the context of the SBM and its generalizations. We defer the reader to [9] for an extensive overview. Substantial focus has in particular been on characterizing the set of parameters for which some recovery objectives can be met.

In the *sparse* regime, that is, when the average degree of vertices is $O(1)$, necessary and sufficient conditions on the parameters have been identified under which it is possible to extract clusters that are positively correlated with the true clusters [7, 20, 21]. More precisely, for example, if $p = a/n$ and $q = b/n$ and in the case of two clusters of equal sizes, it was conjectured in [7] that $a - b \geq \sqrt{2(a+b)}$ is a necessary and sufficient condition for the existence of algorithms that can *detect* the clusters (in the sense that they perform better than a random assignment of items to clusters). This result was established in [21] (necessary condition) and in [20] (sufficient condition).

In the *dense* regime, that is, when the average degree is $\omega(1)$, it is possible to devise algorithms under which the proportion of misclassified vertices vanishes as the size of the graph grows large [31]. In this case, one may actually characterize the *minimal* asymptotic (as n grows large) classification error, and develop clustering algorithms achieving this fundamental limit [33]. We may further establish conditions under which asymptotic *exact* cluster recovery is possible [1–3, 10, 14, 22, 32, 33].

This paper draws considerable inspiration from [31–33]. Over the course of these papers, the authors consider the problem of clustering in the Labeled Stochastic Block Model (LSBM), which is a generalization of the SBM. They identify the set of LSBM-parameters for which the clusters can be detected using change-of-measure arguments, and develop algorithms based on spectral methods that achieve this fundamental performance limit. Our contributions in this paper include the extension of the approaches to the context of Markov chains. This required us in particular to design novel changes-of-measure, carefully incorporate the effect of mixing, deal with new and nonconvex log-likelihood functions, and widen the applicability of spectral methods to random matrices with bias. Note that we restrict the analysis in this paper to the case that the number of clusters K is known. This reduces the complexity of the analysis. Based on the findings in [31–33], however, we are confident that this assumption can be relaxed in future work.

1.2. *Methodology.* Similar to the extensive efforts for the SBM, we will first identify parameters of the BMC for which it is theoretically possible to detect the clusters. To this aim, we use techniques from information theory to derive a lower bound on the number of misclassified states that holds for any classification algorithm. This relies on a powerful change-of-measure argument, originally explored in [17] in the context of online stochastic optimization. First, we relate the probability of misclassifying a state in the BMC to a log-likelihood ratio that the sample path was generated by a perturbed Markov chain instead. Then, given any BMC, we show how to construct a perturbed Markov chain that assigns a

nonzero probability to the event that all clustering algorithms misclassify at least one particular state. Finally, we maximize over all possible perturbations to get the best possible lower bound that holds for any algorithm.

We will further provide a clustering algorithm that achieves this fundamental limit. Specifically, the algorithm consists of two steps. The first step consists in applying a classical *Spectral Clustering Algorithm*. This algorithm essentially creates a rank-K approximation of a random matrix corresponding to the empirical transition rates between any pair of states, and then uses a K-means algorithm [19] to cluster all states. We show that this first step clusters the majority of states roughly correctly. Next, we introduce the *Cluster Improvement Algorithm*. This algorithm uses the rough structure learned from the Spectral Clustering Algorithm, together with the sample path, to move each individual state into the cluster the state most likely belongs to. This is achieved through a recursive, local maximization of a log-likelihood ratio.

The key difference between clustering in SBMs and clustering in BMCs is that instead of observing (the edges of) a random graph, we here try to infer the cluster structure from an as short as possible sample path of the Markov chain. This necessitates a careful analysis of the mixing time of the Markov chain [18], for which we use a rate of convergence result in terms of Dobrushin's ergodicity coefficient [6]. The observed sample path will be inherently noisy and biased by construction. The noise and bias within the sample path have to first be related to the spectrum of the random matrix recording the number of times transitions between any two states have been observed. This is done by using techniques from [11]. The spectrum of this random matrix has then to be analyzed which constitutes a major challenge. Indeed, most results investigating the spectrum of random matrices hold for matrices with independent and weakly dependent entries [12, 15, 16, 28–30], or when the transition matrix of the Markov chain itself is random [4, 5]. Our random matrix has dependent entries, but by taking proof inspiration from [8], using concentration results from [24], and smartly leveraging the way it is constructed from the observed sample path and Markov property, the analysis of its spectrum can be conducted.

1.3. *Overview.* This paper is structured as follows. We introduce the BMC in Section 2. Section 3 provides an overview of our results and our algorithms. We assess the performance of both algorithms, that is, we quantify their asymptotic error rates. Section 4 discusses several numerical experiments designed to test the algorithms. We subsequently prove our results by first deriving an information lower bound and developing an optimal change-of-measure in Section 5, and then by developing the Spectral Clustering Algorithm in Section 6 and the Cluster Improvement Algorithm in Section 7. The details of the proofs can be found in the Supplementary Material [25].

Notation. For any two sets $\mathcal{A}, \mathcal{B} \subseteq \mathcal{V} \triangleq \{1, \dots, n\}$, we define their symmetric difference by $\mathcal{A} \triangle \mathcal{B} = \{\mathcal{A} \setminus \mathcal{B}\} \cup \{\mathcal{B} \setminus \mathcal{A}\}$. For any two numbers $a, b \in \mathbb{R}$, we introduce the shorthand notation $a \wedge b = \min\{a, b\}$ and $a \vee b = \max\{a, b\}$. For any n-dimensional vector $x = (x_1, \dots, x_n)^{\mathrm{T}} \in \mathbb{R}^n$, we define its l_p norms by

$$\|x\|_p = \left(\sum_{r=1}^{n} |x_i|^p \right)^{1/p} \quad \text{where } p \in [1, \infty).$$

The n-dimensional unit vector of which the rth component equals 1 will be denoted by $e_{n,r}$, and the n-dimensional vector for which all elements $r \in \mathcal{A} \subseteq \{1, \dots, n\}$ equal 1 will be denoted by $1_{\mathcal{A}}$. For any $m \times n$ matrix $A \in \mathbb{R}^{m \times n}$, we indicate its rows by $A_{r,\cdot}$ for

$r = 1, \dots, m$ and its columns by $A_{\cdot,c}$ for $c = 1, \dots, n$. We also introduce the shorthand notation $A_{\mathcal{A},\mathcal{B}} = \sum_{x\in\mathcal{A}} \sum_{y\in\mathcal{B}} A_{x,y}$ for all subsets $\mathcal{A}, \mathcal{B} \subseteq \mathcal{V}$. Its Frobenius norm and spectral norm are defined by

$$\|A\|_{\mathrm{F}} = \sqrt{\sum_{r=1}^{m}\sum_{c=1}^{n} A_{r,c}^2}, \qquad \|A\| = \sup_{b\in\mathbb{S}^{n-1}} \{\|Ab\|_2\},$$

respectively. Here, $\mathbb{S}^{n-1} = \{x = (x_1, \dots, x_n) \in (0,1)^n : \|x\|_2 = 1\}$ denotes the n-dimensional unit sphere. We define the probability simplex of dimension $n-1$ by $\Delta^{n-1} = \{x \in (0,1)^n : \|x\|_1 = 1\}$ as well as the set of left-stochastic matrices by $\mathbb{\Delta}^{n\times(n-1)} = \{((x_{1,1}, \dots, x_{1,n}), \dots, (x_{n,1}, \dots, x_{n,n})) \in [0,1]^{n\times n} : \sum_{c=1}^{n} x_{r,c} = 1 \text{ for } r = 1, \dots, n\}$ similarly.

In our asymptotic analyses, we write $f(n) \sim g(n)$ if $\lim_{n\to\infty} f(n)/g(n) = 1$, $f(n) = o(g(n))$ if $\lim_{n\to\infty} f(n)/g(n) = 0$ and $f(n) = O(g(n))$ if $\limsup_{n\to\infty} f(n)/g(n) < \infty$. Whenever $\{X_n\}_{n=1}^{\infty}$ is a sequence of real-valued random variables and $\{a_n\}_{n=1}^{\infty}$ a deterministic sequence, we write

$$\begin{aligned} X_n = o_{\mathbb{P}}(a_n) \quad &\Leftrightarrow \quad \mathbb{P}\left[\left|\frac{X_n}{a_n}\right| \geq \delta\right] \to 0 \quad \forall_{\delta>0} \\ &\Leftrightarrow \quad \forall_{\varepsilon,\delta}\exists_{N_{\varepsilon,\delta}} : \mathbb{P}\left[\left|\frac{X_n}{a_n}\right| \geq \delta\right] \leq \varepsilon \quad \forall_{n>N_{\varepsilon,\delta}} \end{aligned}$$

and

$$X_n = O_{\mathbb{P}}(a_n) \quad \Leftrightarrow \quad \forall_{\varepsilon}\exists_{\delta_\varepsilon,N_\varepsilon} : \mathbb{P}\left[\left|\frac{X_n}{a_n}\right| \geq \delta_\varepsilon\right] \leq \varepsilon \quad \forall_{n>N_\varepsilon}.$$

Similarly, $X_n = \Omega_{\mathbb{P}}(a_n)$ denotes $\forall_\varepsilon \exists_{\delta_\varepsilon,N_\varepsilon} : \mathbb{P}[|X_n/a_n| \leq \delta_\varepsilon] \leq \varepsilon \forall_{n>N_\varepsilon}$, and $X_n \asymp_{\mathbb{P}} (a_n)$ means $\forall_\varepsilon \exists_{\delta_\varepsilon^-,\delta_\varepsilon^+,N_\varepsilon} : \mathbb{P}[\delta_\varepsilon^- \leq |X_n/a_n| \leq \delta_\varepsilon^+] \geq 1-\varepsilon \forall_{n>N_\varepsilon}$.

2. Block Markov Chains (BMCs). We assume that we have n states $\mathcal{V} = \{1, \dots, n\}$, each of which is associated to one of K clusters. This means that the set of states is partitioned so that $\mathcal{V} = \bigcup_{k=1}^{K} \mathcal{V}_k$ with $\mathcal{V}_k \cap \mathcal{V}_l = \varnothing$ for all $k \neq l$. Let $\sigma(v)$ denote the cluster of a state $v \in \mathcal{V}$. We also assume that there exist constants $\alpha \in \Delta^{K-1}$ so that $\lim_{n\to\infty} |\mathcal{V}_k|/(n\alpha_k) = 1$.

For any $\alpha \in \Delta^{K-1}$ and $p \in \mathbb{\Delta}^{K\times(K-1)}$, we define the BMC $\{X_t\}_{t\geq 0}$ as follows. Its transition matrix $P \in \mathbb{\Delta}^{n\times(n-1)}$ will be defined as

$$P_{x,y} \triangleq \frac{p_{\sigma(x),\sigma(y)}}{|\mathcal{V}_{\sigma(y)}| - \mathbb{1}[\sigma(x) = \sigma(y)]} \mathbb{1}[x \neq y] \quad \text{for all } x, y \in \mathcal{V}. \tag{1}$$

Note that this Markov chain is not necessarily reversible. Furthermore, note that in this paper we assume that K, α, p are fixed, and that we study the asymptotic regime $n \to \infty$. We assume that the smallest cluster has a size linearly growing with n: $\alpha_{\min} \triangleq \min_k \alpha_k > 0$. Finally, since we are interested in clustering of the states, we will assume that $\exists_{1<\eta} : \max_{a,b,c}\{p_{b,a}/p_{c,a}, p_{a,b}/p_{a,c}\} \leq \eta$, which guarantees a minimum level of separability of the parameters.

2.1. *Equilibrium behavior.* We assume that the stochastic matrix p is such that the equilibrium distribution of $\{X_t\}_{t\geq 0}$ exists, and we will denote it by Π_x for $x \in \mathcal{V}$. By symmetry, $\Pi_x = \Pi_y \triangleq \bar{\Pi}_k$ for any two states $x, y \in \mathcal{V}_k$ for all $k = 1, \dots, K$. Consider the scaled quantity

$$\pi_k \triangleq \lim_{n\to\infty} \sum_{x\in\mathcal{V}_k} \Pi_x = \lim_{n\to\infty} |\mathcal{V}_k|\bar{\Pi}_k \quad \text{for } k = 1, \dots, K.$$

Proposition 1's proof can be found in Section SM2.1 of the Supplementary Material [25], and follows from the symmetries between the states within the same clusters and the specific scalings of P's elements.

PROPOSITION 1. *The quantity π solves $\pi^{\mathrm{T}} p = \pi^{\mathrm{T}}$, and is therefore the equilibrium distribution of a Markov chain with transition matrix p and state space $\Omega = \{1, \ldots, K\}$.*

2.2. *Mixing time.* Proposition 2 gives a bound on the mixing time $t_{\mathrm{mix}} \in (0, \infty)$, which is defined by $d(t) \triangleq \sup_{x \in \mathcal{V}} \{d_{\mathrm{TV}}(P^t_{x,\cdot}, \Pi)\}$ and $t_{\mathrm{mix}}(\varepsilon) \triangleq \min\{t \geq 0 : d(t) \leq \varepsilon\}$, where

$$d_{\mathrm{TV}}(\mu, \nu) \triangleq \frac{1}{2} \sum_{x \in \mathcal{V}} |\mu_x - \nu_x|.$$

The proof of Proposition 2 is deferred to Section SM2.2 of the Supplementary Material [25]. The result follows after bounding Dobrushin's ergodicity coefficient [6] using P's structure, and invoking a convergence rate result in terms of Dobrushin's coefficient.

PROPOSITION 2. *For any BMC with $n \geq 4/\alpha_{\min}$, $t_{\mathrm{mix}}(\varepsilon) \leq -c_{\mathrm{mix}} \ln \varepsilon$, where $c_{\mathrm{mix}} = -1/\ln(1 - 1/2\eta)$.*

Proposition 2 implies that the mixing times are short enough so that our results will hold *irrespective* of whether we assume that the Markov chain is initially in equilibrium. We will show in Section 5.4 that what is important is that the chain reaches stationarity *within* T steps (the length of the observed trajectory), and consequentially, T needs to be chosen sufficiently large with respect to n to ensure that this occurs. Throughout this paper, we therefore assume for simplicity that the chain is started from equilibrium. This eliminates the need of tracking higher order correction terms.

Examples. Figure 2 illustrates the structure of a BMC when there are $K = 2$ groups. We find after solving the balance equations that the limiting equilibrium behavior is given by $\pi_1 = p_{21}/(p_{12} + p_{21})$ and $\pi_2 = p_{12}/(p_{12} + p_{21})$.

For $K = 3$, we find after solving the balance equations that the limiting equilibrium behavior is given by

$$\pi_1 = \frac{p_{23}p_{31} + p_{21}(p_{31} + p_{32})}{Z(p)}, \qquad \pi_2 = \frac{p_{13}p_{32} + p_{12}(p_{31} + p_{32})}{Z(p)},$$

$\pi_3 = 1 - \pi_1 - \pi_2$, with $Z(p) = (p_{21} + p_{23})(p_{13} + p_{31}) + (p_{13} + p_{21})p_{32} + p_{12}(p_{23} + p_{31} + p_{32})$. Let us also illustrate the structure of the transition matrix when $\alpha = (2/10, 3/10, 5/10)$

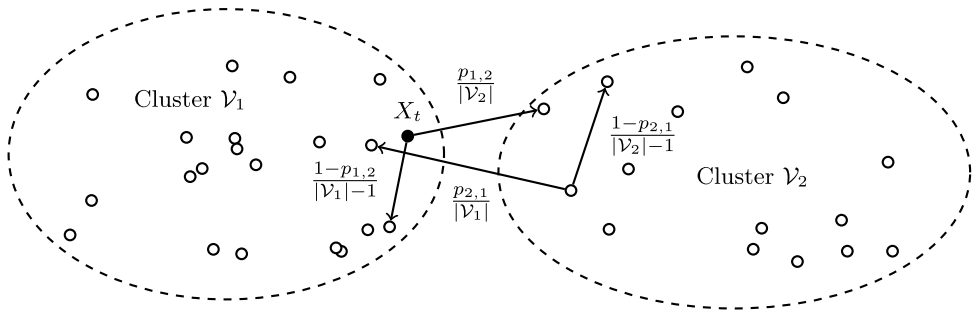


FIG. 2. *In the BMC with $K = 2$ groups $\mathcal{V}_1 \cup \mathcal{V}_2 = \mathcal{V}$, whenever the Markov chain is at some state $X_t \in \mathcal{V}_1$, it will next jump with probability $p_{1,2}$ to cluster $\mathcal{V}_2$, and with probability $1 - p_{1,2}$ to some other state in cluster $\mathcal{V}_1$. Similarly, if $X_t \in \mathcal{V}_2$, it would next jump to cluster $\mathcal{V}_1$ with probability $p_{2,1}$, or stay within its own cluster with probability $1 - p_{2,1}$.*

and $n = 10$:

$$
P = \left(\begin{array}{cc:ccc:ccccc}
0 & p_{1,1} & \frac{p_{1,2}}{3} & \frac{p_{1,2}}{3} & \frac{p_{1,2}}{3} & \frac{p_{1,3}}{5} & \frac{p_{1,3}}{5} & \frac{p_{1,3}}{5} & \frac{p_{1,3}}{5} & \frac{p_{1,3}}{5} \\
p_{1,1} & 0 & \frac{p_{1,2}}{3} & \frac{p_{1,2}}{3} & \frac{p_{1,2}}{3} & \frac{p_{1,3}}{5} & \frac{p_{1,3}}{5} & \frac{p_{1,3}}{5} & \frac{p_{1,3}}{5} & \frac{p_{1,3}}{5} \\
\hdashline
\frac{p_{2,1}}{2} & \frac{p_{2,1}}{2} & 0 & \frac{p_{2,2}}{2} & \frac{p_{2,2}}{2} & \frac{p_{2,3}}{5} & \frac{p_{2,3}}{5} & \frac{p_{2,3}}{5} & \frac{p_{2,3}}{5} & \frac{p_{2,3}}{5} \\
\frac{p_{2,1}}{2} & \frac{p_{2,1}}{2} & \frac{p_{2,2}}{2} & 0 & \frac{p_{2,2}}{2} & \frac{p_{2,3}}{5} & \frac{p_{2,3}}{5} & \frac{p_{2,3}}{5} & \frac{p_{2,3}}{5} & \frac{p_{2,3}}{5} \\
\frac{p_{2,1}}{2} & \frac{p_{2,1}}{2} & \frac{p_{2,2}}{2} & \frac{p_{2,2}}{2} & 0 & \frac{p_{2,3}}{5} & \frac{p_{2,3}}{5} & \frac{p_{2,3}}{5} & \frac{p_{2,3}}{5} & \frac{p_{2,3}}{5} \\
\hdashline
\frac{p_{3,1}}{2} & \frac{p_{3,1}}{2} & \frac{p_{3,2}}{3} & \frac{p_{3,2}}{3} & \frac{p_{3,2}}{3} & 0 & \frac{p_{3,3}}{4} & \frac{p_{3,3}}{4} & \frac{p_{3,3}}{4} & \frac{p_{3,3}}{4} \\
\frac{p_{3,1}}{2} & \frac{p_{3,1}}{2} & \frac{p_{3,2}}{3} & \frac{p_{3,2}}{3} & \frac{p_{3,2}}{3} & \frac{p_{3,3}}{4} & 0 & \frac{p_{3,3}}{4} & \frac{p_{3,3}}{4} & \frac{p_{3,3}}{4} \\
\frac{p_{3,1}}{2} & \frac{p_{3,1}}{2} & \frac{p_{3,2}}{3} & \frac{p_{3,2}}{3} & \frac{p_{3,2}}{3} & \frac{p_{3,3}}{4} & \frac{p_{3,3}}{4} & 0 & \frac{p_{3,3}}{4} & \frac{p_{3,3}}{4} \\
\frac{p_{3,1}}{2} & \frac{p_{3,1}}{2} & \frac{p_{3,2}}{3} & \frac{p_{3,2}}{3} & \frac{p_{3,2}}{3} & \frac{p_{3,3}}{4} & \frac{p_{3,3}}{4} & \frac{p_{3,3}}{4} & 0 & \frac{p_{3,3}}{4} \\
\frac{p_{3,1}}{2} & \frac{p_{3,1}}{2} & \frac{p_{3,2}}{3} & \frac{p_{3,2}}{3} & \frac{p_{3,2}}{3} & \frac{p_{3,3}}{4} & \frac{p_{3,3}}{4} & \frac{p_{3,3}}{4} & \frac{p_{3,3}}{4} & 0
\end{array}\right). \tag{2}
$$

3. Main results. In this paper, we obtain quantitative statements on the set of misclassified states,

$$
\mathcal{E} \triangleq \bigcup_{k=1}^{K} \hat{\mathcal{V}}_{\gamma^{\mathrm{opt}}(k)} \setminus \mathcal{V}_k \quad \text{where } \gamma^{\mathrm{opt}} \in \arg\min_{\gamma \in \mathrm{Perm}(K)} \left| \bigcup_{k=1}^{K} \hat{\mathcal{V}}_{\gamma(k)} \setminus \mathcal{V}_k \right|.
$$

Here, the sets $\hat{\mathcal{V}}_1, \ldots, \hat{\mathcal{V}}_K$ will always denote an approximate cluster assignment obtained from some clustering algorithm. For notational convenience, we will always number the approximate clusters so as to minimize the number of misclassifications, allowing us to forego defining it formally via a permutation.

3.1. *Information theoretical lower bound.* Our results identify an important information quantity $I(\alpha, p) \geq 0$ that measures how difficult it is to cluster in a BMC. Its role will become clear in Theorem 1. The reason we call it an information quantity stems from fact that we have derived it as the leading coefficient in an asymptotic expansion of a log-likelihood function. Note that while it resembles one, this information quantity is not a Kullback–Leibler divergence. The individual terms are weighted according to the equilibrium distribution π, and there are two extra terms.

DEFINITION. For $\alpha \in \Delta^{K-1}$ and $p \in \mathbb{M}^{(K-1)\times K}$, let

$$
I(\alpha, p) \triangleq \min_{a \neq b} I_{a,b}(\alpha, p), \tag{3}
$$

where $I_{a,b}(\alpha, p) \triangleq \{\sum_{k=1}^{K} \frac{1}{\alpha_a}(\pi_a p_{a,k} \ln \frac{p_{a,k}}{p_{b,k}} + \pi_k p_{k,a} \ln \frac{p_{k,a}\alpha_b}{p_{k,b}\alpha_a}) + (\frac{\pi_b}{\alpha_b} - \frac{\pi_a}{\alpha_a})\}$. Here, π denotes the solution to $\pi^{\mathrm{T}} p = \pi^{\mathrm{T}}$.

THEOREM 1. *An algorithm is (ε, c)-locally good at (α, p) if it satisfies $\mathbb{E}_P[|\mathcal{E}|] \leq \varepsilon$ for all BMC models constructed from the given p and partitions satisfying $||\mathcal{V}_k| - \alpha_k n| \leq c$ for all k. Assume that $T = \omega(n)$. Then there exists a strictly positive and finite constant C*

independent of n such that: *there exists no* $(\varepsilon, 1)$*-locally good clustering algorithm at* (α, p) *when*

$$\varepsilon < Cn \exp\left(-I(\alpha, p)\frac{T}{n}\big(1 + o(1)\big)\right).$$

Theorem 1 allows us to state necessary conditions for the existence of $(\varepsilon, 1)$-locally good clustering algorithms at (α, p) that either detect clusters *asymptotically accurately*, namely with $\mathbb{E}_P[|\mathcal{E}|] = o(n)$, or recover clusters *asymptotically exactly*, that is, with $\mathbb{E}_P[|\mathcal{E}|] = o(1)$.

Conditions for asymptotically accurate detection. In view of the lower bound in Theorem 1, there may exist asymptotically accurate $(\varepsilon, 1)$-locally good clustering algorithms at (α, p) only if $I(\alpha, p) > 0$ and $T = \omega(n)$.

Conditions for asymptotically exact detection. Necessary conditions for the existence of an asymptotically exact algorithm are $I(\alpha, p) > 0$ and $T - \frac{n\ln(n)}{I(\alpha,p)} = \omega(1)$. In particular, T must be larger than $n\ln(n)$. We refer to the scenario where T is of the order $n\ln(n)$ as the *critical regime*. In this regime when $T = n\ln(n)$, the necessary condition for exact recovery is $I(\alpha, p) > 1$.

Note that qualitatively, the above conditions on the number of observations for accurate and exact recovery are similar to those in the SBM. In the latter, the average degrees of vertices should be such that the average total number of edges is $\omega(n)$ [31] for accurate detection, whereas this average must be at least $cn\ln(n)$ for exact recovery [2]. Here, c is known and depends on the parameters of the SBM.

The information quantity $I(\alpha, p)$ *for* $K = 2$ *clusters.* In the case of two clusters, we study the set of parameters (α, p) of the BMCs for which $I(\alpha, p) > 0$ and $I(\alpha, p) > 1$, the latter condition being necessary in the critical regime when $T = n\ln(n)$.

A system with two clusters can be specified entirely with three parameters: α_2, $p_{1,2}$ and $p_{2,1}$. Examining the explicit expression for (3) in this case, we can conclude that $I(\alpha, p) = 0$ if and only if $\alpha_2 = p_{1,2} = 1 - p_{2,1}$. Asymptotic accurate (resp., exact) recovery seems thus possible as soon as $T = \omega(n)$ (resp., $T = \omega(n\ln(n))$) for almost any BMC with two clusters—the only exception are BMCs with parameters on this line. Note that if we did not have the information quantity at our disposal, it would be challenging to give a heuristic argument whether a specific BMC allows for asymptotic exact recovery. Consider for instance a BMC with $\alpha_1 = \alpha_2 = \frac{1}{2}$ and $p_{1,2} = 1 - p_{2,1} \neq \frac{1}{2}$ and $p_{1,2} > p_{2,1}$ w.l.o.g. In this scenario, $P_{x,z} = P_{y,z}$ for all $x, y, z \in \mathcal{V}$, that is, every row of the kernel is identical to any other row. Looking at this kernel, we would not expect to be able to cluster. However, here $\pi_2 > \pi_1$, and we could cluster based on the equilibrium distribution as $T \to \infty$. The information quantity takes the fact that we are dealing with a Markov chain appropriately into account, and correctly asserts for this case that asymptotic recovery is possible.

Figure 3 illustrates for which parameters one can possibly recover the two clusters asymptotically exactly when $T = n\ln n$. Specifically, it depicts all parameters $\alpha_2, p_{1,2}, p_{2,1} \in (0, 1)$ for which $I(\alpha, p) > 1$. If we fix α_2, note that when $p_{1,2}, p_{2,1} \downarrow 0$ (bottom left), the Markov chain tends to stay within the current cluster for a substantial time. Similarly, when $p_{1,2}, p_{2,1} \uparrow 1$ (top right), the Markov chain tends to jump into the other cluster every time. In both scenarios, the states are relatively easy to cluster. This draws parallels with the SBM. When either $p_{1,2} \downarrow 0$ (left) or $p_{2,1} \downarrow 0$ (bottom), clustering is again doable: in these scenarios, the Markov chain tends to stay in the cluster of the starting state—and the fact that you never see the other vertices suggests that they have other transitions rates and, therefore, belong to the other cluster,

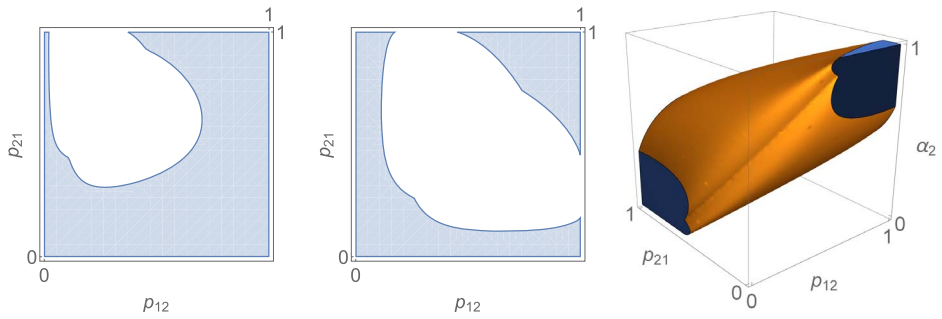


FIG. 3. (*left*, *middle*) *The parameters* $(p_{1,2}, p_{2,1})$ *in blue for which asymptotic exact recovery should be possible in the critical regime* $T = n \ln n$ *for* $K = 2$ *clusters*. (*right*) *The parameters* $(\alpha_2, p_{1,2}, p_{2,1})$ *for which asymptotic exact recovery is likely not possible, that is,* $I(\alpha, p) < 1$.

3.2. *Procedure for cluster recovery.* Theorem 1 has established necessary conditions for asymptotically accurate and exact recovery, and has identified performance limits satisfied by any $(\varepsilon, 1)$-locally good clustering algorithm at (α, p). In this section, we devise a $(\varepsilon, 1)$-locally good clustering algorithm at (α, p) that reaches these limits orderwise. The proposed algorithm proceeds in two steps: The first step performs a spectral decomposition of the random matrix $\hat{N}$ corresponding to the empirical transition rates between any pair of states, and defined by

$$\hat{N}_{x,y} \triangleq \sum_{t=0}^{T-1} \mathbb{1}[X_t = x, X_{t+1} = y] \quad \text{for } x, y \in \mathcal{V}.$$

The rank-K approximation of $\hat{N}$ is used to get initial estimates of the clusters. The second step sequentially improves the cluster estimates. In each iteration, the parameters of the BMCs are inferred from the previous cluster estimates, and states are reassigned to clusters based on these estimated parameters and the observed trajectory (by maximizing a log-likelihood).

3.2.1. *Spectral Clustering Algorithm.* The first step of our procedure is the Spectral Clustering Algorithm, presented in Algorithm 1. It leverages the spectral decomposition of $\hat{N}$ to estimate the clusters.

Before applying a singular value decomposition (SVD) to $\hat{N}$, we first need to trim the matrix so as to remove states that have been visited abnormally often. These states would namely perturb the spectral decomposition of $\hat{N}$. More precisely, we define the set Γ of states obtained from $\mathcal{V}$ by removing the $\lfloor n \exp(-(T/n) \ln(T/n)) \rfloor$ states with the highest numbers of visits in the observed sample path of length T. The spectral decomposition is applied to the matrix $\hat{N}_\Gamma$ obtained from $\hat{N}$ by setting all entries on the rows and columns corresponding to states not in Γ to zero.

The SVD of $\hat{N}_\Gamma$ is $U \Sigma V^{\mathrm{T}}$, from which we deduce $\hat{R}$ the best rank-K approximation of $\hat{N}_\Gamma$: $\hat{R} \triangleq \sum_{k=1}^{K} \sigma_k U_{\cdot,k} V_{\cdot,k}{}^{\mathrm{T}}$, where the values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$ denote the singular values of $\hat{N}_\Gamma$ in decreasing order. We apply a clustering algorithm to the rows and columns of $\hat{R}$ to determine the clusters. While in practice you may choose to use a different algorithm, for the analysis we use the following: first, we calculate the *neighborhoods*

$$\mathcal{N}_x \triangleq \left\{ y \in \mathcal{V} \middle| \sqrt{\|\hat{R}_{x,\cdot} - \hat{R}_{y,\cdot}\|_2^2 + \|\hat{R}_{\cdot,x} - \hat{R}_{\cdot,y}\|_2^2} \leq \frac{1}{n} \cdot \left(\frac{T}{n}\right)^{3/2} \left(\ln \frac{T}{n}\right)^{4/3} \right\}$$

Input: n, K and a trajectory $X_0, X_1, \ldots, X_T$
Output: an approximate cluster assignment $\hat{\mathcal{V}}_1^{[0]}, \ldots, \hat{\mathcal{V}}_K^{[0]}$, and matrix $\hat{N}$
begin
  for $x \leftarrow 1$ **to** n **do**
    for $y \leftarrow 1$ **to** n **do**
      $\hat{N}_{x,y} \leftarrow \sum_{t=0}^{T-1} \mathbb{1}[X_t = x, X_{t+1} = y]$;
    end
  end
  Calculate the trimmed matrix $\hat{N}_\Gamma$;
  Calculate the Singular Value Decomposition (SVD) $U \Sigma V^{\mathrm{T}}$ of $\hat{N}_\Gamma$;
  Order U, Σ, V s.t. the singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$ are in descending order;
  Construct the rank-K approximation $\hat{R} = \sum_{k=1}^{K} \sigma_k U_{\cdot,k} V_{\cdot,k}{}^{\mathrm{T}}$;
  Apply a K-means algorithm to $[\hat{R}, \hat{R}^\top]$ to determine $\hat{\mathcal{V}}_1^{[0]}, \ldots, \hat{\mathcal{V}}_K^{[0]}$;
end

Algorithm 1: Pseudo-code for the Spectral Clustering Algorithm

for $x \in \mathcal{V}$. Then we initialize $\hat{\mathcal{V}}_k \leftarrow \varnothing$ for $k = 1, \ldots, K$ and sequentially select K *centers* $z_1^*, \ldots, z_K^* \in \mathcal{V}$ from which we construct approximate clusters. Specifically, we iterate for $k = 1, \ldots K$:

$$\hat{\mathcal{V}}_k \leftarrow \mathcal{N}_{z_k^*} \setminus \left\{ \bigcup_{l=1}^{k-1} \hat{\mathcal{V}}_l \right\} \quad \text{where } z_k^* \triangleq \arg\max_{x \in \mathcal{V}} \left| \mathcal{N}_x \setminus \left\{ \bigcup_{l=1}^{k-1} \hat{\mathcal{V}}_l \right\} \right|.$$

Any remaining state is finally associated to the center closest to it, that is, we iterate for $y \in \{\bigcup_{k=1}^{K} \hat{\mathcal{V}}_k\}^{\mathrm{c}}$,

$$\hat{\mathcal{V}}_{k_y^*} \leftarrow \hat{\mathcal{V}}_{k_y^*} \cup \{y\} \quad \text{with } k_y^* \triangleq \arg\min_{k=1,\ldots,K} \sqrt{\|\hat{R}_{z_k^*,\cdot} - \hat{R}_{y,\cdot}\|_2^2 + \|\hat{R}_{\cdot,z_k^*} - \hat{R}_{\cdot,y}\|_2^2}.$$

Finally, the Spectral Clustering Algorithm outputs $\hat{\mathcal{V}}_k$ for $k = 1, \ldots, K$. Theorem 2 provides an upper bound on the number of misclassified states after executing the algorithm.

THEOREM 2. *Assume that $T = \omega(n)$ and $I(\alpha, p) > 0$. Then the proportion of misclassified states after the Spectral Clustering Algorithm satisfies*:

$$\frac{|\mathcal{E}|}{n} = O_{\mathbb{P}}\left(\frac{n}{T} \ln \frac{T}{n}\right) = o_{\mathbb{P}}(1).$$

From the above theorem, we conclude that the first step of our recovery procedure (i.e., the Spectral Clustering Algorithm) alone achieves an asymptotically accurate detection whenever this is at all possible, say when $I(\alpha, p) > 0$ and $T = \omega(n)$. However, it fails at ensuring asymptotic exact recovery, even in certain cases of $T = \omega(n \ln(n))$, and we cannot guarantee that its recovery rate approaches the fundamental limit identified in Theorem 1.

3.2.2. *Cluster Improvement Algorithm.* The second step of our clustering procedure, referred to as the Cluster Improvement Algorithm, aims at sequentially improving the cluster estimates obtained from the Spectral Clustering Algorithm until the recovery rate approaches

Input: An approximate assignment $\hat{\mathcal{V}}_1^{[t]}, \dots, \hat{\mathcal{V}}_K^{[t]}$, and matrix $\hat{N}$
Output: A revised assignment $\hat{\mathcal{V}}_1^{[t+1]}, \dots, \hat{\mathcal{V}}_K^{[t+1]}$
begin
  $n \leftarrow \dim(\hat{N})$, $\mathcal{V} \leftarrow \{1, \dots, n\}$, $T \leftarrow \sum_{x\in\mathcal{V}} \sum_{y\in\mathcal{V}} \hat{N}_{x,y}$;
  for $a \leftarrow 1$ **to** K **do**
    $\hat{\pi}_a \leftarrow \hat{N}_{\hat{\mathcal{V}}_a^{[t]},\mathcal{V}}/T$, $\hat{\alpha}_a \leftarrow |\hat{\mathcal{V}}_a^{[t]}|/n$, $\hat{\mathcal{V}}_a^{[t+1]} \leftarrow \varnothing$;
    for $b \leftarrow 1$ **to** K **do**
      $\hat{p}_{a,b} \leftarrow \hat{N}_{\hat{\mathcal{V}}_a^{[t]},\hat{\mathcal{V}}_b^{[t]}}/\hat{N}_{\hat{\mathcal{V}}_a^{[t]},\mathcal{V}}$;
    end
  end
  for $x \leftarrow 1$ **to** n **do**
    $c_x^{\text{opt}} \leftarrow \arg\max_{c=1,\dots,K}\{\sum_{k=1}^{K}(\hat{N}_{x,\hat{\mathcal{V}}_k^{[t]}} \ln \hat{p}_{c,k} + \hat{N}_{\hat{\mathcal{V}}_k^{[t]},x} \ln \frac{\hat{p}_{k,c}}{\hat{\alpha}_c}) - \frac{T}{n} \cdot \frac{\hat{\pi}_c}{\hat{\alpha}_c}\}$;
    $\hat{\mathcal{V}}_{c_x^{\text{opt}}}^{[t+1]} \leftarrow \hat{\mathcal{V}}_{c_x^{\text{opt}}}^{[t+1]} \cup \{x\}$;
  end
end

Algorithm 2: Pseudo-code for the Cluster Improvement Algorithm

the limits predicted in Theorem 1. The pseudo-code of the Cluster Improvement Algorithm is presented in Algorithm 2.

The Cluster Improvement Algorithm works as follows. Given a cluster assignment $\{\hat{\mathcal{V}}_k^{[t]}\}_{k=1,\dots,K}$ obtained after the tth iteration, it first calculates the estimates

$$\hat{p}_{a,b} = \hat{N}_{\hat{\mathcal{V}}_a^{[t]},\hat{\mathcal{V}}_b^{[t]}}/\hat{N}_{\hat{\mathcal{V}}_a^{[t]},\mathcal{V}} \quad \text{for } a, b = 1, \dots, K,$$

$$\hat{\pi}_k = \frac{1}{T} \sum_{x\in\hat{\mathcal{V}}_k^{[t]}} \sum_{y\in\mathcal{V}} \hat{N}_{x,y} \quad \text{and} \quad \hat{\alpha}_k = \frac{|\hat{\mathcal{V}}_k^{[t]}|}{n} \quad \text{for } k = 1, \dots, K.$$

It then initializes $\hat{\mathcal{V}}_k^{[t+1]} = \varnothing$ for $k = 1, \dots, K$, and assigns each state $x = 1, \dots, n$ to $\mathcal{V}_{c_x^{\text{opt}}}^{[t+1]} \leftarrow \mathcal{V}_{c_x^{\text{opt}}}^{[t+1]} \cup \{x\}$, where $c_x^{\text{opt}} \triangleq \arg\max_{c=1,\dots,K} u_x^{[t]}(c)$, and

$$u_x^{[t]}(c) \triangleq \left\{ \sum_{k=1}^{K} \left(\hat{N}_{x,\hat{\mathcal{V}}_k^{[t]}} \ln \hat{p}_{c,k} + \hat{N}_{\hat{\mathcal{V}}_k^{[t]},x} \ln \frac{\hat{p}_{k,c}}{\hat{\alpha}_c} \right) - \frac{T}{n} \cdot \frac{\hat{\pi}_c}{\hat{\alpha}_c} \right\}. \tag{4}$$

This results in a new cluster assignment $\{\hat{\mathcal{V}}_k^{[t+1]}\}_{k=1,\dots,K}$. Note that the algorithm works by placing each state in the cluster it most likely belongs to, based on the known structure and the sample path. This can be seen by noting that the objective function in (4) is the difference between two log-likelihood functions.

The second step of our clustering procedure applies the Cluster Improvement Algorithm several times, using as the initial input the cluster assignment $\{\hat{\mathcal{V}}_k^{[0]}\}_{k=1,\dots,K}$ obtained from the Spectral Clustering Algorithm. We denote by $\mathcal{E}^{[t]}$ the set of misclassified state after the tth iteration of the Clustering Improvement Algorithm. The overall performance of the clustering procedure is quantified in Theorem 3.

THEOREM 3. *Assume that $T = \omega(n)$ and $I(\alpha, p) > 0$. Then for any $t \geq 1$, after t-iterations of the Clustering Improvement Algorithm, initially applied to the output of the*

Spectral Clustering Algorithm, we have

$$\frac{|\mathcal{E}^{[t]}|}{n} = O_{\mathbb{P}}\big(\mathrm{e}^{-t(\ln\frac{T}{n}-\ln\ln\frac{T}{n})} + \mathrm{e}^{-\frac{\alpha_{\min}^2}{720\eta^3\alpha_{\max}^2}\frac{T}{n}I(\alpha,p)}\big).$$

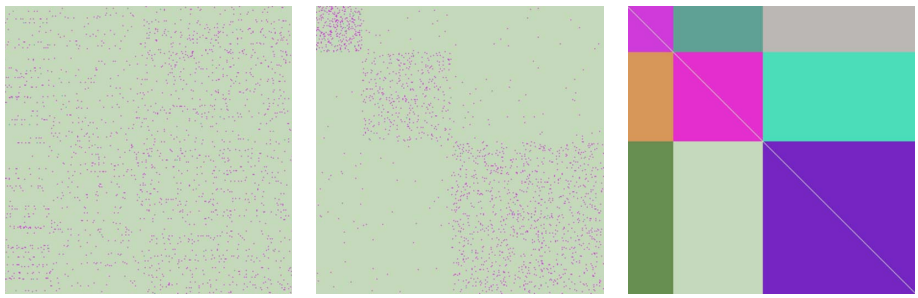
Observe that for $t = \ln n$, the number of misclassified states after applying t times the Clustering Improvement Algorithm is at most of the order $n\exp(-C(T/n)I(\alpha, p))$ with $C \triangleq \frac{\alpha_{\min}^2}{720\eta^3\alpha_{\max}^2}$. Up to the constant C, this corresponds to the fundamental recovery rate limit identified in Theorem 1. In particular, our clustering procedure achieves asymptotically exact detection under the following nearly tight sufficient condition: $I(\alpha, p) > 0$ and $T - \frac{n\ln n}{C\cdot I(\alpha,p)} = \omega(1)$.

4. Numerical experiments. In this section, we numerically assess the performance of our algorithms. We first investigate a simple illustrative example. Then we study the sensitivity of the error rate of the Spectral Clustering Algorithm w.r.t. the number of states and the length of the observed trajectory. Finally we show the performance of the Cluster Improvement Algorithm depending on the number of times it is applied to the output of the Spectral Clustering Algorithm.

4.1. *An example.* Consider $n = 300$ states grouped into three clusters of respective relative sizes $\alpha = (0.15, 0.35, 0.5)$, that is, the cluster sizes are cluster sizes $|\mathcal{V}_1| = 48$, $|\mathcal{V}_2| = 93$ and $|\mathcal{V}_3| = 159$. The transition rates between these clusters are defined by: $p = (0.9200, 0.0450, 0.0350; 0.0125, 0.8975, 0.0900; 0.0175, 0.0200, 0.9625)$.

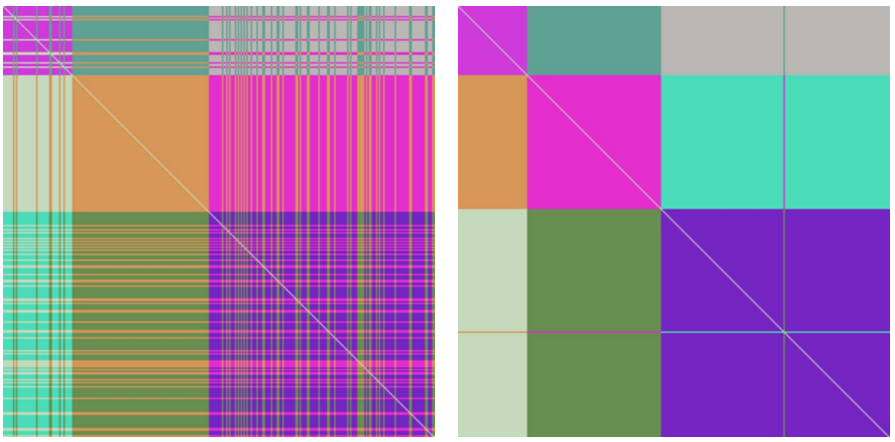
We generate a sample path of the Markov chain of length $T = n^{1.025}\ln n \approx 1973$ and calculate $\hat{N}$. A density plot of a typical sample of $\hat{N}$ is shown in Figure 4(a). The same density plot is presented in Figure 4(b) where the states have been sorted so as states in the same cluster are neighbors. It is important to note that the algorithms are of course not aware of the structure initially—sorting states constitutes their objective. Next, in Figure 4(c), we show a color representation of the kernel P with sorted rows and columns, in which we can clearly see the groups. Note that the specific colors have no meaning, except for the fact that within the same image two entries with the same color have the same numerical value.

Next, we apply the Spectral Clustering Algorithm. This generates an initial approximate clustering $\hat{\mathcal{V}}_1^{[0]}, \hat{\mathcal{V}}_2^{[0]}, \hat{\mathcal{V}}_3^{[0]}$ of the states. We generate a visual representation of this clustering by constructing $\hat{P}^{[0]}$ from the approximate cluster structure and the estimate $\hat{p}^{[0]}$. This represents the belief that the algorithm has at this point of the true BMC kernel P. A color



(a) $\hat{N}$, unsorted (b) $\hat{N}$, sorted (c) P, sorted

FIG. 4. *A sample path of length $T = n^{1.025}\ln n \approx 1973$ was generated, from which $\hat{N}$ is calculated. If we sort the states according to the clusters they belong to, we can see that states within the same cluster share similar dynamics.*



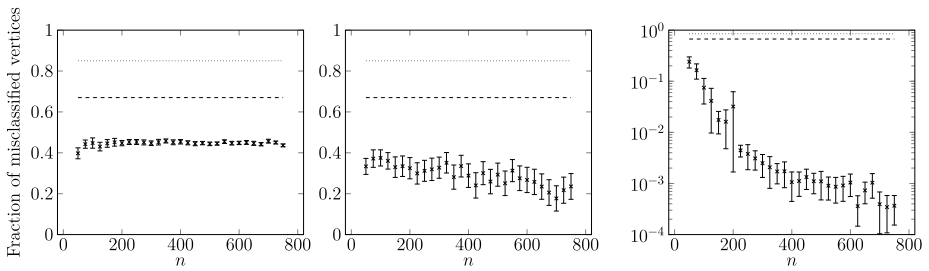
(a) Initial clustering. (b) Final clustering.

FIG. 5. (a) *Result after applying the Spectral Clustering Algorithm to the approximation* $\hat{N}$. (b) *Result after applying* 3 *iterations of the Cluster Improvement Algorithm.* 99.7% *of all states were accurately clustered.*

representation of this kernel is shown in Figure 5(a). We finally execute the Cluster Improvement Algorithm. After 3 iterations, it has settled on a final clustering. We generate a color representation of the clustering similar to before, resulting in Figure 5(b). The algorithms achieved a 99.7% accuracy: all but one state have been accurately clustered.

4.2. *Performance sensitivity of the Spectral Clustering Algorithm.* In this section, we examine the dependency of the number of misclassified states on the size of the kernel n, when we only apply the Spectral Clustering Algorithm. We choose $\alpha = (0.15, 0.35, 0.5)$, and set $p = (0.50, 0.20, 0.30; 0.10, 0.70, 0.20; 0.35, 0.05, 0.60)$. These parameters imply that $I(\alpha, p) \approx 0.88 > 0$. This value for $I(\alpha, p)$ is lower than in the case examined in Section 4.1, so we expect clustering to be more difficult. We have selected a more challenging model so that the initial number of misclassified states will be large and the asymptotics clear.

Figure 6 displays the error rate of the Spectral Clustering Algorithm as a function of n, for different trajectory lengths T. As benchmarks, we include a dashed line that indicates the error rate obtained by assigning states to clusters uniformly at random, that is, $\mathbb{P}[v \notin \mathcal{V}_{\sigma(v)}] = \sum_{k=1}^{K} \mathbb{P}[v \notin \mathcal{V}_k | \sigma(v) = k]\alpha_k = 1 - 1/K$, as well as a dotted line that indicates the error rate when assigning all states to the smallest cluster, that is, $1 - \min_k\{\alpha_k\}$. For the K-means step of the algorithms, we use Mathematica's default implementation for convenience. Observe that when $T = n \ln n$, the fraction of misclassified states hardly decrease as a function of n.



(a) $T = n \ln n$ (b) $T = n(\ln n)^{3/2}$ (c) $T = n(\ln n)^2$

FIG. 6. *The error rate of the Spectral Clustering Algorithm as function of* n, *for different scalings of* T. *Every point is the average result of* 40 *simulations, and the bars indicate a* 95%*-confidence interval.*

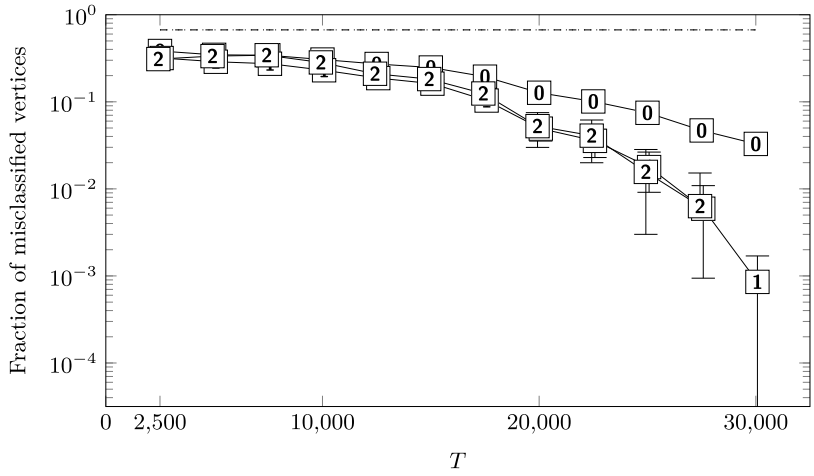


FIG. 7. *The error after applying the Spectral Clustering Algorithm* (*mark* 0), *and subsequently the Cluster Improvement Algorithm* (*marks* 1, 2) *several times*, *as a function of* T. *Each number represents the number of improvement steps*. *Here*, $n = 240$. *Every point is the average result of* 200 *simulations*, *and the bars indicate a* 95%*-confidence interval*. *We have minorly offset marks* 1, 2 *to the right and left for readability*, *respectively*. *At* $T = 30{,}000$, *the Cluster Improvement Algorithm achieved* 100% *accurate detection after* 2 *iterations in* all 200 *instances*.

This is in line with our lower bound. When T gets larger, the error converges to zero faster. Note that the Spectral Clustering Algorithm recovers the clusters exactly when the sample path is sufficiently long.

4.3. *Performance sensitivity of the Cluster Improvement Algorithm*. We now examine the number of misclassified states as a function of T, when we apply the Spectral Clustering Algorithm and a certain number of iterations of the Cluster Improvement Algorithm. We choose $\alpha = (1/3, 1/3, 1/3)$, and set $p = (0.1, 0.4, 0.5; 0.7, 0.1, 0.2; 0.6, 0.3, 0.1)$. Different from the previous experiments, the clusters are now of equal size and the off-diagonal entries of p are dominant. These parameters imply that $I(\alpha, p) \approx 0.27 > 0$, so the cluster algorithms should work, but the situation is again more challenging than in Section 4.1 and Section 4.2.

Figure 7 depicts the error after applying the Spectral Clustering Algorithm and subsequently the Cluster Improvement Algorithm up to two times, as a function of T. We have chosen both n, T relatively small so that the inputs are significantly noisy. For short sample paths, $T \lesssim 15{,}000$, the data is so noisy that the Cluster Improvement Algorithm does not provide any improvement over the Spectral Clustering Algorithm. For $T \gtrsim 15{,}000$, the Spectral Clustering Algorithm provides a sufficiently accurate initial clustering for the Cluster Improvement Algorithm to work. Because marks 1 and 2 overlap in almost all cases, we can conclude that there is (on average, and in the present situation) no benefit in running the Clustering Improvement Algorithm more than once. There is no mark 2 at $T = 30{,}000$ in this logarithmic plot, because the Cluster Improvement Algorithm achieved 100% accurate detection after 2 iterations in *all* 200 simulations.

4.4. *Critical regime where* $T = n\ln(n)$. We now study how well our clustering procedure performs in the critical regime $T = n\ln n$. Here, we will consider $K = 2$ clusters of equal size: $\alpha_1 = \alpha_2 = \frac{1}{2}$. Recall that every such BMC can then be completely parameterized by $(p_{1,2}, p_{2,1}) \in (0, 1)^2$. Our goal in this section is to numerically evaluate $\hat{\mathcal{F}}_1(\varepsilon) = \{(p_{1,2}, p_{2,1}) \in (0, 1)^2 | \mathbb{E}_P[|\mathcal{E}^{[t]}|/n] \geq \varepsilon\}$ as a proxy estimate to the region $\{(p_{1,2}, p_{2,1}) | I(\alpha, p) \leq 1\}$. We rasterized $(0, 1)^2$ and ran our clustering procedure for $n = 300$

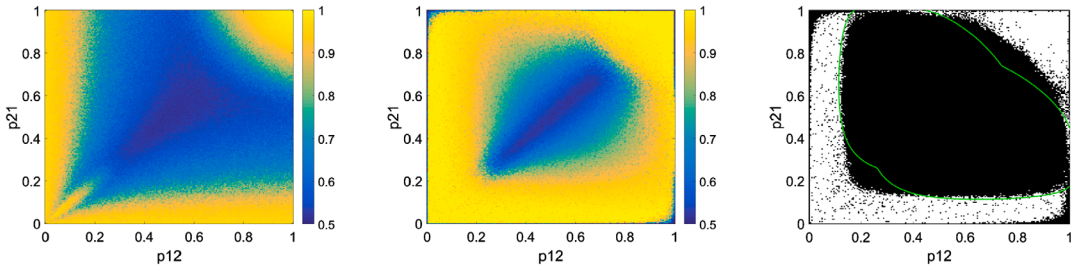


FIG. 8. *The average proportion of well-classified states for each rasterpoint* $(p_{1,2}, p_{2,1}) \in (0,1)^2$ *after the Spectral Clustering Algorithm* (*left*) *and Cluster Improvement Algorithm* (*middle*), *and numerical feasibility region of our clustering procedure* (*right*), *both in the critical regime* $T = n \ln n$. *The line outlines the theoretical region* $I(\alpha, p) \leq 1$ *within which no algorithm exists able to asymptotically recover the clusters exactly.*

with $t = 6$ improvement steps for each parameter pair $(p_{1,2}, p_{2,1})$. The results are shown in Figure 8. Note that the sample mean at each rasterpoint was calculated from 10 independent runs.

5. Lower bounds via change-of-measure arguments. In this section, we prove Theorem 1 using change-of-measure arguments such as those frequently used to identify information-theoretical limits in online stochastic optimization [17]. We sketch the proof below, and provide a detailed description of its steps later in this section. We distinguish two cases:

(i) If $I(\alpha, p) = 0$, we prove that there are two clusters whose states cannot be distinguished from any set of observations. Hence, the number of misclassified states grows linearly with n, which corresponds to the statement of the theorem when $I(\alpha, p) = 0$.

(ii) If $I(\alpha, p) > 0$, we use a change-of-measure argument. We denote by Φ the true stochastic model under which the observations $X_0, \ldots, X_T$ are generated, and by $\mathbb{P}_\Phi$ (resp., $\mathbb{E}_\Phi$) the corresponding probability measure (resp., expectation). The true model is characterized by the transition matrix P. In a change-of-measure argument, we pretend that the observations are instead generated under a slightly different stochastic model Ψ (denote by $\mathbb{P}_\Psi$ and $\mathbb{E}_\Psi$ the corresponding probability measure and expectation). The modified stochastic model Ψ is constructed as follows. We pick a state V^* randomly from clusters $\mathcal{V}_a$ and $\mathcal{V}_b$ with $a \neq b$, and place this state in its own cluster, indexed by 0. We further construct a transition matrix Q depending on V^* and slightly different than P. Ψ is hence characterized by both V^* and the transition matrix Q. Next, we introduce the log-likelihood ratio of the observations $X_0, \ldots, X_T$ under the transition matrices P and Q:

$$L \triangleq \ln \frac{\mathbb{P}_Q[X_0, X_1, \ldots, X_T]}{\mathbb{P}_P[X_0, X_1, \ldots, X_T]},$$

where

$$\mathbb{P}_P[X_0, X_1, \ldots, X_T] = \prod_{t=1}^{T} P_{X_{t-1}, X_t} \quad \text{and} \quad \mathbb{P}_Q[X_0, X_1, \ldots, X_T] = \prod_{t=1}^{T} Q_{X_{t-1}, X_t}.$$

Note that L is random because it depends on the observations, but also on V^*. We then establish that there exist constants $\delta, C > 0$ (independent of n) such that under any clustering algorithm the expected number of misclassified states satisfies:

$$\mathbb{E}_\Phi[|\mathcal{E}|] \geq Cn \exp\left(-\mathbb{E}_\Psi[L] - \sqrt{\frac{2}{\delta}} \sqrt{\mathrm{Var}_\Psi[L]}\right). \tag{5}$$

Finally, for particular choices of a, b and the transition matrix Q, the analysis of $\mathbb{E}_\Psi[L]$ and $\mathrm{Var}_\Psi[L]$ reveals that: for $T = \omega(n)$, $\mathbb{E}_\Psi[L] \le (T/n)I(\alpha, p) + o(T/n)$ and $\mathrm{Var}_\Psi[L] = o(T^2/n^2)$. Combining this with (5) completes the proof of Theorem 1.

5.1. *Necessary and sufficient condition for $I(\alpha, p) = 0$.*

LEMMA 1. *For any BMC, $I(\alpha, p) = 0$ if and only if there exists $i \neq j$ such that $p_{i,c} = p_{j,c}$ and $p_{c,i}/\alpha_i = p_{c,j}/\alpha_j$ for all $c \in \{1, \dots, K\}$.*

Lemma 1 implies that observing a trajectory of the Markov chain does not provide any information allowing us to distinguish between states in $\mathcal{V}_i$ and $\mathcal{V}_j$. As a consequence, any clustering algorithm will misclassify at least a fixed proportion of states. The proof of Lemma 1 is provided in Section SM3 of the Supplementary Material [25].

5.2. *Change-of-measure argument.* In the remainder of this section, we assume that $I(\alpha, p) > 0$. The argument consists in considering that the observations $X_0, \dots, X_T$ are generated by a slightly different stochastic model than the true model defined by the clusters and the transition matrix P. As mentioned above, the modified model is obtained by randomly choosing a state V^* from two clusters $\mathcal{V}_a$ and $\mathcal{V}_b$, and by constructing a transition matrix Q depending on V^* that is slightly different from P. Under the new model, V^* is placed in its own cluster indexed by 0. The matrix Q is parametrized by q, a $2(K+1)$-dimensional vector in

$$\mathcal{Q} \triangleq \left\{ (q_{k,0}, q_{0,k})_{k=0,\dots,K} \in (0,\infty) \,\middle|\, q_{0,0} = 0, \sum_{l=1}^{K} q_{0,l} = 1 \right\},$$

where $q_{k,0}$ (resp., $q_{0,k}$) will define the probability to move from a state in cluster $\mathcal{V}_k$ to V^* (resp., from state V^* to a state in $\mathcal{V}_k$) under the new model Ψ. We also define

$$q_{k,l} \triangleq p_{k,l} - \frac{q_{k,0}}{Kn} \quad \text{for } k, l = 1, \dots, K, \tag{6}$$

and assume that $n > \lceil \max_{k,l=1,\dots,K}\{q_{k,0}/(Kp_{k,l})\} \rceil$ so that the entries of (6) are strictly positive. Note that the collection $\{q_{k,l}\}_{k,l\in\{0,1,\dots,K\}}$ does *not* constitute a stochastic matrix, but does resemble the transition matrix p for sufficiently large n. We are now ready to define Q componentwise:

$$Q_{x,y} \triangleq \frac{q_{\omega(x),\omega(y)} \mathbb{1}[x \neq y]}{|\mathcal{W}_{\omega(y)}| - \mathbb{1}[\omega(x) = \omega(y)]}, \qquad Q_{x,V^*} \triangleq \frac{q_{\omega(x),0}}{n} \quad \text{for } x \in \mathcal{V}, y \neq V^*, \tag{7}$$

where

$$\omega(x) \triangleq \begin{cases} 0 & \text{if } x = V^*, \\ \sigma(x) & \text{if } x \neq V^*, \end{cases} \quad \text{and} \quad \mathcal{W}_k \triangleq \begin{cases} \{V^*\} & \text{if } k = 0, \\ \mathcal{V}_k \setminus \{V^*\} & \text{if } k = 1, \dots, K, \end{cases}$$

for notational convenience. This has the added benefit of giving (7) a similar form as (1).

Q is by construction a stochastic matrix (see Section SM3.1 of the Supplementary Material [25]). Note furthermore that because Q is constructed from P, which by assumption describes an irreducible Markov chain, and because the entries $\{q_{k,0}, q_{0,k}\}_{k=1,\dots,K}$ are all strictly positive, Q also describes an irreducible Markov chain. Next, we highlight other basic properties of this chain, and provide an example of matrix Q.

Equilibrium distribution. Let $\Pi^{(Q)}$ denote the equilibrium distribution of a Markov chain with transition matrix Q, that is, the solution to $\Pi^{(Q)\mathrm{T}} Q = \Pi^{(Q)\mathrm{T}}$. By symmetry of states in the same cluster $\Pi_x^{(Q)} = \Pi_y^{(Q)} \triangleq \bar{\Pi}_k^{(Q)}$ for any two states $x, y \in \mathcal{W}_k$ and all $k \in \{0, 1, \dots, K\}$.

Define

$$\gamma_k^{[0]} \triangleq \lim_{n\to\infty} \sum_{x\in\mathcal{W}_k} \Pi_x^{(Q)} = \lim_{n\to\infty} |\mathcal{W}_k| \bar{\Pi}_k^{(Q)} \quad \text{for } k \in \{0, 1, \dots, K\}.$$

We can expect $\gamma_0^{[0]}$ to be zero, because by our construction of Q we can expect that $\Pi_x^{(Q)} = O(1/n)$ for all $x \in \mathcal{V}$ (including V^*). We therefore also define its higher order statistic $\gamma_0^{[1]} \triangleq \lim_{n\to\infty} n\Pi_{V^*}^{(Q)}$. The following proposition relates these scaled quantities to the parameters of our BMC $\{X_t\}_{t\geq 0}$. The proof is deferred to Section SM3.2 in the Supplementary Material [25], and relies on several applications of the balance equations and a subsequent asymptotic analysis.

PROPOSITION 3. *For* $k = 1, \dots, K$, $\gamma_k^{[0]} = \pi_k$. *Furthermore,* $\gamma_0^{[0]} = 0$ *and* $\gamma_0^{[1]} = \sum_{k=1}^K \pi_k q_{k,0}$.

Mixing time. It is important to note that Proposition 2 holds for a Markov chain with Q as its transition matrix as well. This follows when applying the exact same proof.

Example. It is illustrative to explicitly write down at least one example kernel Q. For $K = 3$, $\alpha = (2/10, 3/10, 5/10)$ and $n = 10$, $V^* = 7$, it is given by

$$Q = \begin{pmatrix}
0 & p_{1,1} & \frac{p_{1,2}}{3} & \frac{p_{1,2}}{3} & \frac{p_{1,2}}{3} & \frac{p_{1,3}}{4} & \frac{q_{1,0}}{10} & \frac{p_{1,3}}{4} & \frac{p_{1,3}}{4} & \frac{p_{1,3}}{4} \\
p_{1,1} & 0 & \frac{p_{1,2}}{3} & \frac{p_{1,2}}{3} & \frac{p_{1,2}}{3} & \frac{p_{1,3}}{4} & \frac{q_{1,0}}{10} & \frac{p_{1,3}}{4} & \frac{p_{1,3}}{4} & \frac{p_{1,3}}{4} \\
\hdashline
\frac{p_{2,1}}{2} & \frac{p_{2,1}}{2} & 0 & \frac{p_{2,2}}{2} & \frac{p_{2,2}}{2} & \frac{p_{2,3}}{4} & \frac{q_{2,0}}{10} & \frac{p_{2,3}}{4} & \frac{p_{2,3}}{4} & \frac{p_{2,3}}{4} \\
\frac{p_{2,1}}{2} & \frac{p_{2,1}}{2} & \frac{p_{2,2}}{2} & 0 & \frac{p_{2,2}}{2} & \frac{p_{2,3}}{4} & \frac{q_{2,0}}{10} & \frac{p_{2,3}}{4} & \frac{p_{2,3}}{4} & \frac{p_{2,3}}{4} \\
\frac{p_{2,1}}{2} & \frac{p_{2,1}}{2} & \frac{p_{2,2}}{2} & \frac{p_{2,2}}{2} & 0 & \frac{p_{2,3}}{4} & \frac{q_{2,0}}{10} & \frac{p_{2,3}}{4} & \frac{p_{2,3}}{4} & \frac{p_{2,3}}{4} \\
\hdashline
\frac{p_{3,1}}{2} & \frac{p_{3,1}}{2} & \frac{p_{3,2}}{3} & \frac{p_{3,2}}{3} & \frac{p_{3,2}}{3} & 0 & \frac{q_{3,0}}{10} & \frac{p_{3,3}}{3} & \frac{p_{3,3}}{3} & \frac{p_{3,3}}{3} \\
\frac{q_{0,1}}{2} & \frac{q_{0,1}}{2} & \frac{q_{0,2}}{3} & \frac{q_{0,2}}{3} & \frac{q_{0,2}}{3} & \frac{q_{0,3}}{4} & 0 & \frac{q_{0,3}}{4} & \frac{q_{0,3}}{4} & \frac{q_{0,3}}{4} \\
\frac{p_{3,1}}{2} & \frac{p_{3,1}}{2} & \frac{p_{3,2}}{3} & \frac{p_{3,2}}{3} & \frac{p_{3,2}}{3} & \frac{p_{3,3}}{3} & \frac{q_{3,0}}{10} & 0 & \frac{p_{3,3}}{3} & \frac{p_{3,3}}{3} \\
\frac{p_{3,1}}{2} & \frac{p_{3,1}}{2} & \frac{p_{3,2}}{3} & \frac{p_{3,2}}{3} & \frac{p_{3,2}}{3} & \frac{p_{3,3}}{3} & \frac{q_{3,0}}{10} & \frac{p_{3,3}}{3} & 0 & \frac{p_{3,3}}{3} \\
\frac{p_{3,1}}{2} & \frac{p_{3,1}}{2} & \frac{p_{3,2}}{3} & \frac{p_{3,2}}{3} & \frac{p_{3,2}}{3} & \frac{p_{3,3}}{3} & \frac{q_{3,0}}{10} & \frac{p_{3,3}}{3} & \frac{p_{3,3}}{3} & 0
\end{pmatrix} \tag{8}$$

$$- \frac{1}{3\cdot 10} \begin{pmatrix}
0 & q_{1,0} & \frac{q_{1,0}}{3} & \frac{q_{1,0}}{3} & \frac{q_{1,0}}{3} & \frac{q_{1,0}}{4} & 0 & \frac{q_{1,0}}{4} & \frac{q_{1,0}}{4} & \frac{q_{1,0}}{4} \\
q_{1,0} & 0 & \frac{q_{1,0}}{3} & \frac{q_{1,0}}{3} & \frac{q_{1,0}}{3} & \frac{q_{1,0}}{4} & 0 & \frac{q_{1,0}}{4} & \frac{q_{1,0}}{4} & \frac{q_{1,0}}{4} \\
\hdashline
\frac{q_{2,0}}{2} & \frac{q_{2,0}}{2} & 0 & \frac{q_{2,0}}{2} & \frac{q_{2,0}}{2} & \frac{q_{2,0}}{4} & 0 & \frac{q_{2,0}}{4} & \frac{q_{2,0}}{4} & \frac{q_{2,0}}{4} \\
\frac{q_{2,0}}{2} & \frac{q_{2,0}}{2} & \frac{q_{2,0}}{2} & 0 & \frac{q_{2,0}}{2} & \frac{q_{2,0}}{4} & 0 & \frac{q_{2,0}}{4} & \frac{q_{2,0}}{4} & \frac{q_{2,0}}{4} \\
\frac{q_{2,0}}{2} & \frac{q_{2,0}}{2} & \frac{q_{2,0}}{2} & \frac{q_{2,0}}{2} & 0 & \frac{q_{2,0}}{4} & 0 & \frac{q_{2,0}}{4} & \frac{q_{2,0}}{4} & \frac{q_{2,0}}{4} \\
\hdashline
\frac{q_{3,0}}{2} & \frac{q_{3,0}}{2} & \frac{q_{3,0}}{3} & \frac{q_{3,0}}{3} & \frac{q_{3,0}}{3} & 0 & 0 & \frac{q_{3,0}}{3} & \frac{q_{3,0}}{3} & \frac{q_{3,0}}{3} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{q_{3,0}}{2} & \frac{q_{3,0}}{2} & \frac{q_{3,0}}{3} & \frac{q_{3,0}}{3} & \frac{q_{3,0}}{3} & \frac{q_{3,0}}{3} & 0 & 0 & \frac{q_{3,0}}{3} & \frac{q_{3,0}}{3} \\
\frac{q_{3,0}}{2} & \frac{q_{3,0}}{2} & \frac{q_{3,0}}{3} & \frac{q_{3,0}}{3} & \frac{q_{3,0}}{3} & \frac{q_{3,0}}{3} & 0 & \frac{q_{3,0}}{3} & 0 & \frac{q_{3,0}}{3} \\
\frac{q_{3,0}}{2} & \frac{q_{3,0}}{2} & \frac{q_{3,0}}{3} & \frac{q_{3,0}}{3} & \frac{q_{3,0}}{3} & \frac{q_{3,0}}{3} & 0 & \frac{q_{3,0}}{3} & \frac{q_{3,0}}{3} & 0
\end{pmatrix}.$$

Here, we have indicated the original cluster structure in dashed lines, and we have colored the row and column corresponding to the modified cluster behavior of state V^*. Comparing (8) to (2) helps understanding how Q is constructed and how Q compares to P. Note in particular the minor changes to the normalizations of all entries.

5.3. *Log-likelihood ratio and its connection to the error rate.* To describe how the log-likelihood ratio L relates to the error rate, we first introduce the following notation. For any $q \in \mathcal{Q}$, and any cluster index a, we define

$$I_a(q||p) \triangleq \sum_{k=1}^{K}\left(\left(\sum_{l=1}^{K}\pi_l q_{l,0}\right)q_{0,k}\ln\frac{q_{0,k}}{p_{a,k}} + \pi_k q_{k,0}\ln\frac{q_{k,0}\alpha_a}{p_{k,a}}\right) + \left(\frac{\pi_a}{\alpha_a} - \sum_{k=1}^{K}\pi_k q_{k,0}\right).$$

As shown later in this section, $I_a(q||p)$ is the leading order of the expected log-likelihood ratio L under Ψ and given that V^* is extracted from $\mathcal{V}_a$. We further define for any pair of cluster indexes a, b: $\mathcal{Q}(a,b) \triangleq \{q \in \mathcal{Q} | I_a(q||p) = I_b(q||p)\}$. These sets are not empty as stated below.

LEMMA 2. *For any two cluster indexes $a \neq b$, $\mathcal{Q}(a,b) \neq \varnothing$.*

Next, Proposition 4 states that choosing $q \in \mathcal{Q}(a,b)$, one may obtain lower bounds on the error rate by analyzing the log-likelihood ratio L.

PROPOSITION 4. *Assume that V^* is chosen uniformly at random from two different clusters $\mathcal{V}_a$ and $\mathcal{V}_b$, that Q is constructed from $q \in \mathcal{Q}(a,b)$, and that there exists a $(\varepsilon, 1)$-locally good clustering algorithm at (α, p). Then*:

(i) *There exists a constant $\delta > 0$ independent of n s.t. $\mathbb{P}_\Psi[V^* \in \mathcal{E}] \geq \delta > 0$.*
(ii) *There exists a constant $C > 0$ independent of n such that*

$$\mathbb{E}_\Phi[|\mathcal{E}|] \geq Cn\exp\left(-\mathbb{E}_\Psi[L] - \sqrt{\frac{2}{\delta}}\sqrt{\mathrm{Var}_\Psi[L]}\right). \tag{9}$$

5.4. *Analysis of the log-likelihood ratio.* Proposition 5 characterizes the leading term of $\mathbb{E}_\Psi[L]$ when the cluster from which V^* originates is fixed.

PROPOSITION 5. *For any given cluster index a, and $q \in \mathcal{Q}$, $\mathbb{E}_\Psi[L|\sigma(V^*) = a] = (T/n)I_a(q||p) + o(T/n)$.*

With Proposition 6, we establish that $\sqrt{\mathrm{Var}_\Psi[L|\sigma(V^*) = a]}$ is asymptotically negligible when compared to $\mathbb{E}_\Psi[L|\sigma(V^*)]$ whenever $T = \omega(n)$. The proof relies on relating the covariances *between* the T transitions in the sample path $\{X_0, X_1, X_2, \ldots, X_T\}$ to the mixing time of the underlying Markov chain. Proposition 2 then allows us to bound variance.

PROPOSITION 6. *For any given cluster index a, and $q \in \mathcal{Q}$, if $T = \omega(1)$, then $\mathrm{Var}_\Psi[L|\sigma(V^*) = a] = O(T/n)$. As a consequence if $T = \omega(n)$, then $\mathrm{Var}_\Psi[L|\sigma(V^*) = a] = o(T^2/n^2)$.*

Combining Propositions 4, 5 and 6, we obtain Corollary 1.

COROLLARY 1. *There exists a constant $C > 0$ such that for any two different cluster indexes a and b, and for any $q \in \mathcal{Q}(a,b)$, if $T = \omega(n)$, then $\mathbb{E}_\Psi[|\mathcal{E}|] \geq Cn\exp(-(T/n) \times I_a(q||p) + o(T/n))$.*

By varying a, b, and $q \in \mathcal{Q}(a,b)$ in Corollary 1, we conclude that if $T = \omega(n)$: $\mathbb{E}_\Phi[|\mathcal{E}|] \geq Cn\exp(-(T/n)J(\alpha, p) + o(T/n))$, where $J(\alpha, p) \triangleq \min_{a\neq b}\min_{q\in\mathcal{Q}(a,b)} I_a(q||p)$.

5.5. *Connecting $I(\alpha, p)$ and $J(\alpha, p)$.* Finally, to complete the proof of Theorem 1, we relate $I(\alpha, p)$ to $J(\alpha, p)$ by proving the following.

LEMMA 3. *For any BMC, we have $J(\alpha, p) \leq I(\alpha, p)$.*

6. Performance of the Spectral Clustering Algorithm. This section is devoted to the proof of Theorem 2. The Spectral Clustering Algorithm relies on a spectral decomposition of the trimmed matrix $\hat{N}_\Gamma$, gathering the empirical transition rates between states. The proof of Theorem 2 hence leverages concentration inequalities for Markov chains, provided in Section SM1 of the Supplementary Material [25], and a spectral analysis of $\hat{N}$ or $\hat{N}_\Gamma$.

6.1. *Spectral analysis.* The main ingredient in the proof of Theorem 2 is a concentration bound on the spectral norm of the matrix $\hat{N}_\Gamma$ centered around its mean, or more precisely on $\|\hat{N}_\Gamma - N\|$. The tighter such a bound is, the tighter our performance analysis of the Spectral Clustering Algorithm will be. Note that the concentration of the spectral norm holds for the trimmed matrix, that is, a matrix based on $\hat{N}$ in which the rows and columns that correspond to states that are visited too often are set to 0.

PROPOSITION 7. *For any BMC, $\|\hat{N}_\Gamma - N\| = O_{\mathbb{P}}(\sqrt{\frac{T}{n} \ln \frac{T}{n}})$.*

The above concentration result, proved in Section SM4.1 of the Supplementary Material [25], is sufficiently tight for the purposes of this paper, but can be improved up to logarithmic terms. The primary challenge one encounters in establishing this bound is that $\hat{N}$ is a random matrix with stochastically *dependent* entries, as explained in the Introduction. The concentration of the entire spectrum of $\hat{N}_\Gamma$ would be an intriguing topic for future study.

6.2. *Proof of Theorem 2.* Throughout the proof, we use the following notation: $N_{x,y} \triangleq \mathbb{E}_\Phi[\hat{N}_{x,y}] = T \Pi_x P_{x,y}$ for $x, y \in \mathcal{V}$, $\hat{R}^0 = [\hat{R}, \hat{R}^\top]$, $N^0 = [N, N^\top]$, and $\hat{N}^0 = [\hat{N}_\Gamma, \hat{N}_\Gamma^\top]$. We further introduce the quantity

$$D(\alpha, p) \triangleq \min_{a,b:a\neq b} \sum_{k=1}^{K} \left(\left(\frac{\pi_a p_{a,k}}{\alpha_k \alpha_a} - \frac{\pi_b p_{b,k}}{\alpha_k \alpha_b} \right)^2 + \left(\frac{\pi_k p_{k,a}}{\alpha_k \alpha_a} - \frac{\pi_k p_{k,b}}{\alpha_k \alpha_b} \right)^2 \right).$$

Note that $D(\alpha, p) = 0$ if and only if there exist a, b such that (C1) $p_{a,k} = p_{b,k}$ for all k, (C2) $p_{k,a}/\alpha_a = p_{k,b}/\alpha_b$ for all k, and (C3) $\pi_a/\alpha_a = \pi_b/\alpha_b$. Under (C1)–(C3), $I(\alpha, p) = 0$. Thus, $D(\alpha, p) > 0$ when $I(\alpha, p) > 0$.

The proof of Theorem 2 consists of four steps.

Step 1. We show that N^0 satisfies a *separability property*: that is, if two states $x, y \in \mathcal{V}$ do not belong to the same cluster, the l_2-distance between their respective rows $N^0_{x,\cdot}$, $N^0_{y,\cdot}$ is at least $\Omega(T\sqrt{D(\alpha, p)}/n^{3/2})$.

Step 2. We upper bound the error $\|\hat{R}^0 - N^0\|_{\mathrm{F}}$ using $\|\hat{N}_\Gamma - N\|$.

Step 3. We prove that if $\|\hat{N}_\Gamma - N\|$ is small enough as n grows large, then for any state x misclassified under the Spectral Clustering Algorithm, $\|\hat{R}^0_{x,\cdot} - N^0_{x,\cdot}\|_2$ is $\Omega_{\mathbb{P}}(T\sqrt{D(\alpha, p)}/n^{3/2})$. In other words, $\hat{R}^0$ inherits the separability property.

Step 4. Proposition 7 ensures that the statement derived in Step 3 holds. From there and using the result of Step 2, we conclude that the number of misclassified states satisfies Theorem 2.

Step 1. This is formalized in Lemma 4, and proven in Section SM4.2 of the Supplementary Material [25]. It is a consequence of the block structure of matrix N.

LEMMA 4. *For $x, y \in \mathcal{V}$ s.t. $\sigma(x) \neq \sigma(y)$, $\|N^0_{x,\cdot} - N^0_{y,\cdot}\|_2 = \Omega(\frac{T\sqrt{D(\alpha,p)}}{n^{3/2}})$.*

Step 2. Lemma 5 shows that the error $\|\hat{R}^0 - N^0\|_{\mathrm{F}}$ is asymptotically bounded by $\|\hat{N}_\Gamma - N\|$, and is proven in Section SM4.3 of the Supplementary Material [25]. The proof relies on a powerful bound relating decompositions of random matrices and their spectra [11].

LEMMA 5. $\|\hat{R}^0 - N^0\|_{\mathrm{F}} \leq \sqrt{16K}\|\hat{N}_\Gamma - N\|$.

Step 3. The result obtained in this step is stated in Lemma 6. Its proof, presented in Section SM4.4 of the Supplementary Material [25], requires a bound on the spectral concentration rate of $\hat{N}_\Gamma$'s noise matrix, relies on Lemmas 4, 5 and exploits the design of the K-means algorithm used after the spectral decomposition.

LEMMA 6. *If $\|\hat{N}_\Gamma - N\| = o_{\mathbb{P}}(f_n)$ for some sequence $f_n = o(T/n)$ and there exists a sequence h_n such that $\omega(f_n/\sqrt{n}) = h_n = o(T\sqrt{D(\alpha,p)}/n^{3/2})$, then*

$$\|\hat{R}^0_{x,\cdot} - N^0_{x,\cdot}\|_2 = \Omega_{\mathbb{P}}\left(\frac{T D^{1/2}(\alpha,p)}{n^{3/2}}\right) \quad \textit{for any misclassified state } x \in \mathcal{E}.$$

Step 4. In view of Proposition 7, the conditions of Lemma 6 are satisfied, for example, $f_n = ((T/n)\ln(T/n))^{1/2+\zeta}$ and $h_n = (f_n/\sqrt{n})(T/n)^\zeta$ with $0 < \zeta < 1/4$. The final step is then almost immediate. Indeed, we have because of Lemma 6 $\|\hat{R}^0 - N^0\|^2_{\mathrm{F}} \geq \sum_{x\in\mathcal{E}} \|\hat{R}^0_{x,\cdot} - N^0_{x,\cdot}\|^2_2 = |\mathcal{E}|\Omega_{\mathbb{P}}(T^2 D(\alpha,p)/n^3)$. Hence, using Lemma 5, $(1/|\mathcal{E}|)\|\hat{N}_\Gamma - N\|^2 = \Omega_{\mathbb{P}}(T^2 D(\alpha,p)/n^3)$, or equivalently, since $I(\alpha,p) > 0$, and hence $D(\alpha,p) > 0$, $(n/|\mathcal{E}|)\|\hat{N}_\Gamma - N\|^2 = \Omega_{\mathbb{P}}(T^2/n^2)$. We also have from Proposition 7 that $\|\hat{N}_\Gamma - N\|^2 = O_{\mathbb{P}}(\frac{T}{n}\ln\frac{T}{n})$. Combining the two previous equalities, we conclude that: $|\mathcal{E}|/n = O_{\mathbb{P}}(\frac{n}{T}\ln\frac{T}{n})$. This is obtained by applying Lemma 22 presented in Section SM6.5 of the Supplementary Material [25] (with $X_n = \|\hat{N}_\Gamma - N\|^2$, $Y_n = |\mathcal{E}|/n$, $y_n = T^2/n^2$, and $x_n = (T/n)\ln(T/n)$).

7. Performance of the Cluster Improvement Algorithm.

7.1. *Intuition behind the algorithm.* Let us first briefly explain the intuition behind the Cluster Improvement Algorithm: given a cluster assignment, the algorithm inspects for each state the cluster assignment that makes the observed sample path the most likely. It then assigns the state accordingly.

Note that such a greedy assignment based on maximizing a likelihood function is generally not guaranteed to converge to a good minimum. Indeed; first, if the initial cluster assignment is far from the true assignment, the approximated BMC parameters $\hat{p}$, $\hat{\pi}$ and $\hat{\alpha}$ will also be far from the true parameters. The cluster improvement algorithm will then weigh the sample paths according to an *incorrect model* and may assign states to incorrect clusters. Second, the induced *dependencies* between consecutive updates of our improvement algorithm threaten our chances of deriving a performance upper bound that gets tighter with the number of iterations.

7.2. *Proof of Theorem 3.* Our proof tackles both concerns—the concerns of assigning vertices according to an incorrect model, and of having strong dependencies between consecutive updates—by considering the precise asymptotic concentration rates of the BMC. Tying the first concern to concentration is straightforward: as long as the initial cluster assignment provides sufficiently good estimates of the BMC parameters, the sample path will be weighted according to a model that is close to the ground truth. The method with which we overcome the second concern is more refined. Specifically, in the proof of Theorem 3, we split the set of all states into a set of *well-behaved* states $\mathcal{H}$ and a set of *forlorn* states $\mathcal{H}^{\mathrm{c}}$.

DEFINITION. The set of well-behaved states $\mathcal{H}$ is the largest set of states $x \in \Gamma$ that satisfy the following two properties:

(H1) When $x \in \mathcal{V}_i$, for all $j \neq i$,

$$\sum_{k=1}^{K}\left(\hat{N}_{x,\mathcal{V}_k}\ln\frac{p_{i,k}}{p_{j,k}}+\hat{N}_{\mathcal{V}_k,x}\ln\frac{p_{k,i}\alpha_j}{p_{k,j}\alpha_i}\right)+\left(\frac{\hat{N}_{\mathcal{V}_j,\mathcal{V}}}{\alpha_j n}-\frac{\hat{N}_{\mathcal{V}_i,\mathcal{V}}}{\alpha_i n}\right)\geq\frac{T}{2n}I(\alpha,p).$$

(H2) $\hat{N}_{x,\mathcal{V}\setminus\mathcal{H}}+\hat{N}_{\mathcal{V}\setminus\mathcal{H},x}\leq 2\ln((T/n)^2)$.

States in $\mathcal{H}$ satisfy properties (H1) and (H2), which guarantee that these states will likely be assigned to their true cluster in our greedy repeated local maximization of the log-likelihood function. To be precise, the set is designed such that the cardinality of the intersection $\mathcal{E}_{\mathcal{H}}^{[t]} \triangleq \mathcal{E}^{[t]} \cap \mathcal{H}$ of incorrectly classified states and well-behaved states shrinks at each iteration with high probability. Furthermore, as we will show this set converges to the empty set with high probability after $t \approx \ln n$ iterations. Because we cannot guarantee that states in $\mathcal{H}^{\mathrm{c}}$ are classified correctly as the algorithm greedily allocates vertices to clusters, we simply treat all of these vertices as being misclassified—a worst-case upper bound. By then estimating the size of $\mathcal{H}^{\mathrm{c}}$, we are able to bound the total number of misclassified vertices after $t \in \mathbb{N}_0$ improvement steps of the Cluster Improvement Algorithm.

How we prove that $|\mathcal{E}_{\mathcal{H}}^{[t]}|$ shrinks. Proposition 8 quantifies how $|\mathcal{E}_{\mathcal{H}}^{[t]}|$ is reduced in a single iteration.

PROPOSITION 8. *If $I(\alpha,p)>0$ and $T=\omega(n)$, and $|\mathcal{E}_{\mathcal{H}}^{[t]}|=O_{\mathbb{P}}(e_n^{[t]})$ for some $0<e_n^{[t]}=o(\frac{n}{\ln(T/n)})$, then*

$$|\mathcal{E}_{\mathcal{H}}^{[t+1]}| \asymp_{\mathbb{P}} e_n^{[t+1]} = O\left(e_n^{[t]}\frac{n}{T}\ln\left(\frac{T}{n}\right)\right)=o(e_n^{[t]}). \tag{10}$$

To establish Proposition 8, observe that after the $(t+1)$th iteration, for any misclassified state x, its true cluster $\sigma(x)$ does not maximize the objective function $u_x^{[t]}(c)$. Hence, summing over all misclassified states that also belong to $\mathcal{H}$, we obtain

$$E \triangleq \sum_{x\in\mathcal{E}_{\mathcal{H}}^{[t+1]}}\left(u_x^{[t]}(\sigma^{[t+1]}(x))-u_x^{[t]}(\sigma(x))\right)\geq 0. \tag{11}$$

We prove Proposition 8 by analyzing E. After substituting $u_x^{[t]}$'s definition (4) into (11), we decompose E as $E=E_1+E_2+U$, where

$$\begin{aligned}
E_1 &= \sum_{x\in\mathcal{E}_{\mathcal{H}}^{[t+1]}}\Bigg\{\sum_{k=1}^{K}\left(\hat{N}_{x,\mathcal{V}_k}\ln\frac{p_{\sigma^{[t+1]}(x),k}}{p_{\sigma(x),k}}+\hat{N}_{\mathcal{V}_k,x}\ln\frac{p_{k,\sigma^{[t+1]}(x)}\alpha_{\sigma(x)}}{p_{k,\sigma(x)}\alpha_{\sigma^{[t+1]}(x)}}\right)\\
&\qquad+\left(\frac{\hat{N}_{\mathcal{V}_{\sigma(x)},\mathcal{V}}}{\alpha_{\sigma(x)}n}-\frac{\hat{N}_{\mathcal{V}_{\sigma^{[t+1]}(x)},\mathcal{V}}}{\alpha_{\sigma^{[t+1]}(x)}n}\right)\Bigg\},\\
E_2 &= \sum_{x\in\mathcal{E}_{\mathcal{H}}^{[t+1]}}\sum_{k=1}^{K}\Bigg((\hat{N}_{x,\hat{\mathcal{V}}_k^{[t]}}-\hat{N}_{x,\mathcal{V}_k})\ln\frac{p_{\sigma^{[t+1]}(x),k}}{p_{\sigma(x),k}}\\
&\qquad+(\hat{N}_{\hat{\mathcal{V}}_k^{[t]},x}-\hat{N}_{\mathcal{V}_k,x})\ln\frac{p_{k,\sigma^{[t+1]}(x)}}{p_{k,\sigma(x)}}\Bigg),\\
U &= E-E_1-E_2.
\end{aligned}$$

Importantly, note that E_1 and E_2 account for the true model parameters p and α, whereas in the functions $u_x^{[t]}$ used in the algorithm, we replace these parameters by their estimates $\hat{p}$ and $\hat{\alpha}$. The term U hence captures issues due to estimation errors. By construction of $\mathcal{H}$, $E_1 \le -(T/n)I(\alpha, p)|\mathcal{E}_{\mathcal{H}}^{[t+1]}|$. Using concentration results for the BMC, we can show that $E_2 \approx \|\hat{N}_\Gamma - N\|(|\mathcal{E}_{\mathcal{H}}^{[t+1]}||\mathcal{E}_{\mathcal{H}}^{[t]}|)^{1/2}$ and that $U \approx 0$, asymptotically. These observations are formalized in Lemma 7, proved in Section SM5.2 of the Supplementary Material [25].

LEMMA 7. *If* $T = \omega(n)$, $|\mathcal{E}_{\mathcal{H}}^{[t]}| = O_{\mathbb{P}}(e_n^{[t]})$ *for some* $0 < e_n^{[t]} = o(\frac{n}{\ln(T/n)})$ *and* $|\mathcal{E}_{\mathcal{H}}^{[t+1]}| \asymp_{\mathbb{P}} e_n^{[t+1]}$, *then*:

$$-E_1 = \Omega_{\mathbb{P}}(I(\alpha, p)\tfrac{T}{n}e_n^{[t+1]}),$$
$$|U| = O(e_n^{[t+1]}\sqrt{\tfrac{T}{n}\ln\tfrac{T}{n}} + e_n^{[t+1]}\tfrac{e_n^{[t]}}{n}\tfrac{T}{n}\ln\tfrac{T}{n}), \text{ and}$$
$$|E_2| \le F_1 + F_2 + F_3 \text{ with } F_1 = O_{\mathbb{P}}(\tfrac{T}{n}\tfrac{e_n^{[t]}}{n}e_n^{[t+1]}),$$
$$F_2 = O_{\mathbb{P}}(\sqrt{\tfrac{T}{n}\ln(\tfrac{T}{n})e_n^{[t]}e_n^{[t+1]}}), \text{ and } F_3 = O_{\mathbb{P}}(\ln(T/n)^2 e_n^{[t+1]}).$$

Combining Lemma 7 with the fact that $E \ge 0$ yields (10). Indeed, we have $-E_1 \le |E_2| + |U|$ almost surely. Now observe that since $\sqrt{\frac{T}{n}}\ln\frac{T}{n} = o(\frac{T}{n})$ and by assumption $e_n^{[t]} = o(n/\ln(T/n))$, $|U|$ is negligible compared to $-E_1$, which implies that $-E_1 - |U| = \Omega_{\mathbb{P}}(I(\alpha, p)\frac{T}{n}e_n^{[t+1]})$. Similarly, since by assumption $e_n^{[t]} = o(n)$ and $\ln((T/n)^2) = o(\frac{T}{n})$, F_1 and F_3 are also negligible compared to $-E_1$, which implies that $-E_1 - |U| - F_1 - F_3 = \Omega_{\mathbb{P}}(I(\alpha, p)\frac{T}{n}e_n^{[t+1]})$. Finally, in view of Lemma 7 $\Omega_{\mathbb{P}}(I(\alpha, p)\frac{T}{n}e_n^{[t+1]}) = -E_1 - |U| - F_1 - F_3 \le F_2 = O_{\mathbb{P}}(\sqrt{\frac{T}{n}\ln\frac{T}{n}e_n^{[t]}e_n^{[t+1]}})$. We deduce that $I(\alpha, p)e_n^{[t+1]} = O(\sqrt{\frac{n}{T}\ln\frac{T}{n}e_n^{[t]}e_n^{[t+1]}})$; see Lemma 25 for the precise justification. Since $I(\alpha, p) > 0$, we obtain $e_n^{[t+1]} = O(e_n^{[t]}\frac{n}{T}\ln\frac{T}{n})$, which concludes the proof of Proposition 8.

How we bound the size of $\mathcal{H}^c$. Proposition 9 provides an upper bound of the number of states in $\mathcal{H}^c$, and is proved in Section SM5.1 of the Supplementary Material [25].

PROPOSITION 9. *If* $I(\alpha, p) > 0$ *and* $T = \omega(n)$, *and* $|\mathcal{E}_{\mathcal{H}}^{[t]}| = O_{\mathbb{P}}(e_n^{[t]})$ *for some* $0 < e_n^{[t]} = o(n)$, *then* $|\mathcal{E}_{\mathcal{H}^c}^{[t]}| \le |\mathcal{H}^c| = O_{\mathbb{P}}(n\exp(-C\frac{T}{n}I(\alpha, p)))$, *where* $C = \alpha_{\min}^2/(720\eta^3\alpha_{\max}^2)$, $\alpha_{\max} = \max_i \alpha_i$, *and* $\alpha_{\min} = \min_i \alpha_i$.

We now sketch the proof of Proposition 9. First, note that the number of states not in Γ (obtained after the trimming process) is negligible, that is, $n\exp(-\frac{T}{n}\ln\frac{T}{n})$. We then upper bound the number of states that do not satisfy (H1). Let $x \in \mathcal{V}_i$. If x does not satisfy (H1), there exists $j \ne i$ such that $\hat{I}_{i,j}(x) < \frac{T}{2n}I(\alpha, p)$, where

$$\hat{I}_{i,j}(x) \triangleq \sum_{k=1}^{K}\left(\hat{N}_{x,\mathcal{V}_k}\ln\frac{p_{i,k}}{p_{j,k}} + \hat{N}_{\mathcal{V}_k,x}\ln\frac{p_{k,i}\alpha_j}{p_{k,j}\alpha_i}\right) + \left(\frac{\hat{N}_{\mathcal{V}_j,\mathcal{V}}}{\alpha_j n} - \frac{\hat{N}_{\mathcal{V}_i,\mathcal{V}}}{\alpha_i n}\right).$$

Observe that $\mathbb{E}[\hat{I}_{i,j}(x)] = \frac{T}{n}I_{i,j}(\alpha, p)$ where $I_{i,j}(\alpha, p)$ is the quantity involved in the definition of $I(\alpha, p)$; see (3). In particular, $\mathbb{E}[\hat{I}_{i,j}(x)] \ge \frac{T}{n}I(\alpha, p)$. Hence, x does not satisfy (H1) implies that for some $j \ne i$, $\hat{I}_{i,j}(x) < \frac{T}{2n}I(\alpha, p)$ and $\mathbb{E}[\hat{I}_{i,j}(x)] \ge \frac{T}{n}I(\alpha, p)$. Using concentration results for the BMC Section SM1 of the Supplementary Material [25], this event happens with probability at most $\exp(-C\frac{T}{n}I(\alpha, p))$. We next deduce a bound on the expected number of states not satisfying (H1). From there, using Markov's inequality, we obtain that

the number of states not satisfying (H1) does not exceed $n \exp(-C\frac{T}{n} I(\alpha, p))$ with high probability. Note that the constant $C = \alpha_{\min}^2/(720\eta^3\alpha_{\max}^2)$ stems from the precise application of concentration results.

We then complete the proof through the following argument. Consider the following iterative construction: start with the set $Z(0)$ of all states that do not satisfy (H1). The tth iteration consists of adding to $Z(t-1)$ a state v not satisfying (H2) written w.r.t. $Z(t-1)$, that is, $\hat{N}_{v,Z(t-1)} + \hat{N}_{Z(t-1),v} > 2\ln((T/n)^2)$. If such a state does not exist, the construction stops. Let $Z(t^*)$ be the final set: t^* is the number of iterations before the construction stops. Then $Z(t^*)$ is such that for all $x \notin Z(t^*)$, x satisfies (H1) and (H2) written w.r.t. $Z(t^*)$. Hence by definition of $\mathcal{H}$, the size of $\mathcal{V} \setminus Z(t^*)$ is smaller than that of $\mathcal{H}$, and thus $|\mathcal{H}^c| \leq |Z(t^*)|$. To get an upper bound on $|Z(t^*)|$, we just establish an upper bound on t^* using concentration results (at each iteration t, a *large* number (specifically $2\ln((T/n)^2)$) of observed transitions inside $Z(t)$ is added, which rapidly becomes impossible). In summary, $|\mathcal{H}^c| \leq |Z(t^*)| \leq t^* + |Z(0)| \leq t^* + n\exp(-C\frac{T}{n} I(\alpha, p))$ with high probability.

Iterating the bound. If we initiate the Cluster Improvement Algorithm using the cluster assignment provided by the Spectral Clustering Algorithm when $T = \omega(n)$, from Theorem 2, we satisfy the initial condition $|\mathcal{E}_{\mathcal{H}}^{[0]}| = o_{\mathbb{P}}(\frac{n}{\ln(T/n)})$ of Propositions 8, 9. Furthermore, since $|\mathcal{E}^{[t]}| = |\mathcal{E}_{\mathcal{H}}^{[t]}| + |\mathcal{E}_{\mathcal{H}^c}^{[t]}|$, we conclude by iterating the bound in (10) that after $t \in \mathbb{N}_0$ improvement steps the Cluster Improvement Algorithm misclassifies at most

$$|\mathcal{E}^{[t]}| = O_{\mathbb{P}}\big(\mathrm{e}^{\ln n - t(\ln\frac{T}{n} - \ln\ln\frac{T}{n})} + \mathrm{e}^{\ln n - \frac{\alpha_{\min}^2}{720\eta^3\alpha_{\max}^2}\frac{T}{n} I(\alpha,p)} + \mathrm{e}^{\ln n - \frac{T}{n}\ln\frac{T}{n}}\big)$$

states. This completes the proof of Theorem 3.

Acknowledgments. We would like to thank our anonymous referees: their careful reading and suggestions have led to improved revisions of this work. We also thank Pascal Lagerweij for having conducted the numerical experiment in Section 4.4. Alexandre Proutière's research is partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. Se–Young Yun would like to acknowledge the support from the National Research Foundation of Korea (NRF) Grant 2019028324 and Artificial Intelligence Graduate School Program (KAIST) Grant IITP 2019-0-00075, funded by the government of Korea (MSIT).

Jaron Sanders is currently at Eindhoven University of Technology.

SUPPLEMENTARY MATERIAL

Supplement to "Clustering in Block Markov Chains" (DOI: 10.1214/19-AOS1939SUPP; .pdf). In our Supplementary Material, we present the detailed proofs of Theorems 1–3, Propositions 1–9, and Lemmas 1–7. You will additionally find supporting theorems stating, for example, key concentration results for Block Markov Chains. We finally present some results pertaining to the order in probability notation that we have used throughout this paper.

REFERENCES

[1] ABBE, E., BANDEIRA, A. S. and HALL, G. (2016). Exact recovery in the stochastic block model. *IEEE Trans. Inf. Theory* **62** 471–487. MR3447993 https://doi.org/10.1109/TIT.2015.2490670

[2] ABBE, E. and SANDON, C. (2015). Community detection in general stochastic block models: Fundamental limits and efficient algorithms for recovery. In 2015 *IEEE* 56*th Annual Symposium on Foundations of Computer Science—FOCS* 2015 670–688. IEEE Computer Soc., Los Alamitos, CA. MR3473334 https://doi.org/10.1109/FOCS.2015.47

[3] ABBE, E. and SANDON, C. (2015). Recovering communities in the general stochastic block model without knowing the parameters. In *Advances in Neural Information Processing Systems* 28 676–684.
[4] BORDENAVE, C., CAPUTO, P. and CHAFAÏ, D. (2010). Spectrum of large random reversible Markov chains: Two examples. *ALEA Lat. Am. J. Probab. Math. Stat.* **7** 41–64. MR2644041
[5] BORDENAVE, C., CAPUTO, P. and CHAFAÏ, D. (2011). Spectrum of large random reversible Markov chains: Heavy-tailed weights on the complete graph. *Ann. Probab.* **39** 1544–1590. MR2857250 https://doi.org/10.1214/10-AOP587
[6] BRÉMAUD, P. (1999). *Markov Chains*: *Gibbs Fields*, *Monte Carlo Simulation*, *and Queues*. *Texts in Applied Mathematics* **31**. Springer, New York. MR1689633 https://doi.org/10.1007/978-1-4757-3124-8
[7] DECELLE, A., KRZAKALA, F., MOORE, C. and ZDEBOROVÁ, L. (2011). Inference and phase transitions in the detection of modules in sparse networks. *Phys. Rev. Lett.* **107** Art. ID 065701. https://doi.org/10.1103/PhysRevLett.107.065701
[8] FEIGE, U. and OFEK, E. (2005). Spectral techniques applied to sparse random graphs. *Random Structures Algorithms* **27** 251–275. MR2155709 https://doi.org/10.1002/rsa.20089
[9] GAO, C., MA, Z., ZHANG, A. Y. and ZHOU, H. H. (2017). Achieving optimal misclassification proportion in stochastic block models. *J. Mach. Learn. Res.* **18** Art. ID 60. MR3687603
[10] HAJEK, B., WU, Y. and XU, J. (2016). Achieving exact cluster recovery threshold via semidefinite programming. *IEEE Trans. Inf. Theory* **62** 2788–2797. MR3493879 https://doi.org/10.1109/TIT.2016.2546280
[11] HALKO, N., MARTINSSON, P. G. and TROPP, J. A. (2011). Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Rev.* **53** 217–288. MR2806637 https://doi.org/10.1137/090771806
[12] HOCHSTÄTTLER, W., KIRSCH, W. and WARZEL, S. (2016). Semicircle law for a matrix ensemble with dependent entries. *J. Theoret. Probab.* **29** 1047–1068. MR3540489 https://doi.org/10.1007/s10959-015-0602-3
[13] HOLLAND, P. W., LASKEY, K. B. and LEINHARDT, S. (1983). Stochastic blockmodels: First steps. *Soc. Netw.* **5** 109–137. MR0718088 https://doi.org/10.1016/0378-8733(83)90021-7
[14] JOG, V. and LOH, P.-L. (2015). Information-theoretic bounds for exact recovery in weighted stochastic block models using the Renyi divergence. Preprint. Available at arXiv:1509.06418.
[15] KIRSCH, W. and KRIECHERBAUER, T. (2018). Sixty years of moments for random matrices. In *Non-linear Partial Differential Equations*, *Mathematical Physics*, *and Stochastic Analysis*. *EMS Ser. Congr. Rep.* 349–379. Eur. Math. Soc., Zürich. MR3823851
[16] KIRSCH, W. and KRIECHERBAUER, T. (2018). Semicircle law for generalized Curie–Weiss matrix ensembles at subcritical temperature. *J. Theoret. Probab.* **31** 2446–2458. MR3866621 https://doi.org/10.1007/s10959-017-0768-y
[17] LAI, T. L. and ROBBINS, H. (1985). Asymptotically efficient adaptive allocation rules. *Adv. in Appl. Math.* **6** 4–22. MR0776826 https://doi.org/10.1016/0196-8858(85)90002-8
[18] LEVIN, D. A., PERES, Y. and WILMER, E. L. (2009). *Markov Chains and Mixing Times*. Amer. Math. Soc., Providence, RI. MR2466937
[19] LLOYD, S. P. (1982). Least squares quantization in PCM. *IEEE Trans. Inf. Theory* **28** 129–137. MR0651807 https://doi.org/10.1109/TIT.1982.1056489
[20] MASSOULIÉ, L. (2014). Community detection thresholds and the weak Ramanujan property. In *STOC'*14—*Proceedings of the* 2014 *ACM Symposium on Theory of Computing* 694–703. ACM, New York. MR3238997
[21] MOSSEL, E., NEEMAN, J. and SLY, A. (2015). Reconstruction and estimation in the planted partition model. *Probab. Theory Related Fields* **162** 431–461. MR3383334 https://doi.org/10.1007/s00440-014-0576-6
[22] MOSSEL, E., NEEMAN, J. and SLY, A. (2015). Consistency thresholds for the planted bisection model. In *Proceedings of the Forty-Seventh Annual ACM Symposium on Theory of Computing* 69–75. ACM, New York.
[23] ORTNER, R. (2013). Adaptive aggregation for reinforcement learning in average reward Markov decision processes. *Ann. Oper. Res.* **208** 321–336. MR3100636 https://doi.org/10.1007/s10479-012-1064-y
[24] PAULIN, D. (2015). Concentration inequalities for Markov chains by Marton couplings and spectral methods. *Electron. J. Probab.* **20** Art. ID 79. MR3383563 https://doi.org/10.1214/EJP.v20-4039
[25] SANDERS, J., PROUTIÈRE, A. and YUN, S.-Y. (2020). Supplement to "Clustering in Block Markov Chains." https://doi.org/10.1214/19-AOS1939SUPP
[26] SINGH, S. P., JAAKKOLA, T. and JORDAN, M. I. (1995). Reinforcement learning with soft state aggregation. In *Advances in Neural Information Processing Systems* 7 (G. Tesauro, D. S. Touretzky and T. K. Leen, eds.) 361–368. MIT Press, Cambridge, MA.
[27] SUTTON, R. S. and BARTO, A. G. (1998). *Introduction to Reinforcement Learning*, 1st ed. MIT Press, Cambridge, MA. MR3889951

[28] TAO, T. (2012). *Topics in Random Matrix Theory*. *Graduate Studies in Mathematics* **132**. Amer. Math. Soc., Providence, RI. MR2906465 https://doi.org/10.1090/gsm/132

[29] TROPP, J. A. (2015). An introduction to matrix concentration inequalities. *Found. Trends Mach. Learn.* **8** 1–230.

[30] WIGNER, E. P. (1958). On the distribution of the roots of certain symmetric matrices. *Ann. of Math.* (2) **67** 325–327. MR0095527 https://doi.org/10.2307/1970008

[31] YUN, S.-Y. and PROUTIERE, A. (2014). Community detection via random and adaptive sampling. In *COLT* 138–175.

[32] YUN, S.-Y. and PROUTIERE, A. (2014). Accurate community detection in the stochastic block model via spectral algorithms. Preprint. Available at arXiv:1412.7335.

[33] YUN, S.-Y. and PROUTIERE, A. (2016). Optimal cluster recovery in the labeled stochastic block model. In *Advances in Neural Information Processing Systems* 29 965–973.