RAPID MIXING OF THE SWITCH MARKOV CHAIN FOR 2-CLASS JOINT DEGREE MATRICES*

GEORGIOS AMANATIDIS[†] AND PIETER KLEER[‡]

Abstract. The switch Markov chain has been extensively studied as the most natural Markov Chain Monte Carlo approach for sampling graphs with prescribed degree sequences. In this work we study the problem of uniformly sampling graphs for which, in addition to the degree sequence, joint degree constraints are given. These constraints specify how much edges there should be between two given degree classes (i.e., subsets of nodes that all have the same degree). Although the problem was formalized over a decade ago, and despite its practical significance in generating synthetic network topologies, small progress has been made on the random sampling of such graphs. In the case of one degree class, the problem reduces to the sampling of regular graphs (i.e., graphs in which all nodes have the same degree), but beyond this very little is known. We fully resolve the case of two degree classes, by showing that the switch Markov chain is always rapidly mixing. We do this by combining a recent embedding argument developed by the authors in combination with ideas of Bhatnagar, Randall, Vazirani and Vigoda (2006) introduced in the context of sampling bichromatic matchings.

Key words. graph sampling, switch Markov chain, joint degree matrix

AMS subject classifications. 68Q25, 05C80, 68R10

1. Introduction.

The (approximate) uniform sampling of simple graphs with given degrees is a problem that has been studied extensively in the last couple of decades. One prominent approach is the Markov Chain Monte Carlo method, with the most notable example being the *switch Markov chain*. Here, starting from a graph with the target degree sequence, one repeatedly selects uniformly at random two edges and "switches" them if this preserves the simplicity of the graph (see Figure 2). Typically, the goal is to show that the induced Markov chain is *rapidly mixing*, i.e., that only a polynomial number of steps (as a function of the number of vertices) is needed to get close to the uniform distribution over all graphs with the given degree sequence. The switch chain has been shown to be rapidly mixing for various degree sequences but it is still open whether or not it is rapidly mixing for all degree sequences; see [3, 14] for the state of the art.

In this work, we focus on the problem where, in addition to the degree sequence, a so-called *joint degree matrix* with degree correlations is specified. This matrix encodes how many edges there should be between nodes of different degrees. The motivation for using such a metric is that this extra information restricts the space of possible realizations to graphs with more desirable structure. This was first observed by Mahadevan et al. [26] who argued that the joint degree matrix is a much more reliable metric for a synthetic graph (i.e., a graph not obtained from empirical data but generated by a random graph model) to resemble a real network topology, compared to just using the degree sequence. The *joint degree matrix model* of Amanatidis, Green, and Mihail [1] formalizes this approach as follows.

Suppose we are given a *degree sequence* $\boldsymbol{d} = (d_i)_{i\in V}$, specifying the degree of each

*A preliminary version of this work appeared, as part of a larger work, in the Proceedings of the 30th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2019) [3].

Funding: This work was funded by the Dutch Research Council (NWO): Gravitation project NETWORKS, Grant Number 024.002.003.

[†]University of Essex, United Kingdom, and University of Amsterdam, The Netherlands; georgios.amanatidis@essex.ac.uk.

[‡]Max-Planck-Institut für Informatik, Germany. Current affiliation: Tilburg University, The Netherlands; p.s.kleer@tilburguniversity.edu.

node of V, where $|V| = n$, and a *joint degree matrix* $\boldsymbol{c} = (c_{ij})_{i,j\in[q]}$ specifying the number of edges between nodes of different degrees, where q is the maximum degree in the sequence $\boldsymbol{d}$. A graph $G = (V, E)$ with degree sequence $\boldsymbol{d}$, for which there are precisely c_{ij} edges between nodes of degree i and degree j for every pair (i, j), is called a *realization* of the pair $(\boldsymbol{c}, \boldsymbol{d})$. We write $\mathcal{G}(\boldsymbol{c}, \boldsymbol{d})$ to denote the set of all such graphs. We want to approximately uniformly sample graphs from $\mathcal{G}(\boldsymbol{c}, \boldsymbol{d})$.

EXAMPLE 1.1. *We let $n = 11$, and consider the degree sequence $\boldsymbol{d}$ and joint degree matrix $\boldsymbol{c}$ given by*

$$\boldsymbol{c} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 7 & 4 \\ 0 & 0 & 4 & 8 \end{pmatrix} \quad \textit{and} \quad \boldsymbol{d} = (3, 3, 3, 3, 3, 3, 4, 4, 4, 4, 4)\,.$$

This means that there are six nodes of degree three, five nodes of degree four, and there are in total four edges between the nodes of degree three and four. In Figure 1 below we give a possible realization of $(\boldsymbol{c}, \boldsymbol{d})$.

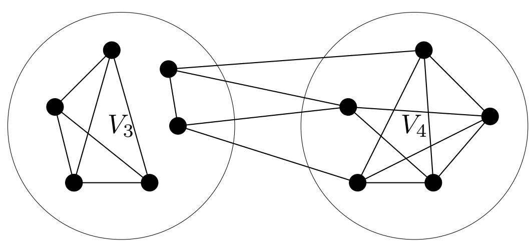

FIG. 1. *An example of a realization for $(\boldsymbol{c}, \boldsymbol{d})$ as given in Example 1.1.*

Although there are polynomial-time algorithms that produce a realization of a given degree sequence and joint degree matrix [1,2,11,19,30], almost nothing is known about how to uniformly sample a realization from $\mathcal{G}(\boldsymbol{c}, \boldsymbol{d})$ efficiently. The only related work here is that of Erdős et al. [17] who show rapid mixing on the subset of *balanced* joint degree matix realizations (see Section 1.1 for a description of this problem). In this work we provide the first polynomial-time sampling results for the case of *any two* degree classes. The instance in Figure 1 is an example of this case.

In particular, we bound the mixing time of the arguably most natural Markov chain on the set of all graphs with a given joint degree matrix, the *(restricted) switch Markov chain*. Bounding the mixing time of this chain has been an open problem since the introduction of the joint degree matrix model [1, 17, 30]. It proceeds by repeatedly selecting two edges of a realization and rewiring them if possible, while preserving the degree sequence *and* the joint degree matrix. An example is given in Figure 2.

Although the switch Markov chain has been extensively studied for the uniform sampling of graphs with a given degree sequence (without a joint degree matrix), nothing is known when additionally degree correlation constraints are given (in [17] the switch chain is considered for the *balanced* joint degree matrix problem).

The main contribution of this work is showing that the switch chain is *always* rapidly mixing on the space of realizations of a given joint degree matrix with two degree classes (Theorem 3.1). Despite being for the case of two classes, this is the

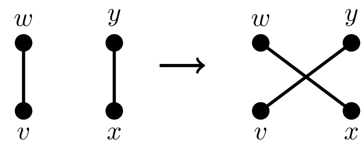

FIG. 2. *Example of a switch in which edges* $\{v, w\}, \{x, y\}$ *are replaced by* $\{v, y\}, \{x, w\}$. *Such a switch operation always preserves the degree sequence of a graph, although not necessarily joint degree constraints.*

very first rapid mixing result for the problem of sampling from $\mathcal{G}(\boldsymbol{c}, \boldsymbol{d})$. The proof consists of two main ingredients.

We first analyze an auxiliary chain, the so-called *hinge flip chain* (see Figure 3 for an example of a hinge flip). The state space of this Markov chain also contains graphs with a slightly perturbed degree sequence or joint degree matrix $(\boldsymbol{c}', \boldsymbol{d}')$ compared to $(\boldsymbol{c}, \boldsymbol{d})$. Therefore, we study this chain in the more general so-called *partition adjacency matrix (PAM) model* [10, 15] for two classes and show it is rapidly mixing in case the pair $(\boldsymbol{c}, \boldsymbol{d})$ comes from a class of *strongly stable pairs*. The PAM model is a generalization of the JDM model in which nodes within the same partition do not necessarily have the same degree (Section 2.2). Strong stability refers to the fact that any graph that has a slightly perturbed degree sequence or joint degree matrix $(\boldsymbol{c}', \boldsymbol{d}')$ can be turned into a graph satisfying the constraints of $(\boldsymbol{c}, \boldsymbol{d})$ with only a few hinge flips. This is a generalization of the notion of strong stability introduced in [4] for degree sequences.

Establishing the rapid mixing of the hinge flip chain in our setting presents significant challenges. To attack this problem, we partly rely on ideas introduced by Bhatnagar, Randall, Vazirani and Vigoda [5] in the context of sampling *exact (perfect) matchings*.[1] At the core of this approach lies the *mountain climbing problem* [21, 31]. Secondly, we use an embedding argument similar to that in [4] to show that the rapid mixing of the hinge flip Markov chain can be carried over to the switch Markov chain in the case of strongly stable pairs $(\boldsymbol{c}, \boldsymbol{d})$ for the joint degree matrix model with two degree classes.

As a byproduct of our analysis for the hinge flip chain, we obtain the first *fully polynomial almost uniform generator* for sampling realizations of certain sparse PAM instances with two partition classes (Corollary 4.3).

1.1. Further Related Work. The joint degree matrix model was first studied by Patrinos and Hakimi [28], albeit with a different formulation and name, and was reintroduced in Amanatidis et al. [1]. While it has been shown that the switch chain restricted on the space of the realizations of any given joint degree matrix is irreducible [1, 11], almost no progress has been made towards bounding its mixing time. Stanton and Pinar [30] performed experiments based on the autocorrelation of each edge, suggesting that the switch chain mixes quickly. The only relevant result is that of Erdős et al. [17] showing rapid mixing for a related Markov chain over the restricted subset of so-called *balanced* joint degree matrix realizations with an arbitrary number of degree classes. A realization is balanced if the following is satisfied for all pair of degree classes: If one considers the edges between two degree classes, then every node in a given class should be adjacent to roughly the same number of edges (with

[1] For a given red-blue edge-colored undirected graph G and $k \in \boldsymbol{N}$, a perfect matching is called *exact* if it has precisely k red edges.

a difference of at most one).

Our hinge-flip Markov chain is essentially a generalization of (a variant of) a Markov chain introduced by Jerrum and Sinclair [23].

The switch Markov chain for sampling graphs with a given degree sequence (without joint degree matrix) has been studied extensively, see, e.g., [3, 6–9, 14, 16, 17, 20, 24, 27]. Switch-based Markov chain have also been studied for sampling connected graphs [18] and (perfect) matchings, see, e.g., [13] and references therein.

1.2. Outline. Section 2 contains all the necessary preliminaries, beginning with all needed Markov chain definitions and facts. Then the PAM and the JDM models are introduced, along with the corresponding Markov chains: the hinge flip chain and the switch chain. In Section 3 we first show rapid mixing of the hinge flip chain for strongly stable instances. After showing that all JDM instances with two degree classes are strongly stable, we prove that the rapid mixing of the hinge flip chain in this case implies the rapid mixing of the switch chain. Finally, in Section 4 we show rapid mixing of the hinge flip chain for certain sparse PAM instances.

2. Preliminaries. We begin with the necessary background on Markov chains and the multicommodity flow method of Sinclair [29]. For Markov chain definitions not given here, see, e.g., [25].

Let $\mathcal{M} = (\Omega, P)$ be an ergodic, time-reversible Markov chain over state space Ω with transition matrix P and stationary distribution π. We write $P^t(x,\cdot)$ for the distribution over Ω at time step t given that the initial state is $x \in \Omega$. The *total variation distance at time t with initial state x* is

$$\Delta_x(t) = d_{TV}(P^t(x,\cdot),\pi) = \max_{S\subseteq\Omega} \left|P^t(x,S) - \pi(S)\right| = \frac{1}{2}\sum_{y\in\Omega}\left|P^t(x,y) - \pi(y)\right|,$$

and the *mixing time* $\tau(\epsilon)$ is defined as

$$\tau(\epsilon) = \max_{x\in\Omega}\left\{\min\{t : \Delta_x(t') \leq \epsilon \text{ for all } t' \geq t\}\right\}.$$

Informally, $\tau(\epsilon)$ is the number of steps until the Markov chain is ϵ-close to its stationary distribution independently of the initial state $x \in \Omega$. A Markov chain is said to be *rapidly mixing* if the mixing time can be upper bounded by a function polynomial in $\ln(|\Omega|/\epsilon)$.

It is well-known that, since the Markov chain is time-reversible, the matrix P only has real eigenvalues $1 = \lambda_0 > \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{|\Omega|-1} > -1$. We may replace the transition matrix P of the Markov chain by $(P + I)/2$, to make the chain *lazy*, and hence guarantee that all its eigenvalues are non-negative. It then follows that the second-largest eigenvalue of P is λ_1. In this work we always consider the lazy versions of the Markov chains involved. It follows directly from Proposition 1 in [29] that

$$\tau(\epsilon) \leq \frac{1}{1-\lambda_1}\big(\ln(1/\pi_*) + \ln(1/\epsilon)\big),$$

where $\pi_* = \min_{x\in\Omega}\pi(x)$. For the special case where π is the uniform distribution, the above bound becomes

$$\tau(\epsilon) \leq \frac{1}{1-\lambda_1}(\ln(|\Omega|) + \ln(1/\epsilon)).$$

The quantity $(1-\lambda_1)^{-1}$ can be upper bounded using the *multicommodity flow method* of Sinclair [29], outlined next.

We define the state space graph of the chain $\mathcal{M}$ as the directed graph $\boldsymbol{G}$ with node set Ω that contains exactly the edges $(x, y) \in \Omega \times \Omega$ for which $P(x, y) > 0$ and $x \neq y$. Let $\mathcal{P} = \cup_{x \neq y} \mathcal{P}_{xy}$, where $\mathcal{P}_{xy}$ is the set of simple paths between x and y in the state space graph $\boldsymbol{G}$. A *flow* f in Ω is a function $\mathcal{P} \to [0, \infty)$ satisfying $\sum_{p \in \mathcal{P}_{xy}} f(p) = \pi(x)\pi(y)$ for all $x, y \in \Omega, x \neq y$. The flow f can be extended to a function on oriented edges of $\boldsymbol{G}$ by setting $f(e) = \sum_{p \in \mathcal{P}: e \in p} f(p)$, so that $f(e)$ is the total flow routed through $e \in E(\boldsymbol{G})$. Let $\ell(f) = \max_{p \in \mathcal{P}: f(p) > 0} |p|$ be the length of a longest flow carrying path, and let $\rho(e) = f(e)/Q(e)$ be the *load* of the edge e, where $Q(e) = \pi(x)P(x, y)$ for $e = (x, y)$. The maximum load of the flow is
$\rho(f) = \max_{e \in E(\boldsymbol{G})} \rho(e)$. Sinclair ([29], Corollary 6′) shows that

$$(1 - \lambda_1)^{-1} \leq \rho(f)\ell(f)\,.$$

We use the following standard technique for bounding the maximum load of a flow in case the chain $\mathcal{M}$ has uniform stationary distribution π. Suppose θ is the smallest positive transition probability of the Markov chain between two distinct states. If b is such that $f(e) \leq b/|\Omega|$ for all $e \in E(\boldsymbol{G})$, then it follows that $\rho(f) \leq b/\theta$. Thus, we have

$$\tau(\epsilon) \leq \frac{\ell(f) \cdot b}{\theta} \ln(|\Omega|/\epsilon)\,.$$

Now, if $\ell(f), b$ and $1/\theta$ can be bounded by a function polynomial in $\log(|\Omega|)$, it follows that the Markov chain $\mathcal{M}$ is rapidly mixing. In this case, we say that f is an *efficient* flow. Note that in this approach the transition probabilities do not play a role as long as $1/\theta$ is polynomially bounded.

2.1. JDM Model and the Restricted Switch Chain. Here we describe the joint degree matrix model, which is a special case of the partition adjacency model that will be described in Section 2.2.

Let $V = \{1, \dots, n\}$ be a set of nodes. An instance of the *joint degree matrix (JDM)* model is given by a partition $V_1 \cup V_2 \cup \cdots \cup V_q$ of V into pairwise disjoint (degree) classes, a symmetric *joint degree matrix* $\boldsymbol{c} = (c_{ij})_{i,j \in [q]}$ of non-negative integers, and a sequence $\boldsymbol{d} = (d_1, \dots, d_q)$ of non-negative integers.[2] Note that in the related literature the d_is are often assumed to be pairwise distinct; however, we do not need this additional assumption for our results. We say that the tuple $((V_i)_{i \in q}, \boldsymbol{c}, \boldsymbol{d})$ (or just $(\boldsymbol{c}, \boldsymbol{d})$ when it is clear what the partition is) is *graphical*, if there exists a simple, undirected, labeled graph $G = (V, E)$ on the nodes in V such that all nodes in V_i have degree d_i and there are precisely c_{ij} edges between nodes in V_i and V_j. Such a G is called a realization of the tuple. We let $\mathcal{G}((V_i)_{i \in q}, \boldsymbol{c}, \boldsymbol{d})$, or just $\mathcal{G}(\boldsymbol{c}, \boldsymbol{d})$, denote the set of all realizations of $((V_i)_{i \in q}, \boldsymbol{c}, \boldsymbol{d})$. We focus on the case of $q = 2$, i.e., when two degree classes are given.

While switches maintain the degree sequence, this is no longer true for the joint degree constraints. However, some switches do respect these constraints as well, e.g., if w, y in Figure 2 are in the same degree class. Thus, we are interested in the following (lazy) *restricted switch Markov chain* for sampling realizations of $\mathcal{G}(\boldsymbol{c}, \boldsymbol{d})$.

[2] This is shorthand notation. More formally, we could write $\hat{\boldsymbol{d}} = (\hat{d}_1, \dots, \hat{d}_n) = (d_1^1, \dots, d_1^{|V_1|}, \dots, d_q^1, \dots, d_q^{|V_q|})$ corresponding to the definition of a graphical degree sequence. In such a case, $d_i^j = d_i$ for $i \in V$ and $j \in \{1, \dots, |V_i|\}$.

Let $G \in \mathcal{G}(\boldsymbol{c}, \boldsymbol{d})$ be the current state of the (restricted) switch chain:

- With probability 1/2, do nothing.
- Otherwise, attempt to perform a *switch* operation: select two edges $\{a, b\}$ and $\{x, y\}$ uniformly at random, and select a perfect matching M on nodes $\{x, y, a, b\}$ uniformly at random. If $M \cap E(G) = \emptyset$ and $G + M - (\{a, b\} \cup \{x, y\}) \in \mathcal{G}(\boldsymbol{c}, \boldsymbol{d})$, then delete $\{a, b\}, \{x, y\}$ from $E(G)$ and add the edges of M.

Theorem 2.1 below, that summarizes some properties of the restricted switch chain, follows from [1, 11].

THEOREM 2.1. *This restricted switch chain is irreducible, aperiodic and symmetric. Like the switch chain defined above, $P(G, G')^{-1} \leq n^4$ for all adjacent $G, G' \in \mathcal{G}'(\boldsymbol{c}, \boldsymbol{d})$, and also the maximum in- and out-degrees of the state space graph are at most n^4.*

For simplicity, in what follows we will drop the term *restricted* and we will simply refer to the *switch chain*.

2.2. PAM Model and the Hinge Flip Chain. We describe the *partition adjacency matrix (PAM)* model [1,10], that generalizes the joint degree matrix model (which was discussed in the Introduction and formally defined in Subsection 2.1). The main difference is that in the PAM model the nodes within a given class need not have the same degrees. Like before, there is a partition of the nodes and there are edge requirements between any two sets of this partition, but now the degrees within each such set can be specified by an arbitrary degree sequence.

Formally, let $V = \{1, \ldots, n\}$ be a given set. An instance of the partition adjacency matrix model is given by a partition $V_1 \cup V_2 \cup \cdots \cup V_q$ of V into pairwise disjoint classes. Moreover, we are given a symmetric *partition adjacency matrix* $\boldsymbol{c} = (c_{ij})_{i,j \in [q]}$ of non-negative integers, and a sequence $\boldsymbol{d} = (d_1, \ldots, d_n)$ of non-negative integers. We say that the tuple $((V_i)_{i \in q}, \boldsymbol{c}, \boldsymbol{d})$ is graphical if there exists a simple, undirected, labeled graph $G = (V, E)$ on the nodes in V with node $v \in V$ having degree d_v, and so that there are precisely c_{ij} edges between endpoints in V_i and V_j. The graph G is then called a *realization* of the tuple. We let $\mathcal{G}((V_i)_{i \in q}, \boldsymbol{c}, \boldsymbol{d})$ denote the set of all realizations of the tuple $((V_i)_{i \in q}, \boldsymbol{c}, \boldsymbol{d})$. We often write $\mathcal{G}(\boldsymbol{c}, \boldsymbol{d})$ instead of $\mathcal{G}((V_i)_{i \in [q]}, \boldsymbol{c}, \boldsymbol{d})$ when it is clear what the partition is.

In this work we focus on the case of a partition into two classes V_1 and V_2, and, without loss of generality, assume that $1 \leq c_{12} \leq |V_1| \cdot |V_2| - 1$. Indeed, it is not hard to see that the cases $c_{12} \in \{0, |V_1| \cdot |V_2|\}$ reduce to the single class case. For the case of two classes, an initial state can be computed in polynomial time [15].[3] We let $\mathcal{G}'(\boldsymbol{c}, \boldsymbol{d}) = \cup_{(\boldsymbol{c}', \boldsymbol{d}')} \mathcal{G}'(\boldsymbol{c}', \boldsymbol{d}')$ with $(\boldsymbol{c}', \boldsymbol{d}')$ ranging over tuples satisfying

(i) $\sum_{i=1}^{n} d_i - d'_i = 0\,,$

(ii) $\sum_{i=1}^{n} |d_i - d'_i| \in \{0, 2, 4\}\,,$

(iii) $c'_{12} \in \{c_{12} - 1, c_{12}, c_{12} + 1\}\,.$

The set $\mathcal{G}'(\boldsymbol{c}, \boldsymbol{d})$ contains all graphs that possibly have a slightly different number of cut edges, as well as at most four nodes who have a slightly different degree than in

[3] For general instances, it is not known if an initial state can be computed in time polynomial in n. It is conjectured to be NP-hard in general [15]; see also [12].

the sequence $\boldsymbol{d}$. We call elements in $\mathcal{G}'(\boldsymbol{c},\boldsymbol{d})\setminus\mathcal{G}(\boldsymbol{c},\boldsymbol{d})$ *perturbed (auxiliary) states*. For any $G\in\mathcal{G}'(\boldsymbol{c},\boldsymbol{d})$ the *perturbation at node* $v\in V$ is defined as $\alpha_v = d'_v - d_v$ where d' is the degree sequence of G. We say that the node v has a degree surplus if $\alpha_v > 0$ and a degree deficit if $\alpha_v < 0$. Moreover, the total degree surplus is defined as $\sum_{v:\alpha_v>0}\alpha_v$, and the total degree deficit as $-\sum_{v:\alpha_v<0}\alpha_v$. Note that

$$\sum_{v:\alpha_v>0}\alpha_v - \sum_{v:\alpha_v<0}\alpha_v = \sum_{i=1}^{n}|d_i - d'_i|\,.$$

Finally, we say that a tuple $(\boldsymbol{c}',\boldsymbol{d}')$ is *edge-balanced* if $\boldsymbol{c}'=\boldsymbol{c}$ (but possibly $\boldsymbol{d}'\neq\boldsymbol{d}$). From the conditions defining $\mathcal{G}'(\boldsymbol{c},\boldsymbol{d})$, we may infer the following properties.

PROPOSITION 2.2. *For any $G\in\mathcal{G}'(\boldsymbol{c},\boldsymbol{d})$, for some tuple $(\boldsymbol{c}',\boldsymbol{d}')$ satisfying (i)-(iii) above, it holds that*

(a) the perturbation at node v satisfies $\alpha_v\in\{-2,-1,0,1,2\}$ for any $v\in V$,

(b) $\max_{i,j=1,2}|c_{ij}-c'_{ij}|\leq 1$, and $\sum_{1\leq i\leq j\leq 2}|c_{ij}-c'_{ij}|\in\{0,2\}$.

Proof. If there is some node with degree surplus greater than or equal to three, then the total degree deficit is also at least three, which follows from the first condition defining $\mathcal{G}'(\boldsymbol{c},\boldsymbol{d})$. This means that $\sum_{i=1}^{n}|d_i-d'_i|\geq 6$, which violates the second condition defining $\mathcal{G}'(\boldsymbol{c},\boldsymbol{d})$. The same argument holds in case there is some node with degree deficit greater than or equal to three.

To see that the second property is true, notice that, because of (iii) in the definition of $\mathcal{G}'(\boldsymbol{c},\boldsymbol{d})$, we only need to consider the cases where $c'_{ii}\geq c_{ii}+2$, for some $i\in\{1,2\}$, or $c'_{ii}\leq c_{ii}-2$, for some $i\in\{1,2\}$. Let us assume $c'_{11}\geq c_{11}+2$. Since $c'_{12}\in\{c_{12}-1,c_{12},c_{12}+1\}$, by (iii), it must be that the total degree surplus of the nodes in V_1 is at least three. This gives a contradiction by arguing exactly like in the first part of the proof above. Analogous arguments hold for the other three cases as well.

Finally, the last property is a direct consequence of $\max_{i,j=1,2}|c_{ij}-c'_{ij}|\leq 1$ and the fact that $\sum_{i,j=1,2}|c_{ij}-c'_{ij}|$ is an even number, because of (i) in the definition of $\mathcal{G}'(\boldsymbol{c},\boldsymbol{d})$. ∎

Remark 2.3. As we focus on the case in which V is partitioned into two classes V_1 and V_2 here, we will sometimes use shorthand notation. Given a sequence $\boldsymbol{d}$, the number $\gamma=c_{12}$ uniquely determines the matrix $\boldsymbol{c}$. Hence, the set $\mathcal{G}((V_1,V_2),\boldsymbol{c},\boldsymbol{d})$ (or, its simplified version $\mathcal{G}(\boldsymbol{c},\boldsymbol{d})$) is then denoted by $\mathcal{G}(V_1,V_2,\gamma,\boldsymbol{d})$ (or, respectively, $\mathcal{G}(\gamma,\boldsymbol{d})$). Similarly for we write $\mathcal{G}'(\gamma,\boldsymbol{d})$ instead of $\mathcal{G}'((V_1,V_2),\boldsymbol{c},\boldsymbol{d})$.

We define the *hinge flip Markov chain* $\mathcal{M}(\gamma,\boldsymbol{d})$ on $\mathcal{G}'(\gamma,\boldsymbol{d})$ as follows.

Let $G\in\mathcal{G}'(\gamma,\boldsymbol{d})$ be the current state of the hinge flip chain:

- With probability 1/2, do nothing.
- Otherwise, attempt to perform a *hinge flip* operation: select an ordered triple i,j,k of nodes uniformly at random. If $\{i,j\}\in E(G)$, $\{j,k\}\notin E(G)$, and $G-\{i,j\}+\{j,k\}\in\mathcal{G}'(\gamma,\boldsymbol{d})$, then delete $\{i,j\}$ and add $\{j,k\}$.

Note that we can check if $G-\{i,j\}+\{j,k\}\in\mathcal{G}'(\gamma,\boldsymbol{d})$ in time polynomial in n based on the state G.

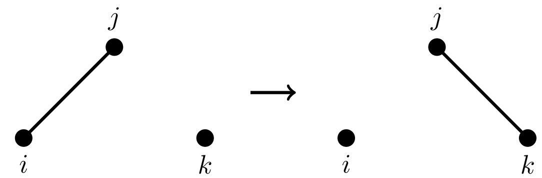

FIG. 3. *Example of a hinge flip operation for the ordered triple* (i, j, k).

Graphs $G, G' \in \mathcal{G}'(\gamma, \boldsymbol{d})$ are said to be *adjacent* in $\mathcal{M}$ if G can be obtained from G' with positive probability in one transition of the chain $\mathcal{M}$. We say that two graphs G, G' are within distance r in $\mathcal{M}$ if there exists a path of at most length r from G to G' in the state space graph of $\mathcal{M}$. By $\text{dist}(G, \gamma, \boldsymbol{d})$ we denote the minimum distance of G from an element in $\mathcal{G}(\gamma, \boldsymbol{d})$. The following parameter is the analogue of $k(\boldsymbol{d})$ of [4] for the current setting and will be used in a similar manner to define the appropriate variant of strong stability.

We define

$$k(\gamma, \boldsymbol{d}) = \max_{G \in \mathcal{G}'(\gamma, \boldsymbol{d})} \text{dist}(G, \gamma, \boldsymbol{d})\,. \tag{2.1}$$

In the PAM model with two degree classes, a family $\mathcal{D}$ of graphical tuples $(\gamma, \boldsymbol{d})$ is called *strongly stable* if there exists a constant k such that $k(\gamma, \boldsymbol{d}) \leq k$ for all $(\gamma, \boldsymbol{d}) \in \mathcal{D}$.

THEOREM 2.4. *Let $\mathcal{D}$ be a strongly stable family of graphical tuples. Then for every $(\gamma, \boldsymbol{d}) \in \mathcal{D}$, the chain $\mathcal{M}(\gamma, \boldsymbol{d})$ is irreducible, aperiodic and symmetric, and, hence, has uniform stationary distribution over $\mathcal{G}'(\gamma, \boldsymbol{d})$. Moreover, $P(G, G')^{-1} \leq n^3$ for all adjacent $G, G' \in \mathcal{G}'(\gamma, \boldsymbol{d})$, and also the maximum in- and out-degrees of the state space graph of the chain $\mathcal{M}(\gamma, \boldsymbol{d})$ are bounded by n^3.*[4]

Proof. The only claim that requires a detailed argument, and uses the assumption of strong stability, is that of the irreducibility of the chain. By definition of strong stability, we always know that every perturbed state is connected to some element in $\mathcal{G}(\gamma, \boldsymbol{d})$ so it suffices to show that there is a path between any two states in $\mathcal{G}(\gamma, \boldsymbol{d})$. This follows from the analysis in Section 3. Aperiodicity follows from the holding probability in the description of the chain $\mathcal{M}$, and symmetry is straightforward. The bound on $P(G, G')^{-1}$ follows directly from the description of the chain, as do the bounds on the in- and out-degrees of the state space graph. ∎

Remark 2.5. In general, the space of all realizations satisfying a given partition adjacency matrix constraint with two classes is not connected under switches [15], which is true in the special case of the joint degree matrix model (see next section). However, it is shown in [15] that it is connected if one also allows *double switches*, in which one is, roughly speaking, allowed to perform two switches simultaneously (these double switches are only needed in rare, well characterized cases).

3. Switch Chain for 2-Class JDM Instances. In this section we show that the switch chain defined in Subsection 2.1 is *always* rapidly mixing for JDM instances with two degree classes. The formal statement is as follows.

[4] It might be the case that the chain is always irreducible, even if $\mathcal{D}$ is not strongly stable, but this is not relevant at this point. The assumption of strong stability allows for a shortcut in the proof of irreducibility.

THEOREM 3.1. *Let $\mathcal{D}$ be the family of instances of the joint degree matrix model with two degree classes. Then the switch chain is rapidly mixing for instances in $\mathcal{D}$.*

The proof of Theorem 3.1 consists of three parts. In analogy to the approach taken in [4], we first analyze a simpler Markov chain, called the *hinge flip chain*, that adds and removes (at most) one edge at a time (see Figure 3). The hinge flip chain might slightly violate the degree constraints, as well as the joint degree constraints. The definition of *strong stability* used in [4] for degree sequences is appropriately adjusted here to account for both deviations from the original requirements. In Section 3.2 we show that instances of the JDM model with two degree classes are indeed strongly stable under this definition. Finally, we use a similar embedding argument as in [4] to argue that the (restricted) switch chain is rapidly mixing. Next, we give a more detailed description of these three parts.

Proof overview. The first step of the proof is to show that the hinge flip chain defined on a *strict superset* of the state space mixes rapidly for strongly stable instances. This is done in Section 3.1. The auxiliary states have the property that the joint degree constraint may only be violated slightly, by an additive value of one to be precise. In order to overcome the difficulties that arise due to the fact that the number of edges across the two degree classes should remain almost the same, we use ideas introduced by Bhatnagar et al. [5] for uniformly sampling bichromatic matchings. In particular, in the circuit processing part of the proof, we process a circuit at multiple places *simultaneously* in case there is only one circuit in the canonical decomposition of a pairing; or we process multiple circuits *simultaneously* in case the decomposition yields multiple circuits. At the core of this approach lies a variant of the *mountain climbing problem* [21, 31]. In our case the analysis is more involved than that of [5], and we therefore use different arguments in various parts of the proof.

It is interesting to note that the analysis of the hinge flip chain is not carried out in the JDM model but in the more general partition adjacency matrix model (Section 2.2). The difference from the JDM model is that in each class V_i the nodes need not have the same degree but rather follow a given degree sequence of size $|V_i|$. Given that small deviations from the prescribed degrees cannot be directly handled—by definition—by the JDM model, the PAM model is indeed a more natural choice for this step.

Next, in Section 3.2, we show that for any JDM instance, any graph in the state space of the hinge flip chain (i.e., graphs that satisfy or *almost* satisfy the joint degree requirements) can be transformed to a realization of the original instance within 11 hinge flips at most. That is, the set of JDM instances is a strongly stable family of instances of the PAM model and thus the hinge flip chain mixes rapidly for JDM instances.

The final step is an embedding argument for transforming the efficient flow for the hinge flip chain to an efficient flow for the switch chain similar to that in [4]. This step is presented in Section 3.3.

3.1. Rapid Mixing of the Hinge Flip Chain. In this section we show that the hinge flip chain is rapidly mixing for strongly stable tuples (Theorem 3.2). We prove Theorem 3.2 based on ideas introduced in [5]. Throughout this section we always consider tuples $(\gamma, \boldsymbol{d})$ coming from strongly stable families.

THEOREM 3.2. *Let $\mathcal{D}$ be a strongly stable family of tuples $(\gamma, \boldsymbol{d})$ with respect to some constant k. Then there exist polynomials $p(n)$ and $r(n)$ such that for any $(\gamma, \boldsymbol{d}) \in \mathcal{D}$, with $\boldsymbol{d} = (d_1, \ldots, d_n)$, there exists an efficient multicommodity flow f for the hinge* *flip chain* $\mathcal{M}(\gamma, \boldsymbol{d})$ *on* $\mathcal{G}'(\gamma, \boldsymbol{d})$ *satisfying* $\max_e f(e) \leq p(n)/|\mathcal{G}'(\gamma, \boldsymbol{d})|$ *and* $\ell(f) \leq r(n)$. *Hence, the hinge flip chain* $\mathcal{M}(\gamma, \boldsymbol{d})$ *is rapidly mixing for families of strongly stable tuples.*

We will use the following lemma in order to simplify the proof of Theorem 3.2.

LEMMA 3.3. *Let* f' *be a flow that routes* $1/|\mathcal{G}'(\gamma, \boldsymbol{d})|^2$ *units of flow between any pair of states in* $\mathcal{G}(\gamma, \boldsymbol{d})$ *in the chain* $\mathcal{M}(\gamma, \boldsymbol{d})$*, so that* $f'(e) \leq b/|\mathcal{G}'(\gamma, \boldsymbol{d})|$ *for all* e *in the state space graph of* $\mathcal{M}(\gamma, \boldsymbol{d})$*. Then* f' *can be extended to a flow* f *that routes* $1/|\mathcal{G}'(\gamma, \boldsymbol{d})|^2$ *units of flow between any pair of states in* $\mathcal{G}'(\gamma, \boldsymbol{d})$ *with the property that for all* e

$$f(e) \leq q(n) \frac{b}{|\mathcal{G}'(\gamma, \boldsymbol{d})|}, \tag{3.1}$$

where $q(\cdot)$ *is a polynomial whose degree only depends on* $k(\gamma, \boldsymbol{d})$ $(\leq k)$*. Moreover,* $\ell(f) \leq \ell(f') + 2k(\gamma, \boldsymbol{d})$.

Proof. We extend the flow f' to f as follows. For any $G \in \mathcal{G}'(\gamma, \boldsymbol{d}) \setminus \mathcal{G}(\gamma, \boldsymbol{d})$ fix some $\phi(G) \in \mathcal{G}(\gamma, \boldsymbol{d})$ within distance k of G (which exists by assumption of strong stability), and fix some path in the state space graph from G to $\phi(G)$ of length at most k. Moreover, define $\phi(H) = H$ for all $H \in \mathcal{G}(\gamma, \boldsymbol{d})$. The flow between G and any given $G' \in \mathcal{G}'(\gamma, \boldsymbol{d})$ is now sent as follows.

First route $1/|\mathcal{G}'(\gamma, \boldsymbol{d})|^2$ units of flow from G to $\phi(G)$ over the fixed path from G to $\phi(G)$. Then use the flow-carrying paths used to send $1/|\mathcal{G}'(\gamma, \boldsymbol{d})|^2$ units of flow between $\phi(G)$ and $\phi(G')$ as in the flow f' (note that in general multiple paths might be used for this in the flow f'). Finally, use the reverse of the fixed path from G' to $\phi(G')$ to route $1/|\mathcal{G}'(\gamma, \boldsymbol{d})|^2$ from $\phi(G')$ to G'. For any $H \in \mathcal{G}(\gamma, \boldsymbol{d})$, we have $|\phi^{-1}(H)| \leq \text{poly}(n^k)$, as the in- and out-degrees of the nodes in the state space graph of $\mathcal{M}(\gamma, \boldsymbol{d})$ are polynomially bounded. It can then be shown that this extension of f', yielding the flow f, only gives an additional term of at most $\text{poly}(n^k)\frac{b}{|\mathcal{G}'(\gamma, \boldsymbol{d})|}$ to the congestion of every edge in the state space graph of the chain $\mathcal{M}(\gamma, \boldsymbol{d})$ in the flow f'. Hence, the extended flow f satisfies (3.1) for some appropriately chosen polynomial $q(n)$. ∎

Because of Lemma 3.3 it now suffices to show that there exists a flow f' that routes $1/|\mathcal{G}'(\gamma, \boldsymbol{d})|^2$ units of flow between any two pair of states in $\mathcal{G}(\gamma, \boldsymbol{d})$, in the state space graph of the chain $\mathcal{M}(\gamma, \boldsymbol{d})$, with the property that $f'(e) \leq p(n)/|\mathcal{G}'(\gamma, \boldsymbol{d})|$, and $\ell(f') \leq q(n)$ for some polynomials $p(\cdot), q(\cdot)$ whose degrees may only depend on $k(\gamma, \boldsymbol{d})$. Note that f' is not a feasible multi-commodity flow as defined in Section 2, but should rather be interpreted as an intermediate flow. The proof of Theorem 3.2 will consist of multiple parts following, conceptually, the proof template in [7] developed for proving rapid mixing of the switch chain for regular graphs. The main difference is that for the so-called *canonical paths* between states we rely on ideas introduced in [5].

3.1.1. Canonical Paths. We first introduce some basic terminology similar to that in [7]. Let V be a set of labeled nodes and let $\prec_V$ be a total ordering of the nodes. Let $\prec_E$ be the total order on the set $\{\{v, w\} : v, w \in V\}$ given by lexicographic comparison based on $\prec_V$, and let $\prec_{\mathcal{C}}$ be a total order on all *circuits* on the complete graph K_V, i.e., $\prec_{\mathcal{C}}$ is a total order on the closed walks in K_V that visit every edge at most once. We fix for every circuit one of its nodes where the walk begins and ends.

For given $G, G \in \mathcal{G}(\gamma, \boldsymbol{d})$, let $H = G \triangle G'$ be their symmetric difference. We refer to the edges in $G \setminus G'$ as *blue*, and the edges in $G' \setminus G$ as *red*. A *pairing* of red and blue edges in H is a bijective mapping that, for each node $v \in V$, maps every red edge adjacent to v, to a blue edge adjacent to v. The set of all pairings is denoted by $\Psi(G, G')$, and, with θ_v the number of red edges adjacent to v (which is the same as the number of blue edges adjacent to v), we have $|\Psi(G, G')| = \Pi_{v \in V} \theta_v!$.

Remember that we are considering an instance of the PAM model with two classes V_1 and V_2. For a given realization $G \in \mathcal{G}(\gamma, \boldsymbol{d})$ we say that $e \in E(G)$ is a *cut* edge if it has an endpoint in both V_1 and V_2. Otherwise we say that e is an *internal* edge, as both endpoints either lie both in the class V_1 or both in class V_2.

Similar to the approach in [7], the goal is to construct for each pairing $\psi \in \Psi(G, G')$ a canonical path from G to G' that carries a fraction $|\Psi(G, G')|^{-1}$ of the total flow from G to G' in f'. For a given pairing ψ and the total order $\prec_E$ given above, we first decompose H into the edge-disjoint union of circuits in a canonical way. We start with the lexicographically least edge w_0w_1 in E_H and follow the pairing ψ until we reach the edge w_kw_0 that was paired with w_0w_1. This defines the circuit C_1 (which is indeed a closed walk). If $C_1 = E_H$, we are done. Otherwise, we pick the lexicographically least edge in $H \setminus C_1$ and repeat this procedure. We continue generating circuits until $E_H = C_1 \cup \cdots \cup C_s$. Note that all circuits have even length and alternate between red and blue edges, and that they are pairwise edge-disjoint.

We form a path from G to G' in the state space graph of the chain $\mathcal{M}(\gamma, \boldsymbol{d})$ by changing the blue edges of G into the red edges of G' using hinge flip operations. For certain pairings this can be done in a straightforward way, but in general this is not the case. As a warm-up, we first consider a simple case (this case essentially describes how we would process the circuits in case there is only one class).

Warm-up example. If for every i, the circuit C_i exclusively consists of internal edges, only within V_1 or only within V_2, or exclusively of cut edges, then circuits can be processed according to the ordering $\prec_{\mathcal{C}}$ as follows. Let $C = x_0x_1x_2 \dots x_qx_0$ be a circuit, and assume w.l.o.g. that x_0x_1 is the lexicographically smallest blue edge adjacent to the starting node x_0 of the circuit. The processing of C now consists of performing a sequence of hinge flips on the ordered pairs (x_{i-1}, x_i, x_{i+1}) for $i = 1, \dots, q$ with the convention that $x_{q+1} = x_0$. This is illustrated in Figures 4, 5 and 6 for an example of C as illustrated in Figure 4 on the left. We have also indicated the degree surplus and deficit at every step. By assumption, the edges of C either are all internal edges or all cut edges. Therefore, throughout the processing of C, we never violate the constraint that there should be γ edges between the classes V_1 and V_2, and, in particular, this implies that every intermediate state is an element of $\mathcal{G}'(\gamma, \boldsymbol{d})$.

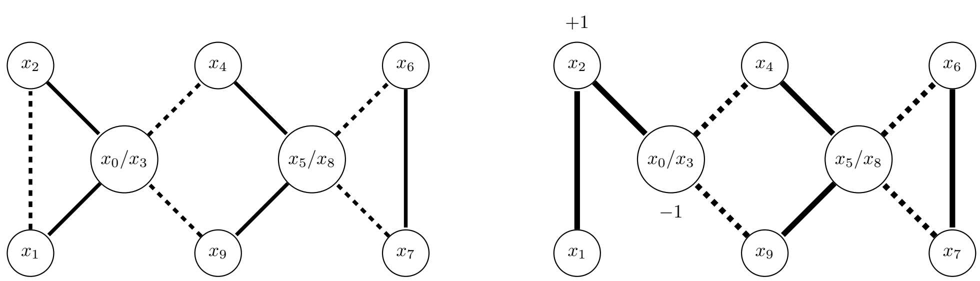

FIG. 4. *The circuit $C = x_0x_1x_2x_3x_4x_5x_6x_7x_8x_9x_0$ with $x_0 = x_3$ and $x_5 = x_8$. The blue edges are represented by the solid edges, and the red edges by the dashed edges (left). The edge x_0x_1 is removed and x_1x_2 is added (right).*

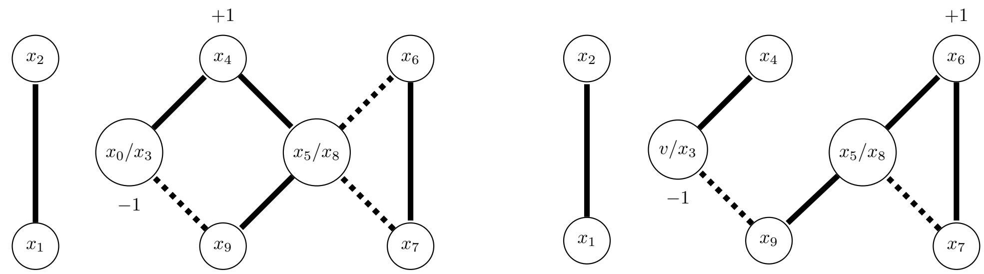

FIG. 5. *The edge x_2x_3 is removed and x_3x_4 is added (left). The edge x_4x_5 is removed and x_5x_6 is added (right).*

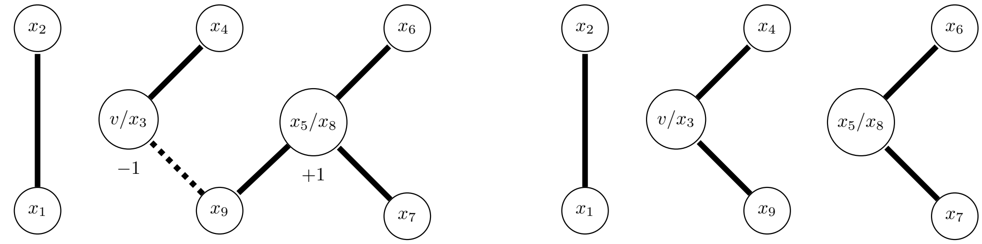

FIG. 6. *The edge x_6x_7 is removed and x_7x_8 is added (left). The edge x_8x_9 is removed and x_9x_0 is added (right).*

In general, however, it might happen that circuits contain both cut and internal edges, in which case we cannot use the circuit processing procedure explained above, as the processing of a circuit might result in a realization for which the number of edges between the classes V_1 and V_2 lies outside the set $\{\gamma - 1, \gamma, \gamma + 1\}$. The latter condition is necessary for the intermediate states in the circuit processing procedure to be elements of $\mathcal{G}'(\gamma, \boldsymbol{d})$, by definition of that set. In order to overcome the issue described above, we will use the ideas in [5], and process a circuit at multiple places *alternately* in case there is only one circuit in the canonical decomposition of a pairing, or, process multiple circuits alternately (without completing the processing of a circuit in one go) in case the decomposition yields multiple circuits. At the core of this approach lies (a variation of) the *mountain climbing problem* [21,31]. We begin with introducing this problem, and afterwards continue with the description of the circuit processing procedure, based on the solution to the mountain climbing problem.

Intermezzo: mountain climbing problem. We first introduce some notation and terminology. For non-negative integers a, b with $a+1 < b$ we define an $\{a, b\}$-*mountain* as a function $P : \{a, a+1, \ldots, b\} \to \boldsymbol{Z}_{\geq 0}$ with the properties that (i) $P(a) = P(b) = 0$; (ii) $P(i) > 0$ for all $i \in \{a + 1, \ldots, b - 1\}$; and (iii) $|P(i + 1) - P(i)| = 1$ for all $i \in \{a, \ldots, b - 1\}$. A function $P : \{a, a + 1, \ldots, b\} \to \boldsymbol{Z}_{\leq 0}$ is called an $\{a, b\}$-*valley* if the function $-P$ is an $\{a, b\}$-mountain. We subdivide a mountain into a left side $\{a, \ldots, t\}$ and right side $\{t, \ldots, b\}$ where t is the smallest integer maximizing the function P. For a valley function P, the left and right side are determined by the smallest integer t minimizing the function P.

DEFINITION 3.4. *A* traversal *of the $\{a, b\}$-mountain P is a sequence $(a, t) = (i_1, j_1), \ldots, (i_k, j_k) = (t, b)$ with the properties*

(a) $|i_r - i_{r+1}| = |j_r - j_{r+1}| = 1$,

(b) $P(i_r) + P(j_r) = P(t)$,

(c) $a \leq i_r \leq t$ *and* $t \leq j_r \leq b$,

for all $1 \leq r \leq k-1$*. We always assume that a traversal is minimal, in the sense that there is no subsequence of* $(a, t) = (i_1, j_1), \ldots, (i_k, j_k) = (t, b)$ *which is also a traversal.*

Roughly speaking, we place one person at the far left end of the mountain, and one at the first top. These persons now simultaneously traverse the mountain in such a way that the sum of their heights is always equal, and they always stay on their respective sides of the mountain that they started. The goal of the person on the left it to ascend to the top, whereas the goal of the player at the top is to descend to the far right of the mountain.

LEMMA 3.5 ([5]). *For any mountain or valley function* P *on* $\{a, \ldots, b\}$ *with first top* t*, there exists a traversal of* P *of length at most* $O((t-a)(b-t))$*, that can be found in time* $O((t-a)(b-t))$*.*

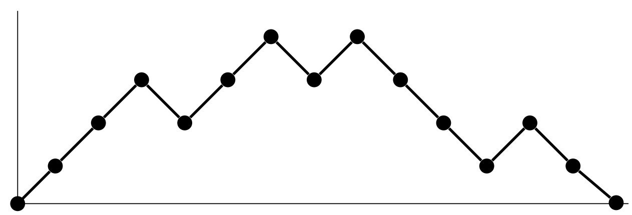

FIG. 7. *Example of a mountain function* P *on the integers in* $\{0, \ldots, 14\}$ *with the first top at* $t = 6$*. The left side of the mountain is given by* $\{0, \ldots, 6\}$ *and the right side by* $\{6, \ldots, 14\}$*. A traversal of* P *is given by the sequence* $(0, 6), (1, 7), (0, 8), (1, 9), (2, 10), (3, 11), (4, 12), (5, 13), (6, 14)$.

We finish this part with some additional notation that will be used later on. Let $P_j : \{a_j, \ldots, b_j\} \to \boldsymbol{Z}$ for $j = 1, \ldots, l$ be a collection of mountain and valley functions such that $a_1 = 0$, $b_j = a_{j+1}$ for $j = 1, \ldots, p-1$, and every P_j is either a mountain or a valley. We define the *landscape* Q of the functions $P_1, \ldots, P_l$ as the function $Q : \{0, 1, \ldots, b_l\} \to \boldsymbol{Z}$ given by $Q(i) = P_j(i)$ where $j = j(i)$ is such that $i \in \{a_j, \ldots, b_j\}$. Note that $Q(0) = Q(b_l) = 0$, and $|Q(i+1) - Q(i)| = 1$ for all $i \in \{0, \ldots, b_j - 1\}$. Moreover, for any function $R : \{0, \ldots, r\} \to \boldsymbol{Z}$ satisfying the latter two conditions, there is a unique collection of mountain and valley functions so that R is the landscape of those functions. We call functions satisfying these conditions *landscape functions.*

General case. We first partition every circuit into a collection of so-called sections, which in turn will be grouped into so-called segments. Let $C_1, \ldots, C_s$ be the canonical circuit decomposition of the symmetric difference $G \triangle G'$ for some pairing ψ, and assume w.l.o.g. that $C_i \prec_C C_j$ whenever $i < j$. We write $C_i = x_0^i x_1^i \ldots x_{q_i}^i x_0^i$ where $x_0^i x_1^i$ is the lexicographically smallest blue edge adjacent to the starting point x_0^i of the circuit C_i, and where q_i is such that C_i has $q_i + 1$ edges (and where $x_0 = x_{q_i+1}$). For any i, we define the function

$$l_i(r) = \begin{cases} -1 & \text{if } \{x_{r-2}^i, x_{r-1}^i\} \text{ is cut edge and } \{x_{r-1}^i, x_r^i\} \text{ is internal edge,} \\ 1 & \text{if } \{x_{r-2}^i, x_{r-1}^i\} \text{ is internal edge and } \{x_{r-1}^i x_r^i\} \text{ is cut edge,} \\ 0 & \text{otherwise,} \end{cases}$$

for $r = 2, 4, \ldots, q_i + 1$. The function l_i indicate what happens to the number of cut edges of a realization when we perform a hinge flip on a pair of consecutive edges $\{x^i_{r-2}, x^i_{r-1}\}$ and $\{x^i_{r-1}, x^i_r\}$ on the circuit C_i.

Decomposition into segments. We subdivide every circuit C_i into a sequence of (not necessarily closed) walks of even length, called *sections*. Let $Z_i = \{r : l_i(r) \neq 0\} = \{z_1, \dots, z_{u_i}\} \subseteq \{2, 4, \dots, q_i + 1\}$ be the set of indices that represent a change in cut edges along the circuit, where we assume that $z_1 \leq z_2 \leq \cdots \leq z_{u_i}$. We define $C_i^1 = x^i_0 x^i_1 \dots x^i_{z_1}$ and $C_i^j = x^i_{z_{j-1}} \dots x^i_{z_j}$ for $j = 2, \dots, u_i - 1$. If $l_i(q_i + 1) \neq 0$ this procedure partitions the circuit C_i completely, with $C_i^{u_i}$ being the last section. Otherwise, we define $C_i^{u_1+1} = x^i_{z_{u_i}} \dots x^i_0$ as the final section, which is the remainder of the circuit C_i. We define U_i as the total number of obtained sections, which is either u_i or $u_i + 1$. Note that when $Z_i = \emptyset$, the whole circuit will form one section $C_i = C_i^1$. Also note that a section always starts with a blue edge. We extend the function l_i to sections in the following way:

$$l_i\left(C_i^j\right) = \sum_{r=z_{j-1}+2,\dots,z_j} l_i(r) = \begin{cases} -1 & \text{if } l_i(z_j) = -1, \\ 1 & \text{if } l_i(z_j) = 1, \\ 0 & \text{otherwise}, \end{cases}$$

for $j = 1, \dots, U_i$. Note that $l(C_i^j) \in \{-1, 1\}$ for $j = 1, \dots, u_i$, and zero for $j = u_i + 1$ if this term is present. An example is given in Figure 8.

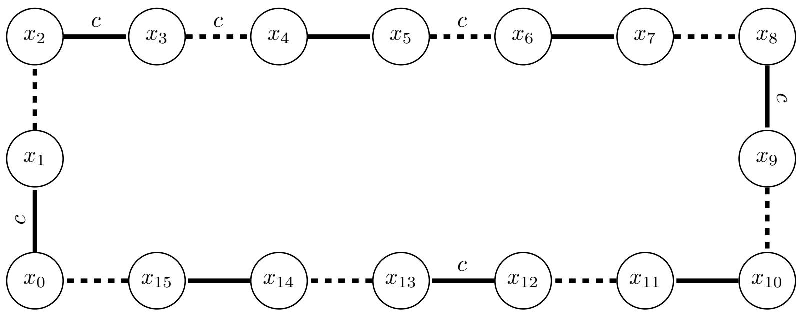

FIG. 8. *The circuit $C_1 = x_0 x_1 \dots x_{15} x_0$ with $q_1 = 15$. The blue edges are represented by the solid edges, and the red edges by the dashed edges. A label c on an edge indicates that it is a cut edge (all others are internal edges). We have $C_1^1 = x_0 x_1 x_2$ with $l_1(C_1^1) = -1$; $C_1^2 = x_2 x_3 x_4 x_5 x_6$ with $l_1(C_1^2) = 1$; $C_1^3 = x_6 x_7 x_8 x_9 x_{10}$ with $l_1(C_1^3) = -1$; $C_1^4 = x_{10} x_{11} x_{12} x_{13} x_{14}$ with $l_1(C_1^4) = -1$; and $C_1^5 = x_{14} x_{15} x_0$ with $l_1(C_1^5) = 0$ (note that $U_1 = 5$ in this example).*

We continue by grouping the union of all sections into segments in a similar flavor. For sake of readability, we rename the sections

$$C_1^1, \dots, C_1^{U_1}, C_2^1, \dots, C_2^{U_2}, \dots, C_s^1, \dots, C_s^{U_s}$$

as $D_1, \dots, D_U$ in the obvious way, where $U = \sum_{i=1}^s U_i$, and we define $l(D_k) = l_i(C_i^j)$ if C_i^j was renamed D_k. We define $W = \{k : l(D_k) \neq 0\} = \{w_1, \dots, w_B\}$ as the set of sections representing a change in cut edges along a circuit, where we assume that $w_1 \leq \cdots \leq w_B$. We define the *segment* $S_1 = (D_1, \dots, D_{w_1})$, and $S_i = (D_{w_{i-1}+1}, \dots, D_{w_i})$ for $i = 2, \dots, w_B - 1$. If $l(D_U) \neq 0$, i.e., when $w_B = U$, this procedure completely groups the collection of sections into segments. Otherwise, we redefine the last segment as $S_B = (D_{w_{B-1}+1}, \dots, D_U)$. We can extend the function l

to segments in the following way:

$$l\left(S_i\right) = \sum_{j=w_{i-1}+1}^{w_i} l(D_j) = \begin{cases} -1 & \text{if } l(D_{w_i}) = -1, \\ 1 & \text{if } l(D_{w_i}) = 1, \end{cases}$$

for $i = 1, \ldots, B-1$, and $l\left(S_B\right) = \sum_{j=w_{B-1}+1}^{U} l(D_j)$. Note that

$$l(S_i) \in \{-1, 1\} \quad \text{for } i = 1, \ldots, B, \tag{3.2}$$

unless in the special case that there is only one segment S_1 covering all circuits, then $l(S_1) = 0$. This happens, e.g., in the situation of the warm-up example.

An example of a decomposition into segments is given in Figures 11 and 12 later on. Roughly speaking, a segment is a maximal collection of edges that could be processed, using hinge flips operations as in the warm-up example, until the number of cut-edges changes. In particular, the first segment represents precisely the point up to where we could carry out the same processing steps as in the warm-up example until the number of cut edges will have changed for the first time. Note that a segment might contain sections from multiple circuits, in particular, it might consist of a final section of a circuit J_1, then some full circuits $J_2, \ldots, J_h$ (which all form a section on their own) and then the first section of some circuit J_{h+1}. The function l is then zero on the last section of J_1 and all circuits (sections) $J_2, \ldots, J_h$, and non-zero on the section of J_{h+1}.

Unwinding/rewinding of a segment. The *unwinding* of a section $D = x_f \ldots x_g$ consists of performing a number of hinge flip operations, that represent transitions in the Markov chain $\mathcal{M}'(\gamma, \boldsymbol{d})$. That is we perform a sequence of hinge flip operations replacing the (blue) edges $\{x_{r-2}, x_{r-1}\}$ by (red) edges $\{x_{r-1}, x_r\}$ for $r = f+2, \ldots, g$, in increasing order of r. Sometimes, we need to temporarily undo the unwinding of a section, in which case we perform a sequence of hinge flip operations replacing the (red) edges $\{x_{r-1}, x_r\}$ by (blue) edges $\{x_{r-2}, x_{r-1}\}$ for $r = f+2, \ldots, g$, in decreasing order of r this time. That is, we reverse the operations done during the unwinding. This is called *rewinding* a section. We say that a circuit is (currently) processed if all its sections have been unwound, and it is (currently) unprocessed if at least one section has not been unwound.

The unwinding of a segment $S_i = (D_{a_i}, \ldots, D_{a_i+1})$ consists of unwinding the sections $D_{a_i}, \ldots, D_{a_i+1}$ in increasing order. The rewinding of S_i consists of rewinding the section $D_{a_i}, \ldots, D_{a_i+1}$ in decreasing order.

Landscape processing. Remember that B is the number of segments obtained from the decomposition of circuits into segments. We define the function $P : \{0, 1, \ldots, B\} \to \boldsymbol{Z}$ by $P(0) = 0$ and $P(i) = \sum_{j=1}^{i} l(S_j)$ for $i = 1, \ldots, B$.

LEMMA 3.6. *The function P is a landscape function.*

Proof. We have to check that $P(0) = P(B) = 0$ and that $|P(i+1) - P(i)| = 1$ for all $i = 0, \ldots, B-1$, see the description of the mountain climbing problem. We have $P(0)$ by definition. Moreover, since both realizations G and G' contain γ cut edges, it holds that $P(B) = \sum_{i=1}^{B} l(S_i) = 0$. Finally, using (3.2) and the definition of P, it follows that

$$|P(i+1) - P(i)| = \left| \sum_{j=1}^{i+1} l(S_j) - \sum_{j=1}^{i} l(S_j) \right| = |l(S_i)| = 1\,,$$

for all $i = 1, \ldots, B-1$. □

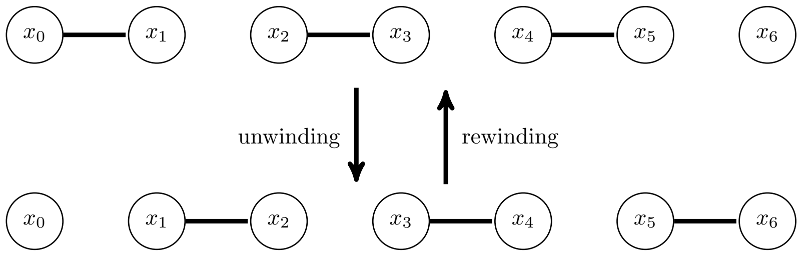

FIG. 9. *A section $D = x_0x_1 \ldots x_6$. The blue edges are represented by the solid edges. The unwinding consists of performing first a hinge flip with $\{x_0, x_1\}$ to $\{x_1, x_2\}$; then $\{x_2, x_3\}$ to $\{x_3, x_4\}$; and finally $\{x_4, x_5\}$ to $\{x_5, x_6\}$. The rewinding consist of first a hinge flip with $\{x_5, x_6\}$ to $\{x_4, x_5\}$; then $\{x_3, x_4\}$ to $\{x_2, x_3\}$; and finally $\{x_1, x_2\}$ to $\{x_0, x_1\}$*

Based on the segments $S_1, \ldots, S_B$, we define the canonical path from G to G' in the state space graph of the chain $\mathcal{G}'(\gamma, \boldsymbol{d})$ that replaces all the blue edges in $G \triangle G'$ with the red edges in $G \triangle G'$. By Lemma 3.6 we know P is a landscape function and therefore there is a unique decomposition into mountain and valley functions $P_1, \ldots, P_p$ so that P is the landscape function for this collection, where every function is of the form $P_j : \{a_j, \ldots, b_j\} \to \boldsymbol{Z}$ with $a_1 = 0, b_j = a_{j+1}$ for $j = 1, \ldots, p-1$, and $b_p = B$.[5] The *processing of a mountain/valley* P_j means that all segments $S_{a_j+1}, \ldots, S_{b_j}$ will be unwound (it might be that during this procedure segments are temporarily rewound). This processing will rely on a traversal of the mountain, see Definition 3.4. We say that the segments $S_{a_j+1}, \ldots, S_{t_j}$ are on the *left side of the mountain*, and the segments $S_{t_j+1}, \ldots, S_{b_j}$ on the *right side of the mountain*, where t_j is the first top of the mountain. Let $P = P_j$ for some j and assume that P is a mountain function. For sake of notation, we write $a = a_j$ and $b = b_j$, and $t = t_j$.

Now, fix some traversal $(a, t) = (r_1, s_1), \ldots, (r_k, s_k) = (t, b)$ of P. For $\tau = 1, \ldots, k-1$ in increasing order, do the following:

1. if $r_{\tau+1} > r_\tau$ and $s_{\tau+1} > s_\tau$: first unwind segment $S_{r_{\tau+1}}$, then unwind $S_{s_{\tau+1}}$;
2. if $r_{\tau+1} > r_\tau$ and $s_{\tau+1} < s_\tau$: first unwind segment $S_{r_{\tau+1}}$, then rewind S_{s_τ};
3. if $r_{\tau+1} < r_\tau$ and $s_{\tau+1} > s_\tau$: first rewind segment S_{r_τ}, then unwind $S_{s_{\tau+1}}$;
4. if $r_{\tau+1} < r_\tau$ and $s_{\tau+1} < s_\tau$: first rewind segment S_{r_τ}, then rewind S_{s_τ}.

This describes the processing of a mountain based on a traversal. Note that after the processing of a mountain, indeed all its segments have been unwound (see also the example worked out in the Figures 11, 12, 13 and 14). If P is a valley function, we can use essentially the same procedure performed on $-P$. The *processing of a landscape* is done by processing the mountains/valleys $P_1, \ldots, P_p$ in increasing order.

This procedure generates a sequence $G = Z_1, Z_2, \ldots, Z_l = G'$ of realizations transforming G into G' where any two consecutive realizations differ by a hinge flip operation. The following lemma shows that this sequence indeed defines a (canonical) path from G to G' in the state space graph of $\mathcal{M}(\gamma, \boldsymbol{d})$, for a given pairing ψ. This lemma is essentially the motivation for the definition of $\mathcal{G}'(\gamma, \boldsymbol{d})$.

LEMMA 3.7. *Let $Z = Z_i$ be a realization on the constructed path from G to G' for pairing ψ, with degree sequence $\boldsymbol{d}'$ and γ' cut edges. Then $(\gamma', \boldsymbol{d}')$ satisfies the*

[5] The function P_1 can be found by determining the first $j > 0$ so that $P(j) = 0$. The sign of $P(1)$ determines if it is a mountain or a valley. The remaining mountains and valleys can be found similarly.

properties (i), (ii) and (iii) defining $\mathcal{G}'(\gamma, \boldsymbol{d})$ (see Section 2.2).

Moreover, there exists a polynomial $r(\cdot)$ such that the length of any constructed (canonical) path carrying flow is at most $r(n)$.

Proof. Since hinge flip operations never change the number of edges in a graph, property (i) is clearly satisfied. Since the operations (1)–(4) given above unwind and rewind at most two segments, and by construction of the trajectories describing the traversal, the property (ii) is also satisfied. Finally, the cases (1)–(4), in combination with the second property of a traversal as in Definition 3.4, guarantee that property (iii) is satisfied. To see that all canonical paths have polynomial length, note that the traversal has polynomial length, and also every individual segment has polynomial length. ∎

3.1.2. Encoding. We continue with defining the notion of an *encoding* that will be used in the next section to bound the congestion of an edge in the state space graph of $\mathcal{M}(\gamma, \boldsymbol{d})$. Let $\tau = (Z, Z')$ be a given transition of the Markov chain. Suppose that a canonical path from G to G' for some pairing $\psi \in \Psi(G, G')$, with canonical circuit decomposition $\{C_1, \ldots, C_s\}$, uses the transition τ. We define $L_\tau(G, G') = (G \triangle G') \triangle Z$. An example is given in Figures 11, 12, 13 and 14.

LEMMA 3.8. *Given $\tau = (Z, Z')$, L, and ψ, if there is some pair (G, G') so that $L = L_\tau(G, G')$, then there are at most $\frac{1}{8}n^4$ such pairs.*

Proof. For any pair (G, G'), let P be the landscape function of this canonical path between G and G' using the transition τ, and $P_1, \ldots, P_p$ its decomposition into mountain and valley functions. Let $T_{\tau,\psi}(G, G') \in \{C_1, \ldots, C_s\}$ be the circuit containing the first node of the first segment of the right part of the mountain/valley P_j containing the transition τ. Without loss of generality, we assume that P_j is a mountain. Moreover, let Γ be the circuit containing the transition τ. If τ is used in the processing of a segment on the left side of the mountain P_j containing τ, let $\sigma_\psi(G, G')$ be the circuit containing the last node of the segment with highest index on the right side of the mountain that is currently (at the time the transition τ is performed) unwound. If τ lies on the right side of the mountain, we let $\sigma_\psi(G, G')$ be the circuit containing the last node of the segment with highest index on the left side of the mountain that is currently unwound.

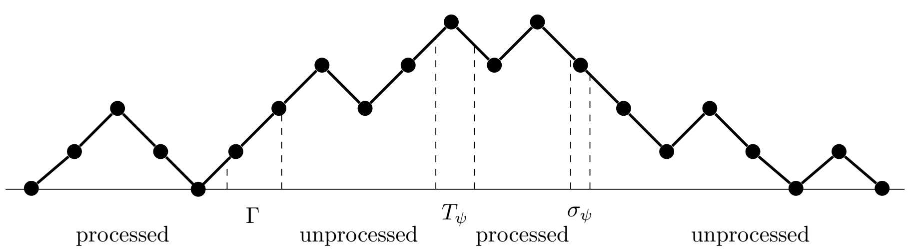

FIG. 10. *The dashed vertical lines sketch the ranges of the circuits T_ψ, σ_ψ and Γ. For every other circuit, contained in one of the four regions represented below the landscape, we know whether it has currently been processed or not.*

We claim that, given $T_\psi, \sigma_\psi \in \{C_1, \ldots, C_s\}$, it can be argued that there are at most 8 pairs (G, G') so that $T_\psi = T_\psi(G, G')$, $\sigma_\psi = \sigma_\psi(G, G')$. This can be seen as follows. Note that we can infer for all other circuits in $\{C_1, \ldots, C_s\} \setminus \{T_\psi,\ \sigma_\psi, \Gamma\}$ which edges belong to G and which to G' using the (global) circuit ordering. To see this, assume that $\Gamma \preceq_C T_\psi \preceq_C \sigma_\psi$ (the only other case $\sigma_\psi \preceq_C T_\psi \preceq_C \Gamma$ is similar).

Because the landscapes of the canonical paths always respect the circuit ordering, we know that all circuits in the canonical decomposition of ψ appearing before Γ have been unwound at this point. All circuits lying strictly between Γ and T_ψ are not unwound. The circuits strictly between T_ψ and σ_ψ again have been unwound, and finally, all circuits appearing after σ_ψ have not been unwound (see Figure 10). By comparison with Z, it is uniquely determined which edges on these circuits belong to G and which to G'. For the remaining three circuits T_ψ, σ_ψ and Γ there are for every circuit two possible configurations of the edges of G and G', since every circuit alternates between edges of G and G'.[6] Hence, there are at most $2^3 = 8$ possible pairs (G, G') with the desired properties given T_ψ and σ_ψ.

Finally, note that for any pairing ψ, there are at most $\frac{1}{4}\binom{n}{2}$ circuits in the canonical circuit decomposition $\{C_1, \dots, C_s\}$ of the pairing ψ, as every circuit has length at least four. Hence, for both T_ψ and σ_ψ there are at most $\frac{1}{4}\binom{n}{2}$ possible choices. Since Γ is uniquely determined by the transition τ, this implies that there are at most

$$8 \cdot \frac{1}{4}\binom{n}{2} \cdot \frac{1}{4}\binom{n}{2} \leq \frac{n^4}{8}$$

possible pairs (G, G') with $L = L_\tau(G, G')$. It should be noted here that a canonical path uses each transition at most once; repeated transitions would contradict the minimality of a traversal (Definition 3.4). ∎

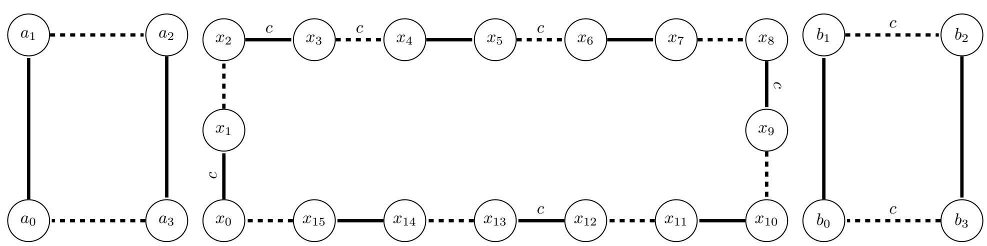

FIG. 11. *Symmetric difference $H = G \triangle G'$ where the solid edges represent the (blue) edges G and the dashed edges the (red) edges of G'. From left to right the circuit are numbered $C_1 = a_0a_1a_2a_3a_0$, $C_2 = x_0 \cdots x_{15}x_0$ and $C_3 = b_0b_1b_2b_3b_0$, and assume that this is also the order in which they are processed. Cut edges are indicated with the label c.*

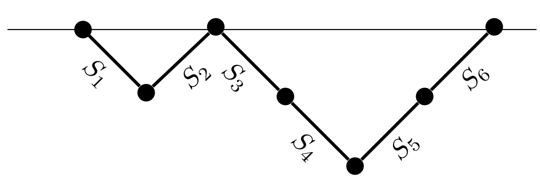

FIG. 12. *The landscape, consisting of two valleys, corresponding to the symmetric difference in Figure 11. The segments are given by $S_1 = (a_0a_1a_2a_3a_0, x_0x_1x_2)$, $S_2 = (x_2x_3x_4x_5x_6)$, $S_3 = (x_6x_7x_8x_9x_{10})$, $S_4 = (x_{10}x_{11}x_{12}x_{13}x_{14})$, $S_5 = (x_{14}x_{15}x_0, b_0b_1b_2)$, and $S_6 = (b_2b_3b_0)$.*

[6] Note that we cannot use the transition τ to infer which edges belong to G and G' on the circuit Γ, as we do not know (i.e., we do not encode) whether we are unwinding or rewinding the segment containing τ.

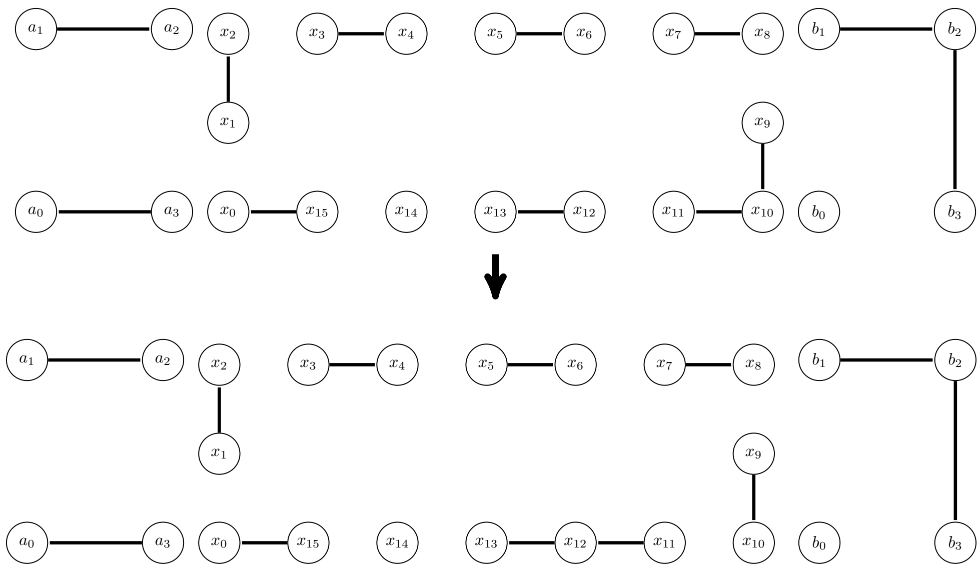

FIG. 13. *The transition $\tau = (Z, Z')$ that is the hinge flip operation that removes the edge $\{x_{10}, x_{11}\}$ and adds the edge $\{x_{11}, x_{12}\}$ as part of the unwinding of S_4. Note that the segments S_1 and as S_2, forming the first valley, have been processed already. Also, the first segment S_3 of the left part of the second valley, as well as the segment S_5 being the first segment of the right part of the second valley, have been processed already. The segment S_6 has not been processed yet. The edges in $(E(G) \cup E(G')) \setminus E(H)$ are left out.*

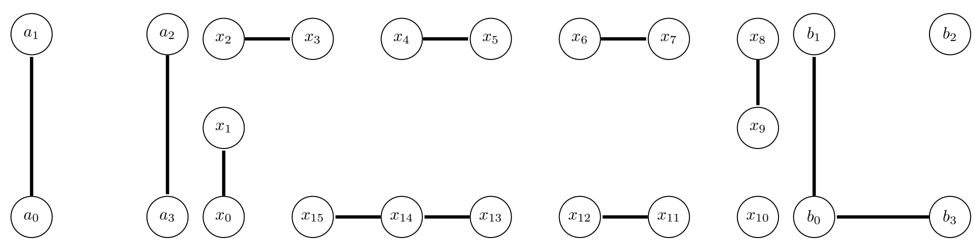

FIG. 14. *The encoding $L = L_t(G, G') = (G \triangle G') \triangle Z$ for the symmetric difference in Figure 11 and transition as in Figure 13, where again the edges in $(E(G) \cup E(G')) \setminus E(H)$ are left out.*

3.1.3. Bounding the Congestion. For a tuple (G, G', ψ), let $p_\psi(G, G')$ denote the canonical path from G to G' for pairing ψ. Let

$$\mathcal{L}_\tau = \cup_{(G,G',\psi)\in\mathcal{F}_\tau} L_\tau(G, G')$$

be the union of all distinct encodings L_τ, where $\mathcal{F}_\tau = \{(G, G', \psi) : \tau \in p_\psi(G, G')\}$ is the set of all tuples (G, G', ψ) such that the canonical path from G to G' under pairing ψ uses the transition τ. (The reason why we consider distinct encodings will become clear in the final calculation for upper bounding f'.) A crucial observation is summarized in the following lemma.

LEMMA 3.9. *If $L = L_\tau(G, G') = (G \triangle G') \triangle Z$ for transition $\tau = (Z, Z')$ used by a canonical path between G and G', then $L \in \mathcal{G}'(\gamma, \boldsymbol{d})$. This implies that*

$$|\mathcal{L}_\tau| \leq |\mathcal{G}'(\gamma, \boldsymbol{d})| \,. \tag{3.3}$$

Proof. We check that the properties (i), (ii) and (iii) defining the set $\mathcal{G}'(\gamma, \boldsymbol{d})$ (see Section 2.2) are satisfied by L. Note that $L \triangle Z = G \triangle G'$. As every individual hinge flip operations adds and removes an edge from the symmetric difference, it follows that L and Z have the same number of edges. This proves property (i). Also, if Z has a perturbation of $\alpha_v \in \{-2,-1,0,1,2\}$ (see Proposition 2.2) at node v, then L has a perturbation of $-\alpha_v$ at node v, which shows that property (ii) is satisfied for L. Finally, with $\beta \in \{-1,0,1\}$, if Z contains $\gamma - \beta$ cut edges, then L contains $\gamma + \beta$ cut edges (using implicitly that G and G' contain the same number of cut edges). This implies that property (iii) is satisfied. ∎

Moreover, with $H = G \triangle G'$ and $L = L_\tau(G, G')$, the pairing ψ has the property that it pairs up the edges of $E(H) \setminus E(L)$ and $E(H) \cap E(L)$ in such a way that for every node v each edge in $E(H) \setminus E(L)$ that is incident to v is paired up with an edge in $E(H) \cap E(L)$ that is incident to v, except for at most four pairs. These pairs occur at nodes whose degree is currently one higher or lower than it is supposed to be (in G and G'). Roughly speaking, during the processing of the symmetric difference, there are at most two circuits being processed "at the same time". For every circuit there is at most one node whose degree in the encoding is one lower than in G (and G'), and at most one node whose degree is one higher than its degree in G (see Figures 4-6 for an example). In Figure 14 these are the nodes x_{10}, x_{14}, b_0 and b_2.

Let $\Psi'(L)$ be the set of all pairings with the property described in the previous paragraph. Remember that we do not need to know G and G' in order to determine the set $H = L \triangle Z$. Also, note that not every pairing in $\Psi'(L)$ has to correspond to a tuple (G, G', ψ) for which $\tau \in p_\psi(G, G')$. Using a standard counting argument (see, e.g., [4]), we can upper bound $|\Psi'(L)|$ in terms of $|\Psi(G, G')|$. There are several cases but all of them are equally simple. Indicatively, we show the calculation for one case where there are indeed four pairs of edges that do not consist of one edge from $E(H) \setminus E(L)$ and one from $E(H) \cap E(L)$: one pair incident to node u, one incident to w, and two incident to x. The other cases are similar and the same upper bound works for all of them. Recall that θ_v denotes the number of red (or blue) edges adjacent to a node v, and that $|\Psi(G, G')| = \Pi_{v \in V} \theta_v!$. We have

$$\begin{aligned}
|\Psi'(L)| &= \frac{(\theta_u+1)!}{2} \cdot \frac{(\theta_w+1)!}{2} \cdot \frac{(\theta_x+2)!}{2\cdot 3} \cdot \prod_{v \in V \setminus \{u,w\}} \theta_v! \\
&= \frac{(\theta_u+1)(\theta_w+1)(\theta_x+1)(\theta_x+2)}{24} \cdot |\Psi(G, G')| \\
&\leq n^4 \cdot |\Psi(G, G')| \,. \qquad (3.4)
\end{aligned}$$

Putting everything together, we have

$$\begin{aligned}
|\mathcal{G}'(\gamma, \boldsymbol{d})|^2 f'(\tau) &= \sum_{(G,G')} \sum_{\psi \in \Psi(G,G')} \mathbf{1}(\tau \in p_\psi(G, G')) \cdot |\Psi(G, G')|^{-1} \\
&\leq \frac{1}{8} n^4 \sum_{L \in \mathcal{L}_\tau} \sum_{\psi' \in \Psi'(L)} |\Psi(G, G')|^{-1} && \text{(by Lemma 3.8)} \\
&\leq \frac{1}{8} n^4 \cdot n^4 \cdot \sum_{L \in \mathcal{L}_\tau} 1 && \text{(by (3.4))} \\
&\leq \frac{1}{8} n^8 \cdot |\mathcal{G}'(\gamma, \boldsymbol{d})| \,. \qquad (3.5) && \text{(by (3.3))}
\end{aligned}$$

The usage of Lemma 3.8 for the first inequality works as follows. Every tuple $(G, G', \psi) \in \mathcal{F}_\tau$ with encoding $L_\tau(G, G')$ generates a unique tuple in $\{L_\tau(G, G')\} \times \Psi'(L_\tau(G, G'))$. But since, by Lemma 3.8, there are at most $\frac{1}{8}n^4$ pairs (G, G') with $L = L_\tau(G, G')$ for given L, τ and ψ, we have that $\frac{1}{8}n^4 \sum_{L \in \mathcal{L}_\tau} |\{L\} \times \Psi'(L)| = \frac{1}{8}n^4 \sum_{L \in \mathcal{L}_\tau} \sum_{\psi' \in \Psi'(L)} 1$ is an upper bound on the number of canonical paths that contain τ.

By rearranging (3.5) we get the upper bound for f' required in Lemma 3.3. We already observed that the length of any canonical path is polynomially bounded as well. This then completes the proof of Theorem 3.2.

3.2. Strong Stability of 2-Class JDM Instances. In Section 3.1 we have shown that the hinge flip Markov chain for PAM instances with two classes is rapidly mixing on $\mathcal{G}'(\gamma, \boldsymbol{d})$ in case $(\gamma, \boldsymbol{d})$ comes from a family of strongly stable tuples. In this section we show that JDM instances with two degree classes are strongly stable. When dealing with a family of instances, even when this is not explicitly mentioned, we only consider the tuples $(\boldsymbol{c}, \boldsymbol{d})$ for which there is at least one realization.

THEOREM 3.10. *Let $\mathcal{D}$ be the family of instances of the joint degree matrix model, i.e., where for every tuple $(V_1, V_2, \gamma, \boldsymbol{d})$ it holds that $1 \leq \beta_1, \beta_2 \leq |V| - 1$, and $1 \leq \gamma \leq |V_1||V_2| - 1$, where β_1 and β_2 are the common degrees in the classes V_1 and V_2, respectively. The family $\mathcal{D}$ is strongly stable for $k = 11$, and, hence, the hinge flip chain is rapidly mixing for all tuples in $\mathcal{D}$.*

Proof. We first show that this family is strongly stable for $k = 7$. For convenience, we will work with the notation $\mathcal{G}'(\boldsymbol{c}, \boldsymbol{d})$ instead of $\mathcal{G}'(\gamma, \boldsymbol{d})$. Remember that

$$c_{ii} = \frac{1}{2}\left[\left(\sum_{j \in V_i} d_j\right) - \gamma\right],$$

for $i = 1, 2$, is the number of internal edges that V_i has in any realization in $\mathcal{G}(\gamma, \boldsymbol{d})$, and that $\gamma = c_{12} = c_{21}$. For sake of readability, we define the notion of a cancellation hinge flip. For either $i = 1$ or $i = 2$, suppose nodes $v, w \in V_i$, are such that v has a degree deficit of at least one, and w a degree surplus of at least one. Then w has a neighbor $z \in V$ that is not a neighbor of v (using that v and w have the same degree β_i). The hinge flip operation that removes the edge $\{z, w\}$ and adds the edge $\{z, v\}$ is called a *cancellation flip on v and w*. Note that the number of internal edges in V_1 and V_2 as well as the number of cut edges does not change with such an operation.[7] Moreover, we say that an edge $\{a, b\}$ is a non-edge of a realization G if $\{a, b\} \notin E(G)$.

Let $G \in \mathcal{G}'(\boldsymbol{c}, \boldsymbol{d})$ for some tuple $(\boldsymbol{c}', \boldsymbol{d}')$ as in the definition of $\mathcal{G}'(\boldsymbol{c}, \boldsymbol{d})$ at the start of Section 3.1. We first show that with at most four hinge flip operations, we can obtain a perturbed auxiliary state $G^* \in \mathcal{G}'(\boldsymbol{c}, \boldsymbol{d})$ for which its tuple $(\boldsymbol{c}^*, \boldsymbol{d}^*)$ is edge-balanced. That is, it satisfies $\boldsymbol{c}^* = \boldsymbol{c}$. Remember that the value c'_{12} uniquely determines the matrix $\boldsymbol{c}'$, and, by assumption of $\mathcal{G}'(\boldsymbol{c}, \boldsymbol{d})$, we have $c'_{12} \in \{c_{12} - 1, c_{12}, c_{12} + 1\}$. We can therefore distinguish the following cases.

- **Case 1:** $c'_{12} = c_{12} + 1$. Then, by Proposition 2.2, either $c'_{11} = c_{11} - 1$ and $c'_{22} = c_{22}$, or, $c'_{22} = c_{22} - 1$ and $c'_{11} = c_{11}$. Assume without loss of generality

[7] That is, either z lies in the other class, in which case the cancellation flip removes and adds a cut edge, or, z lies in the same class as v and w in which case an internal edge in V_i is removed and added.

that we are in the first case. Then it holds that

$$\sum_{j\in V_1} d'_j = \left(\sum_{j\in V_1} d_j\right) - 1 \quad \text{and} \quad \sum_{j\in V_2} d'_j = \left(\sum_{j\in V_2} d_j\right) + 1\,. \tag{3.6}$$

Moreover, there must be at least one node $v_2 \in V_2$ with a degree surplus (of either one or two), and there is at least one non-edge $\{a, b\}$ with both endpoints in V_1. If v_2 is adjacent to either a or b, we can perform a hinge flip to make the realization G edge-balanced, so assume this is not the case. Also, if the total deficit of a and b is -2, there must be a node in V_1 with degree surplus, otherwise (3.6) is violated. Then we can perform a cancellation flip in V_1 to remove the deficit at either a or b. Hence, we may assume without loss of generality that a does not have a degree deficit at the cost of one hinge flip.

- **Case A: v_2 has a neighbor $v_1 \in V_1$.** If v_1 has a degree surplus we can perform a cancellation flip in V_1 to remove it, which must exist by (3.6). So assume v_1 has no degree surplus. As node a has no deficit, and is not connected to v_2, whereas v_1 is, there must be some neighbor p of a which is not a neighbor of v_1. This holds since v_1 and a have the same degree β_1 in the sequence $\boldsymbol{d}$. Then the path $v_2 - v_1 - p - a - b$ alternates between edges and non-edges of G, and with two hinge flips we can obtain an edge-balanced realization in $\mathcal{G}'(\gamma, \boldsymbol{d})$. This case therefore requires at most three hinge flips in total.

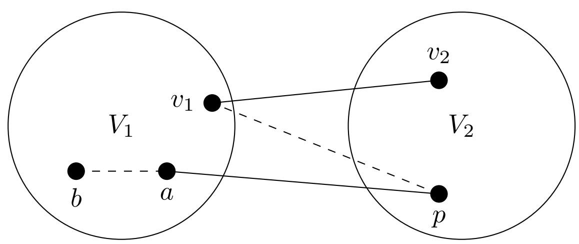

FIG. 15. *Sketch of first case with subcase A.*

- **Case B: v_2 has no neighbors in V_1.** We know that there is at least one cut edge $\{q, r\}$, with $q \in V_2$ and $r \in V_1$, in the realization G, since $c'_{12} = c_{12} + 1$. If q has a degree surplus, we are in the situation of Case A. Otherwise v_2 has a neighbor u which is not a neighbor of q, since q and v_2 have the same degree β_2 in the sequence $\boldsymbol{d}$. We can then perform the hinge flip that removes $\{v_2, u\}$ and adds $\{u, q\}$. If q now has a degree surplus, we are in Case A. Otherwise, in case this hinge flip canceled out a degree deficit at q, there must be at least one other node in V_2 with a degree surplus, because of (3.6). We can then perform the same step again, which will now result in a degree surplus at q. This is true since the node q cannot have a deficit of -2, since (3.6) would then imply that the the total degree surplus of nodes in V_2 is at least three, which violates the second property defining $\mathcal{G}'(\boldsymbol{c}, \boldsymbol{d})$. That is, we can always reduce to the situation in Case A with at most two hinge flips.

Summarizing, we can always find an edge-balanced realization G^* using at most six hinge flip operations in case $c'_{12} = c_{12} + 1$.

- **Case 2:** $c'_{12} = c_{12} - 1$. Using complementation, it can be seen that this case is similar to Case 1. That is, we consider the tuple $(\bar{\boldsymbol{c}}, \bar{\boldsymbol{d}})$ in which all nodes in V_1 have degree $|V| - \beta_1$, all nodes in V_2 have degree $|V| - \beta_2$, and where all feasible realizations have $\bar{c}_{12} = |V_1||V_2| - c_{12}$ cut edges. The case $c_{12} = c_{12} - 1$ then corresponds to the case $\bar{c}'_{12} = \bar{c}_{12} + 1$.

- **Case 3:** $c'_{12} = c_{12}$. If also $c_{11} = c'_{11}$ we are done. Otherwise, suppose that $c'_{11} = c_{11} + 1$. Then it must be that $c'_{22} = c_{22} - 1$, as $c'_{12} = c_{12}$, and it holds that

$$\sum_{j \in V_1} d'_j = \left(\sum_{j \in V_1} d_j \right) + 2 \quad \text{and} \quad \sum_{j \in V_2} d'_j = \left(\sum_{j \in V_2} d_j \right) - 2\,. \tag{3.7}$$

Then there is at least one edge $\{a, b\}$ in the realization with $a, b \in V_1$. Moreover, we may assume that a has a degree surplus. If not, then there is at least one other node u with a degree surplus because of (3.7). Performing a cancellation flip then gives the node a a degree surplus (it could not be that a had a degree deficit, as this would imply, in combination with (3.7), that the total degree surplus of nodes in V_1 is at least three).

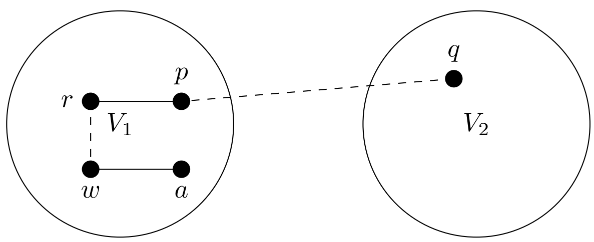

FIG. 16. *Sketch of last situation in Case 3.*

Now, if there is a non-edge of the form $\{b, v_2\}$ for some $v_2 \in V_2$, we can perform a hinge flip operation removing $\{a, b\}$ and adding $\{b, v_2\}$ in order to end up in Case 1. Otherwise, assume that b is adjacent to all $v_2 \in V_2$. As b is also adjacent to a, and a has a degree surplus of at least one,[8] it follows that $\beta_1 \geq |V_2|$. Now, by the assumption that $c_{12} \leq |V_1||V_2| - 1$, there is at least one non-edge $\{p, q\}$ with $p \in V_1$ and $q \in V_2$. As p is not adjacent to q, but has degree at least $\beta_1 \geq |V_2|$, it must be that p is adjacent to some $r \in V_1$. If r has a degree surplus, then we can perform a hinge flip that removes $\{p, r\}$ and adds $\{p, q\}$ in order to end up in the situation of Case 1. Otherwise, node a, which has a degree surplus, has some neighbor w which is not a neighbor of r. This implies the path $a - w - r - p - q$ alternates between edges and non-edges of G. Performing two hinge flips then brings us in the situation of Case 1. Overall, we can resolve this case using at most three hinge flips, or provide a reduction to Case 1 with at most three hinge flips.

The arguments above imply that with at most nine hinge flips we can obtain some $G^* \in \mathcal{G}'(\boldsymbol{c}, \boldsymbol{d})$ that is edge-balanced. (In the worst-case we need three hinge flips to

[8] That is, b can have a degree surplus of at most one. A degree surplus of two at b would only give a bound of $\beta_1 \geq |V_2| - 1$.

reduce Case 3 to Case 1, after which we need six hinge flips to resolve the latter.) This implies that

$$\sum_{j \in V_1} d_j^* = \sum_{j \in V_1} d_j \quad \text{and} \quad \sum_{j \in V_2} d_j^* = \sum_{j \in V_2} d_j \,. \tag{3.8}$$

Now, if $v \in V_1$ has a degree surplus, there must be some $w \in V_1$ that has a degree deficit, because of (3.8). We can then perform a cancellation flip to decrease the sum of the total degree deficit and degree surplus. A similar statement is true if $v \in V_2$ has a degree surplus. By performing this step at most twice, we obtain a realization $H \in \mathcal{G}(\boldsymbol{c}, \boldsymbol{d})$. That is, with at most eleven hinge flip operations in total we can transform G into a realization in $\mathcal{G}(\boldsymbol{c}, \boldsymbol{d})$. This shows that $\mathcal{D}$ is strongly stable for $k = 11$. ◻

3.3. Rapid Mixing of the Switch Chain. In this section we will use an embedding argument similar to that in [4] to show that the restricted switch chain is rapidly mixing in the case that all degrees in each class are the same, i.e., for instances that are essentially JDM instances with two degree classes.

While the restricted switch chain is known to be irreducible for the instances of the JDM model [1, 11], in general this is not true [15]. To the best of our knowledge, there is no clear understanding for which pairs $\boldsymbol{c}$ and $\boldsymbol{d}$ it is irreducible in general. Nevertheless, we present the following meta-result for the rapid mixing of the switch chain, which in particular applies when the degrees are the same within each component (Theorem 3.1).

THEOREM 3.11. *Let $\mathcal{D}$ be a strongly stable family of tuples $(\gamma, \boldsymbol{d})$ with respect to some constant k, and suppose there exists a function $p_0 : \boldsymbol{N} \to \boldsymbol{N}$ with the property that, for any fixed $x \in \boldsymbol{N}$: if $(\gamma, \boldsymbol{d}) \in \mathcal{D}$, and $G, G' \in \mathcal{G}(\gamma, \boldsymbol{d})$ so that $|E(G) \triangle E(G')| \leq x$, the switch-distance satisfies $dist_{\mathcal{G}(\gamma,\boldsymbol{d})}(G, G') \leq p_0(x)$. Then the switch chain is rapidly mixing for all tuples in the family $\mathcal{D}$ with respect to the uniform stationary distribution over $\mathcal{G}(\gamma, \boldsymbol{d})$.*

Proof. First note that by definition of the function p_0, the switch chain is irreducible. It is also not hard to see that the switch chain is aperiodic and symmetric as well. This implies that it has a unique stationary distribution which is the uniform distribution over $\mathcal{G}(\gamma, \boldsymbol{d})$. In the remainder we will use $\boldsymbol{G}_h$ to denote the state space graph of the hinge flip chain with node set $\mathcal{G}'(\gamma, \boldsymbol{d})$, and $\boldsymbol{G}_{sw}$ to denote that of the switch chain with node set $\mathcal{G}(\gamma, \boldsymbol{d})$.

By assumption of strong stability, we know that the hinge flip chain $\mathcal{M}(\gamma, \boldsymbol{d})$ is rapidly mixing. In particular, from the proof of Theorem 3.2, we know there exists a flow f' that routes $1/|\mathcal{G}'(\gamma, \boldsymbol{d})|^2$ units of flow between any pair of states in $\boldsymbol{G}_h$, with the property that $f'(e) \leq p(n)/|\mathcal{G}'(\gamma, \boldsymbol{d})|$, and $\ell(f') \leq q(n)$, for some polynomials $p(\cdot), q(\cdot)$ whose degrees may only depend on $k = k(\gamma, \boldsymbol{d})$.

For the switch chain we are interested in efficiently routing $1/|\mathcal{G}(\gamma, \boldsymbol{d})|^2$ units of flow between any two states in $\boldsymbol{G}_{sw}$. We therefore first modify f' into a flow that routes $1/|\mathcal{G}(\gamma, \boldsymbol{d})|^2$ units of flow between any two states in $\mathcal{G}(\gamma, \boldsymbol{d})$ in the state space graph $\boldsymbol{G}_h$ (which will then be transformed into the desired flow in $\boldsymbol{G}_{sw}$).

We first simply remove all flow from f' in $\boldsymbol{G}_h$ routed between states (G, G') where at least one of G or G' is an element of $\mathcal{G}'(\gamma, \boldsymbol{d})$. We call the resulting intermediate flow g'. Note that the removal of this flow from f' can only result in a lower congestion for g', so in particular we have $g'(e) \leq f'(e)$ for every edge in $E(\boldsymbol{G}_h)$. Next, because of strong stability, it follows directly that

$$\frac{|\mathcal{G}'(\gamma, \boldsymbol{d})|}{|\mathcal{G}(\gamma, \boldsymbol{d})|} \leq t_1(n)\,,$$

for some polynomial $t_1(n)$ (the leading exponent of which depends on k). It then follows that if we route $1/|\mathcal{G}(\gamma, \boldsymbol{d})|^2$ units of flow over every path in the flow g', instead of $1/|\mathcal{G}'(\gamma, \boldsymbol{d})|^2$, the resulting flow g satisfies

$$g(e) \leq t_1(n) \frac{1}{|\mathcal{G}(\gamma, \boldsymbol{d})|}\,.$$

The problem is that the flow g might still route flow through states in $\mathcal{G}'(\gamma, \boldsymbol{d}) \backslash \mathcal{G}(\gamma, \boldsymbol{d})$, which are not a part of $\boldsymbol{G}_{sw}$.

In order to address this problem, the idea is to "merge" every auxiliary state in $\mathcal{G}'(\gamma, \boldsymbol{d}) \setminus \mathcal{G}(\gamma, \boldsymbol{d})$ with some node in $\mathcal{G}(\gamma, \boldsymbol{d})$ that is only a small number of hinge flips away from it (which is possible because of strong stability). This results in a graph with node set $\mathcal{G}(\gamma, \boldsymbol{d})$ but with edges possibly between states that are not connected by a switch operation (and, hence, not present in $\boldsymbol{G}_{sw}$). In order to overcome this issue, we will show that the flow on such an 'illegal' edge can be rerouted over a short detour consisting only of edges present in $\boldsymbol{G}_{sw}$. For this we will use the function p_0 (whose definition precisely guarantees that this is possible).

We let ϕ define a function that maps every auxiliary state $G' \in \mathcal{G}'(\gamma, \boldsymbol{d}) \setminus \mathcal{G}(\gamma, \boldsymbol{d})$ to some $G = \phi(G') \in \mathcal{G}(\gamma, \boldsymbol{d})$ that is at most k hinge flips away from G'. For every $G \in \mathcal{G}(\gamma, \boldsymbol{d})$ we then merge all nodes in $\phi^{-1}(G)$ into one super node identified with G. Self-loops in the resulting graph are removed and any parallel edges are merged into one edge, and in particular, all flow of g from the parallel edges is routed over the single edge. We call this graph, with node set $\mathcal{G}(\gamma, \boldsymbol{d})$, $\boldsymbol{H}$. Because of the definition of ϕ, the above procedure gives rise to a new flow $\bar{g}$ in $\boldsymbol{H}$ that routes $1/|\mathcal{G}(\gamma, \boldsymbol{d})|^2$ units of flow between any two states in the resulting graph of the node set $\mathcal{G}(\gamma, \boldsymbol{d})$ with the property that

$$\bar{g}(e) \leq t_2(n) \frac{1}{|\mathcal{G}(\gamma, \boldsymbol{d})|}\,,$$

for some polynomial t_2 for every edge in $\boldsymbol{H}$. The final issue that we have to resolve is that $\boldsymbol{H}$ contains edges between states that are not connected by a switch operation. However, because ϕ maps every auxiliary states to a state in $\mathcal{G}(\gamma, \boldsymbol{d})$ close to it, every illegal edge has the property that its endpoints have a symmetric difference which is upper bounded by an expression $x = x(k)$ linear in k. The assumed function p_0 then implies that there exists a short path in $\boldsymbol{G}_{sw}$ over which we can reroute the flow of this illegal edge. Because this rerouting is done on a local level, it can be shown that the congestion of the resulting flow f blows up by at most a polynomial factor compared to $\bar{g}$. Note that f is the desired efficient multi-commodity flow for the switch Markov chain. ∎

We conclude this section with the proof of Theorem 3.1.

Proof of Theorem 3.1. Strong stability was shown in the previous section in Theorem 3.10. Moreover, from the proof of Lemma 7 in [1] it follows that for any two graphs $H, H' \in \mathcal{G}(\gamma, \boldsymbol{d})$, H can be transformed into H' using at most $\frac{3}{2}|E(H) \triangle E(H')|$ switches of the restricted switch chain. That is, we may take $p_0(x) = \frac{3}{2}x$. Then the statement follows from Theorem 3.11. ∎

4. Rapid Mixing of the Hinge Flip Chain for Certain Sparse Regimes. We next show that certain families of sparse instances are strongly stable as well. Sparsity here refers to the fact that the maximum degree in a class is significantly smaller than the size of the class, as well as the fact that the number of cut edges is (much) smaller than the total number of edges in a graphical realization.

THEOREM 4.1 (Sparse irregular families). *Let $0 < \alpha < 1/2$ be fixed, and let $\mathcal{D}_\alpha$ be the family of tuples $(V_1, V_2, \boldsymbol{c}, \boldsymbol{d})$ for which*

(i) $|V_1|, |V_2| \geq \alpha n$,

(ii) $2 \leq d_i \leq \sqrt{\frac{\alpha n}{4}}$ *for $i \in V_1 \cup V_2$, and,*

(iii) $1 \leq c_{12} \leq \frac{\alpha n}{2}$.

The class $\mathcal{D}_\alpha$ is strongly stable for $k = 11$, and, hence, the hinge flip chain is rapidly mixing for all tuples in $\mathcal{D}_\alpha$.

Proof. We proceed in a similar fashion as the proof of Corollary 3.10. We will use the notion of an alternating path. For a given graph $H = (W, E)$, an alternating path $(x_1, \ldots, x_q)$ is an odd sequence of nodes so that $\{x_i, x_{i+1}\} \in E$ for i even, and $\{x_i, x_{i+1}\} \notin E$ for i odd.

LEMMA 4.2 (following from [22]). *Let $\boldsymbol{\delta} = (\delta_1, \ldots, \delta_r)$ be a degree sequence with $1 \leq \delta_i \leq \sqrt{r/2}$ for all $i = 1, \ldots, r$. Fix $x, y \in [r]$ and let $H = ([r], E)$ be a graphical realization of the degree sequence $\boldsymbol{\delta}'$ where $\delta'_x = \delta_x + 1$, $\delta'_y = \delta_y - 1$ and $\delta'_i = \delta_i$ for all $i \in [r] \setminus \{x, y\}$. Then there exists an alternating path of length at most four starting at x and ending at y.*

Now, let $G \in \mathcal{G}'(\boldsymbol{c}, \boldsymbol{d})$ for some tuple $(\boldsymbol{c}', \boldsymbol{d}')$ as in the definition of $\mathcal{G}'(\boldsymbol{c}, \boldsymbol{d})$ at the start of Section 2.2. We show that with at most five hinge flip operations, we can transform G into a perturbed auxiliary state $G^* \in \mathcal{G}'(\boldsymbol{c}, \boldsymbol{d})$ for which its tuple $(\boldsymbol{c}^*, \boldsymbol{d}^*)$ is edge-balanced.

- **Case 1:** $c'_{12} = c_{12} + 1$. Then, by Proposition 2.2, either $c'_{11} = c_{11} - 1$ and $c'_{22} = c_{22}$, or, $c'_{22} = c_{22} - 1$ and $c'_{11} = c_{11}$. Assume without loss of generality that we are in the first case. Then it holds that

$$\sum_{j \in V_1} d'_j = \left(\sum_{j \in V_1} d_j \right) - 1 \quad \text{and} \quad \sum_{j \in V_2} d'_j = \left(\sum_{j \in V_2} d_j \right) + 1 . \tag{4.1}$$

  Moreover, there must be at least one node $v_2 \in V_2$ with a degree surplus (of either one or two).

  - **Case A: v_2 has a neighbor $v_1 \in V_1$.** By assumptions i) and ii) it must be that there is some $b \in V_1$ so that v_1 is not a neighbor of b. Then we can perform the hinge flip that removes $\{v_2, v_1\}$ and adds $\{v_1, b\}$, resulting in an edge-balanced realization.

  - **Case B: v_2 has no neighbors in V_1.** Since $c_{12} \geq 1$ by assumption iii), there is some edge $\{a, b\}$ in G with $a \in V_1$ and $b \in V_2 \setminus \{v_2\}$. We consider the induced subgraph H on the nodes in V_2 and use $\boldsymbol{\delta}'$ to denote its degree sequence. We next apply Lemma 4.2 with $v_2 = x$ and $b = y$. To see that this is possible, note that $c_{12} \leq \alpha n/2$ by assumption iii). This implies, in combination with assumption i) that at least $\alpha n/2$ nodes in V_1 have degree at least one in H. Thus, we may apply Lemma

4.2 with $r = |V(H)| \geq \alpha n/2$, since $d_i \leq \sqrt{\alpha n/4}$ by assumption $ii)$ (which is less or equal than $\sqrt{r/2}$ if $r \geq \alpha n/2$) . Hence, there exists an alternating path of length at most four starting at v_2 and ending at b. Then by performing two hinge flips, resulting in the removal of $\{v_2, f\}, \{g, h\}$ and addition of $\{f, g\}$, $\{h, b\}$, we are in the situation of Case A where b now plays the role of v_2.

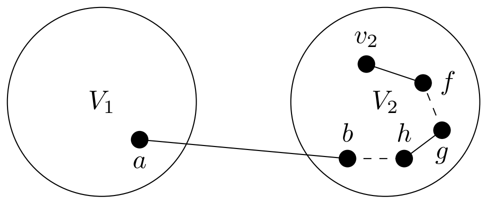

FIG. 17. *Sketch of subcase B in Case 1.*

Overall, in this case we can reach an edge-balanced realization with at most three hinge flips.

- **Case 2:** $c'_{12} = c_{12} - 1$. Suppose without loss of generality that $c'_{11} = c_{11} + 1$ and $c'_{22} = c_{22}$, and note that there must be at least one node $v_1 \in V_1$ with a degree surplus. If v_1 has a neighbor in V_1 we can perform a hinge flip operation to obtain an edge-balanced realization as desired (as this neighbor has at least one non-neighbor in V_2 by assumptions $i)$ and $ii)$). Therefore, assume that all neighbors of v_1 lie in V_2. Pick some neighbor $b \in V_2$ of v_1. Since all nodes in V_1 have degree at least one, and $c_{12} \leq \alpha n/2$, it must be that b is not adjacent to some node $a \in V_1$ that has degree at least one in the induced subgraph H on the nodes of V_1, as $d_b \leq \sqrt{\alpha n/4}$ and $|V(H)| \geq \alpha n/2$. Let c be some neighbor of a in V_1 that exists by assumption that a has degree at least one in H. Also, c is not adjacent to some $d \in V_2$ for similar reasons as that b was not adjacent to a. This means that with two hinge flip operations we can obtain an edge-balanced realization.

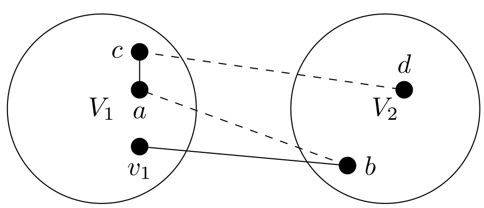

FIG. 18. *Sketch of Case 2.*

- **Case 3:** $c'_{12} = c_{12}$. If also $c_{11} = c'_{11}$ we are done. Otherwise, suppose that $c'_{11} = c_{11} + 1$. Then it must be that $c'_{22} = c_{22} - 1$, as $c'_{12} = c_{12}$, and it holds that

$$\sum_{j \in V_1} d'_j = \left(\sum_{j \in V_1} d_j \right) + 2 \quad \text{and} \quad \sum_{j \in V_2} d'_j = \left(\sum_{j \in V_2} d_j \right) - 2 \,. \tag{4.2}$$

There is at least one node $v_1 \in V_1$ with a degree surplus. If v_1 has a neighbor $a \in V_1$, which in turn has a non-neighbor $b \in V_2$ by assumptions $i)$ and $ii)$, then we can reduce to the situation of Case 1 by performing a hinge flip removing $\{v_1, a\}$ and adding $\{a, b\}$. Therefore, assume that all neighbors of v_1 are in V_2. We can then pick some neighbor $b \in V_2$ and perform a similar step as in Case 2 to find an alternating path (v_1, b, a, c) with $v_1, a, c \in V_1$ and $b \in V_2$. Then we can perform two hinge flip operations to reduce to Case 1 again. Overall, we can reduce this case to Case 1 with at most two hinge flips.

We have shown that with at most five hinge flips we can always obtain some $G^* \in \mathcal{G}'(\boldsymbol{c}, \boldsymbol{d})$ that is edge-balanced (in the worst case by using two hinge flips in Case 3 to reduce to Case 1 that requires three additional hinge flips). This implies that

$$\sum_{j \in V_1} d_j^* = \sum_{j \in V_1} d_j \quad \text{and} \quad \sum_{j \in V_2} d_j^* = \sum_{j \in V_2} d_j \, . \tag{4.3}$$

Now, if $v \in V_1$ has a degree surplus, there must be some $w \in V_1$ that has a degree deficit, because of (4.3). Moreover, if all neighbors of v are in V_2, we can transfer the degree deficit to some node with degree at least one in the subgraph induced on V_1 at the cost of one hinge flip operation (similar as the analysis in Case 2). That is, we may assume that v has a neighbor in V_1. Then, using similar arguments as in Case 1.B, it follows that there exists an alternating path from v to w in V_1, which allows us to decrease the total degree surplus/deficit using two hinge flip operations. In total this procedure needs at most three hinge flips. This can be repeated to obtain a feasible realization $H \in \mathcal{G}(\gamma, \boldsymbol{d})$ at the cost of another three hinge flips.

Overall, we obtain strong stability with $k = 5 + 3 + 3 = 11$. ∎

In particular, Theorem 4.1 directly implies the following which, to the best of our knowledge, is the first result of its kind for the PAM model.

Corollary 4.3. *Let $\mathcal{D}_\alpha$ be as in Theorem 4.1. Then there is an fully polynomial almost uniform generator for tuples in the family $\mathcal{D}_\alpha$.*

5. Conclusion. We have shown rapid mixing of the restricted switch chain for sampling realizations of a given joint degree matrix instance with two degree classes. While this is the first result of its kind, finding a polynomial time sampling algorithm for the general case remains open, either by showing rapid mixing of the switch Markov chain or any other method for that matter. Although our proof breaks down for more than two classes, we believe that our high level approach can facilitate progress on the problem. A key missing ingredient here would be a multi-dimensional substitute of the mountain climbing problem. As a byproduct of our main result, we also obtain the first sampling results for sparse PAM instances with two partition classes. However, it seems that we are a long way from constructing polynomial-time sampling algorithms for general PAM instances. For three or more classes it is not even known whether we can efficiently construct a realization (or decide that none exists).

Acknowledgments. We are grateful to Péter Erdős, Tamás Mezei and István Miklós for their useful comments, as well as to the anonymous SODA 2019 reviewers.

REFERENCES

[1] G. Amanatidis, B. Green, and M. Mihail, *Graphic realizations of joint-degree matrices*, CoRR, abs/1509.07076 (2008), https://arxiv.org/abs/1509.07076.

[2] G. Amanatidis, B. Green, and M. Mihail, *Connected realizations of joint-degree matrices*, Discret. Appl. Math., 250 (2018), pp. 65–74.

[3] G. Amanatidis and P. Kleer, *Rapid mixing of the switch Markov chain for strongly stable degree sequences and 2-class joint degree matrices*, in Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, 2019, pp. 966–985.

[4] G. Amanatidis and P. Kleer, *Rapid mixing of the switch Markov chain for strongly stable degree sequences*, To appear in Random Structures and Algorithms, (2020).

[5] N. Bhatnagar, D. Randall, V. V. Vazirani, and E. Vigoda, *Random bichromatic matchings*, Algorithmica, 50 (2008), pp. 418–445.

[6] C. J. Carstens and P. Kleer, *Speeding up switch Markov chains for sampling bipartite graphs with given degree sequence*, in Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2018, vol. 116 of LIPIcs, Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018, pp. 36:1–36:18.

[7] C. Cooper, M. E. Dyer, and C. S. Greenhill, *Sampling regular graphs and a peer-to-peer network*, Combinatorics, Probability & Computing, 16 (2007), pp. 557–593.

[8] C. Cooper, M. E. Dyer, and C. S. Greenhill, *Corrigendum: Sampling regular graphs and a peer-to-peer network*, CoRR, abs/1203.6111 (2012), https://arxiv.org/abs/1203.6111.

[9] C. Cooper, M. E. Dyer, C. S. Greenhill, and A. J. Handley, *The flip Markov chain for connected regular graphs*, Discrete Applied Mathematics, 254 (2018), pp. 56–79.

[10] É. Czabarka, *Partition adjacency matrices in network modeling*, Technical report on GRAPHS project FA9550-12-1-040, (2014).

[11] É. Czabarka, A. Dutle, P. L. Erdös, and I. Miklós, *On realizations of a joint degree matrix*, Discrete Applied Mathematics, 181 (2015), pp. 283–288.

[12] É. Czabarka, L. A. Szekely, Z. Toroczkai, and S. Walker, *An algebraic Monte-Carlo algorithm for the Partition Adjacency Matrix realization problem*, CoRR, abs/1708.08242 (2017), https://arxiv.org/abs/1708.08242.

[13] M. Dyer, M. Jerrum, and H. Müller, *On the switch Markov chain for perfect matchings*, Journal of the ACM, 64 (2017), pp. 12:1–12:33.

[14] P. L. Erdős, C. S. Greenhill, T. R. Mezei, I. Miklós, D. Soltész, and L. Soukup, *The mixing time of the switch Markov chains: a unified approach*, CoRR, abs/1903.06600 (2019), https://arxiv.org/abs/1903.06600.

[15] P. L. Erdös, S. G. Hartke, L. van Iersel, and I. Miklós, *Graph realizations constrained by skeleton graphs*, Electr. J. Comb., 24(2), P2.47 (2017).

[16] P. L. Erdős, T. R. Mezei, I. Miklós, and D. Soltész, *Efficiently sampling the realizations of bounded, irregular degree sequences of bipartite and directed graphs*, PLOS ONE, 13 (2018), pp. 1–20.

[17] P. L. Erdős, I. Miklós, and Z. Toroczkai, *A decomposition based proof for fast mixing of a Markov chain over balanced realizations of a joint degree matrix*, SIAM Journal on Discrete Mathematics, 29 (2015), pp. 481–499.

[18] T. Feder, A. Guetz, M. Mihail, and A. Saberi, *A local switch Markov chain on given degree graphs with application in connectivity of peer-to-peer networks*, in Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science, 2006, pp. 69–76.

[19] M. Gjoka, B. Tillman, and A. Markopoulou, *Construction of simple graphs with a target joint degree matrix and beyond*, in Proceedings of IEEE Conference on Computer Communications, 2015, pp. 1553–1561.

[20] C. S. Greenhill and M. Sfragara, *The switch Markov chain for sampling irregular graphs and digraphs*, Theoretical Computer Science, 719 (2018), pp. 1–20.

[21] T. Homma, *A theorem on continuous functions*, Kodai Mathematical Seminar Reports, 4 (1952), pp. 13–16.

[22] M. Jerrum and A. Sinclair, *Approximating the permanent*, SIAM Journal on Computing, 18 (1989), pp. 1149–1178.

[23] M. Jerrum and A. Sinclair, *Fast uniform generation of regular graphs*, Theoretical Computer Science, 73 (1990), pp. 91–100.

[24] R. Kannan, P. Tetali, and S. Vempala, *Simple Markov-chain algorithms for generating bipartite graphs and tournaments*, Random Structures and Algorithms, 14 (1999), pp. 293–308.

[25] D. A. Levin, Y. Peres, and E. L. Wilmer, *Markov chains and mixing times*, American Mathematical Society, 2009.

[26] P. Mahadevan, D. V. Krioukov, K. R. Fall, and A. Vahdat, *Systematic topology analysis and generation using degree correlations*, in Proceedings of the ACM SIGCOMM 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, ACM, 2006, pp. 135–146.

[27] I. Miklós, P. L. Erdös, and L. Soukup, *Towards random uniform sampling of bipartite graphs with given degree sequence*, Electr. J. Comb., 20(1), P16 (2013).

[28] A. Patrinos and S. Hakimi, *Relations between graphs and integer-pair sequences*, Discrete Mathematics, 15 (1976), pp. 347–358.
[29] A. Sinclair, *Improved bounds for mixing rates of Markov chains and multicommodity flow*, Comb. Probab. Comput., 1 (1992), pp. 351–370.
[30] I. Stanton and A. Pinar, *Constructing and sampling graphs with a prescribed joint degree distribution*, ACM Journal of Experimental Algorithmics, 17(1), 3.5 (2011).
[31] J. V. Whittaker, *A mountain-climbing problem*, Canadian Journal of Mathematics, 18 (1966), pp. 873–882.