Polaris: Sampling from the Multigraph Configuration Model with Prescribed Color Assortativity

Giulia Preti
giulia.preti@centai.eu
CENTAI
Turin, Italy

Matteo Riondato
mriondato@amherst.edu
Amherst College
Amherst, MA, USA

Aristides Gionis
argioni@kth.se
KTH Royal Institute of Technology
Stockholm, Sweden

Gianmarco De Francisci Morales
gdfm@acm.org
CENTAI
Turin, Italy

Abstract

We introduce Polaris, a network null model for colored multigraphs that preserves the Joint Color Matrix. Polaris is specifically designed for studying network polarization, where vertices belong to a side in a debate or a partisan group, represented by a vertex color, and relations have different strengths, represented by an integer-valued edge multiplicity. The key feature of Polaris is preserving the Joint Color Matrix (JCM) of the multigraph, which specifies the number of edges connecting vertices of any two given colors. The JCM is the basic property that determines color assortativity, a fundamental aspect in studying homophily and segregation in polarized networks. By using Polaris, network scientists can test whether a phenomenon is entirely explained by the JCM of the observed network or whether other phenomena might be at play.

Technically, our null model is an extension of the configuration model: an ensemble of colored multigraphs characterized by the same degree sequence and the same JCM.To sample from this ensemble, we develop a suite of Markov Chain Monte Carlo algorithms, collectively named Polaris-$*$. It includes Polaris-B, an adaptation of a generic Metropolis-Hastings algorithm, and Polaris-C, a faster, specialized algorithm with higher acceptance probabilities. This new null model and the associated algorithms provide a more nuanced toolset for examining polarization in social networks, thus enabling statistically sound conclusions.

CCS Concepts

• **Information systems** → *Web mining*; • **Theory of computation** → *Graph algorithms analysis*; **Random walks and Markov chains**; **Generating random combinatorial structures**; **Social networks**; • **Mathematics of computing** → *Random graphs*.

Keywords

Hypothesis Testing, Null Model, Polarization

ACM Reference Format:
Giulia Preti, Matteo Riondato, Aristides Gionis, and Gianmarco De Francisci Morales. 2025. Polaris: Sampling from the Multigraph Configuration Model with Prescribed Color Assortativity. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining (WSDM '25), March 10–14, 2025, Hannover, Germany*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3701551.3703560

WSDM '25, March 10–14, 2025, Hannover, Germany

ACM ISBN 979-8-4007-1329-3/25/03
https://doi.org/10.1145/3701551.3703560

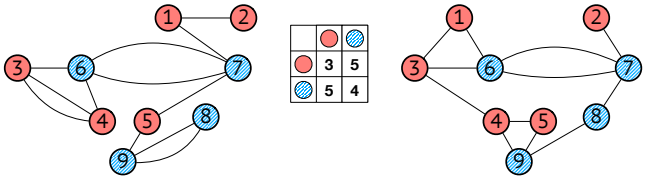

Figure 1: Two multigraphs with the same degree sequence and JCM.

1 Introduction

Polarization is perceived as one of the largest problems in our society [16]. Scientists have studied the phenomenon extensively, more recently by using data from social media [10, 25, 28, 29, 37]. Many different theories try to explain the phenomenon, from affective polarization to partisan identity and echo chambers [2, 8, 21, 24, 31, 45]. However, definite evidence is still lacking.

Existing observational studies often use network representations to study the problem. This choice allows employing the ample network- and graph-theoretical toolset to define properties and compute relevant measures. However, such quantities are only significant insofar as they are not statistical noise. For this purpose, *null-hypothesis models* are used to assess statistical significance.

Unfortunately, to date, the network null models used in these studies are exceedingly simple [40]. They usually preserve only basic characteristics of the graph structure, such as density or degree sequences [18], but they ignore the interplay between opinions or communities that describe the polarization phenomenon.

The main contribution of this work is to propose a new network null model geared towards the study of network polarization. In particular, our null model is a *statistical ensemble of colored multigraphs*: graphs where each vertex has a color (i.e., a single label) and edges can appear multiple times. Vertex colors, or labels, are often used to represent the different sides in a controversial argument or debate, or groups such as partisan identities [10, 19, 22]. Multiedges are commonly used to represent endorsement networks (e.g., retweet or interaction networks) [19, 20, 22], where the multiplicity represents the strength of the relationship between two vertices.

The ensemble we consider is a microcanonical one akin to the configuration model [4, 5], i.e., its members are all and only the graphs with a specific degree sequence. The graph ensemble for our null model is additionally defined by a property shared by all members of the ensemble: the Joint Color Matrix (JCM). This matrix determines the number of edges that connect vertices of different colors. Figure 1 depicts two small graphs belonging to the same

ensemble and their associated JCM. The JCM determines important properties of the graph, e.g., its color assortativity [36] which is fundamental in the study of homophily and segregation [30].

We devise a suite of Markov chain Monte Carlo algorithms, named Polaris-$*$, to sample from the ensemble. We prove the Markov chain is irreducible and aperiodic, thus having a unique stationary distribution. The first algorithm, Polaris-B, is an adaptation of an existing algorithm [17] using the Metropolis-Hastings method. The second algorithm, Polaris-C, takes into account the vertex colors in a more judicious manner. As a result, Polaris-C has higher acceptance probabilities than Polaris-B, and mixes faster.

2 Related Work

When searching for patterns in network data it is essential to be able to reason about their significance. In statistics, there is a long tradition of assessing significance by comparing an observed pattern with its occurrence in a *randomized null model* [15]. Extending this idea to networks leads to *random-graph null models*, where one compares properties observed in real-world networks with properties observed in networks sampled from a certain random-graph distribution. While there is a vast literature on random-graph models, such as the Erdős-Rényi random graph [13] and the preferential attachment model [3, 6], which are simple to generate via an iterative sampling process, practitioners often seek to sample networks from a space of networks satisfying certain constraints. The most commonly-used constrained random-graph null model is the *configuration model* [4, 5, 17], where the sample space consists of all networks having a specified degree sequence. Configuration models have a long research history with applications in sociology [34], ecology [9], systems biology [32], and other disciplines.

The *Markov chain Monte Carlo (MCMC)* method is an archetypal approach for sampling from the space of networks with a fixed degree sequence. The MCMC technique performs a random walk over the sample space $\mathcal{G}$ of feasible networks and appropriately modifies the transition probabilities of the walk, e.g., using the Metropolis-Hastings algorithm [38], so that the *stationary distribution* is a desirable target distribution, such as the *uniform* one. Typically, the neighboring states in the Markov chain are networks that differ only in a *two-edge swap*, making it easy to transition between states and sample random networks from the whole space $\mathcal{G}$. By proving that the Markov chain is *strongly connected* (*irreducible*) and *aperiodic*, it can be argued that there is a *unique stationary distribution*, and the Metropolis-Hastings algorithm can be used to obtain samples from it.

A question of theoretical interest is the *mixing time* of the MCMC method, i.e., the number of steps required before the actual sample distribution is ϵ-close to the stationary distribution. Theoretical papers have derived conditions that the method is *rapidly mixing*, i.e., the number of required steps is polynomial [11, 27]. However, the general case is still not fully understood. Furthermore, existing bounds are high-degree polynomials and are mostly of theoretical interest. In practice, researchers employ various diagnostics to assess empirically the MCMC convergence [12, 39], such as comparing the variance of sample graph statistics inside a sequence of the chain and against the variance across multiple sequences [23].

Other types of graph ensembles have been proposed as a null model to assess the statistical importance of patterns and graph structure. Those include *maximum-entropy models* [42], which however preserve the degree sequence only in expectation, and *exponential random graph models* [41], which increase the probability of observing certain subgraph structures, and which are also typically sampled using MCMC methods.

Overall, a large number of methods for sampling graph null models have been presented in the literature, and the ideas have been applied to analyzing data from different disciplines. Nevertheless, perhaps surprisingly, no previous work on preserving the color assortativity of a network exists.

3 Preliminaries

This section introduces the main concepts and notation used throughout the work. We use double curly braces to denote multisets, e.g., $A = \{\!\{a, b, b, c, d, d, d\}\!\}$, and $\omega_A(a)$ is the *multiplicity* of an element a in a multiset A, i.e., the number of times a appears in A. $|A|$ denotes the multiset cardinality of a set, e.g., $|A| = 7$ in the example above. The notation $\mathrm{set}(A)$ represents the set obtained by removing all duplicates from A, e.g., $\mathrm{set}(A) = \{a, b, c, d\}$.

Definition 3.1 (Colored Multigraph). A *colored, undirected, multigraph* is a tuple $G \doteq (V, E, \mathcal{L}, \lambda)$, where V is a set of vertices, E is a multiset of edges between vertices, each edge being an unordered pair of vertices from V, and $\lambda : V \to \mathcal{L}$ is a labeling function assigning a color (i.e., a single label) from the set $\mathcal{L}$ to each vertex.

We allow (multiple) self-loops, i.e., edges of the type (u, u), $u \in V$. All multigraphs we consider are colored, so henceforth we use "multigraph" to mean "colored multigraph". We refer to edges incident to vertices with the same color as *monochrome* edges, and edges incident to vertices with different colors as *bichrome* edges.

Two edges (u, w), (v, z) are *distinct* when they are two different members of E. Two distinct edges may be *copies* of the same multiedge, i.e., be incident to the same pair of vertices, or to the same vertex if they are self-loop. We write $(u, w) = (v, z)$ if two edges are copies, but since edges are unordered pairs of vertices, this notation does not imply $u = v$ and $w = z$, as it may be $u = z$ and $w = v$.

A multigraph $G = (V, E, \mathcal{L}, \lambda)$ can be seen as an integer-weighted graph $G' = (V, E', \mathcal{L}, \lambda, \mathrm{w})$, with $E' = \mathrm{set}(E)$, and w a function that assigns a natural weight to edges in E', so that $\mathrm{w}(e) = \omega_E(e)$.

Given a multigraph $G = (V, E, \mathcal{L}, \lambda)$, for each $u \in V$, $\Gamma_G(u)$ denotes the multiset of neighbors of u, and $d_G(u) \doteq |\Gamma_G(u)|$ the *degree of u in G*. For each $\ell \in \mathcal{L}$, let $V^\ell \doteq \{v \in V : \lambda(v) = \ell\}$ be the set of vertices with color ℓ. For each $u \in V$ and $\ell \in \mathcal{L}$, let $\Gamma_G^\ell(u) \doteq \{\!\{v \in V^\ell : (u, v) \in E\}\!\}$ be the multiset of neighbors of u in G with color ℓ, and let $\gamma_G^\ell(u) \doteq \left|\Gamma_G^\ell(u)\right|$. Clearly, $d_G(u) = \sum_{\ell \in \mathcal{L}} \gamma_G^\ell(u)$.

Definition 3.2 (JCM). The *Joint Color Matrix* J_G of a multigraph $G = (V, E, \mathcal{L}, \lambda)$ is the symmetric square matrix $\mathrm{J}_G \in \mathbb{N}^{|\mathcal{L}| \times |\mathcal{L}|}$ where each entry $J_G[\ell, r]$ is the number of edges between a vertex with color ℓ and a vertex with color r, i.e.,

$$J_G[\ell, r] \doteq |\{\!\{(u, w) \in E : \lambda(u) = \ell \wedge \lambda(w) = r\}\!\}| \ .$$

Figure 1 shows two multigraphs with two colors, the same degree sequence, and the same JCM (shown in the center of the figure).

3.1 Null Models for Graph Properties

For any multigraph G, let $\mathcal{P}_G$ be a set of properties from G, e.g., the number of edges, the degree sequence, the diameter, or similar structural properties, which may be scalars, vectors, or matrices.

Let $\mathring{G} = (V, E, \mathcal{L}, \lambda)$ be an observed multigraph. Given $\mathcal{P}_{\mathring{G}}$, the microcanonical *null model* $\Pi \doteq (\mathcal{Z}, \pi)$ is a tuple where $\mathcal{Z}$ is the set of all and only the multigraphs $G = (V, E_G, \mathcal{L}, \lambda)$ on the same set of vertices, with the same colors and coloring function as $\mathring{G}$, and that preserve each property in $\mathcal{P}_{\mathring{G}}$ i.e., such that $\mathcal{P}_G = \mathcal{P}_{\mathring{G}}$, and where π is a probability distribution over $\mathcal{Z}$. Clearly $\mathring{G} \in \mathcal{Z}$, and the graphs in $\mathcal{Z}$ only differ by their multisets of edges.

3.2 Markov Chain Monte Carlo Methods

All algorithms in Polaris-$*$ follow the *Markov chain Monte Carlo (MCMC) method*, using the *Metropolis-Hastings (MH) approach* [33, Ch. 7 and 10]. Let us now introduce these concepts.

Let $\mathcal{G} = (\mathcal{S}, \mathcal{E}, \mathrm{w})$ be a directed, weighted, strongly connected, and aperiodic[1] graph, which may have self-loops. The vertices in $\mathcal{S}$ are known as *states*, and $\mathcal{G}$ is known as a state graph. Using the same notation we defined previously, for any state $s \in \mathcal{S}$, $\Gamma_{\mathcal{G}}(s)$ denotes the set of (out-)*neighbors* of s, i.e., the set of states u s.t. $(s, u) \in \mathcal{E}$. Iff $u \in \Gamma_{\mathcal{G}}(s)$, then $\mathrm{w}(s, u) > 0$, and it holds $\sum_{u \in \Gamma_{\mathcal{G}}(s)} \mathrm{w}(s, u) = 1$. Thus, for any $u \in \mathcal{S}$, we can define the *transition probability* $\tau_{s,u}$ *from* s *to* u as $\mathrm{w}(s, u)$ if $u \in \Gamma_{\mathcal{G}}(s)$, and 0 otherwise.

Given any $\mathcal{G}$ as above, a *neighbor proposal probability distribution* ξ_v over $\Gamma_{\mathcal{G}}(v)$ for any $v \in \mathcal{S}$, and any probability distribution ϕ over $\mathcal{S}$, the *MH approach* is a generic procedure to sample a state $s \in \mathcal{S}$ according to ϕ. Starting from any $v \in \mathcal{S}$, one first draws a neighbor u of v according to ξ_v, and then "moves" to u with probability

$$\alpha_v(u) = \min\left\{1, \frac{\phi(u)}{\phi(v)} \frac{\xi_u(v)}{\xi_v(u)}\right\},$$

otherwise remains in v. The quantity $\alpha_v(u)$ is known as the *acceptance probability* of u. The sequence of states obtained by repeating this procedure forms a Markov chain over $\mathcal{S}$ with unique stationary distribution ϕ. Thus, after a sufficiently large number of steps t, the state v_t at time t is distributed according to ϕ, and can be considered a sample of $\mathcal{S}$ according to ϕ.

To use MH, it is necessary to define: *(i)* the graph $\mathcal{G}$ as above, taking special care in ensuring that it is strongly-connected and aperiodic; *(ii)*, the neighbor sampling distribution $\xi_s()$ for every state $s \in \mathcal{S}$; and *(iii)* the desired sampling distribution ϕ over $\mathcal{S}$.

4 A Null Model for Vertex-Colored Graphs

Given an observed $\mathring{G} \doteq (V, E, \mathcal{L}, \lambda)$, with $V = \{v_1, \ldots, v_{|V|}\}$, we consider the null model $\Pi = (\mathcal{Z}, \pi)$ where $\mathcal{P}_{\mathring{G}}$ consists of the degree sequence $\left[d_{\mathring{G}}(v_1), \ldots, d_{\mathring{G}}\left(v_{|V|}\right)\right]$ and the JCM $J_{\mathring{G}}$.

This null model is essentially the simplest one that considers the color information, if one assumes that the color of a vertex is an intrinsic property. While one could think of preserving only the colored degree sequences for each color, doing so is equivalent to preserving the "generic" degree sequence, and thus does not leverage the color information in any meaningful way. Indeed, this can be done on the unlabeled version of the multigraph [18].

[1] A graph is aperiodic iff the greatest common divisor of the lengths of its cycles is 1.

Our goal is to design efficient MCMC algorithms to sample from $\mathcal{Z}$ w.r.t. π as defined above. We first define two operations that allow transforming a multigraph G into a multigraph H, potentially identical to G. The first operation is the classic Double Edge Swap (DES), known under many names and introduced many times in the literature [1, 7, 26, 43, 44, 46].

Definition 4.1 (Double Edge Swap (DES)). Given a multigraph $G \doteq (V, E, \mathcal{L}, \lambda)$, let $(u, w), (v, z)$ be two distinct edges in E. Consider the multigraph $H = (V, (E \setminus \{\!\{(u, w), (v, z)\}\!\}) \cup \{\!\{(u, z), (w, v)\}\!\}, \mathcal{L}, \lambda)$. We call the operation that "swaps" $(u, w), (v, z)$ with $(u, z), (w, v)$ a *Double Edge Swap (DES)*, and denote it $(u, w), (v, z) \to (u, z), (w, v)$.

We say that a DES is *applied to* the origin multigraph G to obtain the destination multigraph H, or that a DES *transforms* G *into* H.

For every unordered pair $((u, w), (v, z))$ of distinct edges in the origin graph, there are exactly two DESs that involve them: $(u, w), (v, z) \to (u, z), (v, w)$ and $(u, w), (v, z) \to (u, v), (z, w)$. If the destination multigraph H is the same for both DESs, we say that the DESs are *equivalent*. If $H = G$, we say that the DES is a *no-op*, otherwise we say that the DES is a *moving* DES. For the same unordered pair of distinct edges in the origin graph, one DES may be a no-op, and the other may be a moving DES.

Multiple expressions may correspond to the same DES, as a DES is defined by the multiset of edges in the origin multigraph and by the multiset of edges in the destination multigraph. For example, the expressions $(u, w), (v, z) \to (u, z), (w, v)$ and $(z, v), (u, w) \to (u, z), (w, v)$ both denote the same DES.

DESs can be used in MCMC algorithms to sample from a null model that preserves the degree sequence of an observed multigraph [18, and references therein]: given a DES, the destination multigraph has the same vertices and the same degree sequence as the origin. Conversely, the JCM may or may not be preserved by a DES. Thus we define the following specific operation.

Definition 4.2 (JCM-preserving Double Edge Swap (JDES)). A *JCM-preserving Double Edge Swap (JDES)* is a DES such that the destination multigraph H retains the JCM of the origin multigraph G.

An example of JDES that can be applied to the left multigraph in Figure 1 is $(1, 7), (3, 6) \to (1, 6), (3, 7)$, while the operation $(3, 4), (9, 8) \to (3, 8), (9, 4)$ is a DES but not a JDES.

For any unordered pair $((u, w), (v, z))$ of distinct edges in the origin multigraph, zero, one, or both DESs may be JDESs. We now give a complete characterization of which DESs are JDESs, considering different cases based on the properties of the edges involved.

Case 0: $\{\lambda(u), \lambda(w)\} \cap \{\lambda(v), \lambda(z)\} = \emptyset$, i.e., the two edges have disjoint vertex colors. Then, neither DES is a JDES.

Case 1: $|\{u, w, v, z\}| = 1$, i.e., (u, w) and (v, z) are copies of the same self-loop multiedge. Both DESs are JDESs and no-ops.

Case 2A: $|\{u, w, v, z\}| = 2 \wedge u = w \wedge v = z \wedge \lambda(u) = \lambda(v)$, i.e., (u, w) and (v, z) are two different self-loops on vertices with the same color. Both DESs are equivalent JDESs and moving DESs.

Case 2B: $|\{u, w, v, z\}| = 2 \wedge u \neq w \wedge v \neq z \wedge \lambda(u) = \lambda(w)$, i.e., (u, w) and (v, z) are identical non-self-loop monochrome multiedges. Both DESs are JDESs; one is a no-op, while the other creates a self-loop.

Case 2C: $|\{u,w,v,z\}| = 2 \wedge u \neq w \wedge v \neq z \wedge \lambda(u) \neq \lambda(w)$, i.e., (u,w) and (v,z) are identical non-self-loop bichrome multiedges. Only one DES is a JDES, and is a no-op.

Case 2D: $|\{u,w,v,z\}| = 2 \wedge (u = w \vee v = z) \wedge \neg(u = w \wedge v = z)$, i.e., one edge is a self-loop, the other is not but is incident to the self-loop vertex. Both DESs are JDESs and no-ops.

Case 3A: $|\{u,w,v,z\}| = 3 \wedge (w = u \vee v = z) \wedge \{\lambda(u),\lambda(w)\} \cap \{\lambda(v),\lambda(z)\} \neq \emptyset$, i.e., one edge is a self-loop, the other is not and is incident to different vertices than the self-loop, and at least one of these vertices has the same color as the self-loop. Both DESs are equivalent JDESs and moving DESs.

Case 3B: $|\{u,w,v,z\}| = 3 \wedge u \neq w \wedge v \neq z \wedge |\{\lambda(u),\lambda(w),\lambda(v),\lambda(z)\}| = 1$, i.e., neither edge is a self loop, and since $|\{u,w,v,z\}| = 3$, it holds exactly one of $u = v$, $u = z$, $w = v$, or $w = z$, so the edges form a wedge with vertices sharing the same color. Assume, w.l.o.g., that $u = v$. Then both DESs are JDESs; one is a no-op, while the other creates a self-loop on u and an edge between the vertices at the extremes of the former wedge.

Case 3C: $|\{u,w,v,z\}| = 3 \wedge u \neq w \wedge v \neq z \wedge |\{\lambda(u),\lambda(w),\lambda(v),\lambda(z)\}| = 2 \wedge (\lambda(u) = \lambda(w) \vee \lambda(v) = \lambda(z))$. Similar to Case 3B, but exactly one edge is monochrome. Both DESs are JDESs; one is a no-op, while the other creates a self-loop on u and an edge between the two extremes of the former wedge.

Case 3D: $|\{u,w,v,z\}| = 3 \wedge u \neq w \wedge v \neq z \wedge |\{\lambda(u),\lambda(w),\lambda(v),\lambda(z)\}| = 2 \wedge \lambda(u) \neq \lambda(w) \wedge \lambda(v) \neq \lambda(z)$. The edges form a wedge where the endpoints of the wedge have the same color and the vertex in the middle has a different color. One DES is a JDES and is a no-op.

Case 3E: $|\{u,w,v,z\}| = 3 \wedge u \neq w \wedge v \neq z \wedge |\{\lambda(u),\lambda(w),\lambda(v),\lambda(z)\}| = 3$. The edges form a wedge, but all three vertices have different colors. One DES is a JDES and is a no-op.

Case 4A: $|\{u,w,v,z\}| = 4 \wedge |\{\lambda(u),\lambda(w),\lambda(v),\lambda(z)\}| = 3 \wedge \lambda(u) \neq \lambda(w) \wedge \lambda(v) \neq \lambda(z)$. The edges are incident to four distinct vertices, neither of them is monochrome, they are not both bichrome with the same two colors, but they are incident to one vertex with the same color. One DES is a JDES and is a moving DES.

Case 4B: $|\{u,w,v,z\}| = 4 \wedge |\{\lambda(u),\lambda(w),\lambda(v),\lambda(z)\}| = 2 \wedge \lambda(u) \neq \lambda(w) \wedge \lambda(v) \neq \lambda(z)$. The edges are incident to four distinct vertices and are both bichrome with the same two colors. One DES is a JDES and is a moving DES.

Case 4C: $|\{u,w,v,z\}| = 4 \wedge |\{\lambda(u),\lambda(w),\lambda(v),\lambda(z)\}| \in \{1,2\} \wedge (\lambda(u) = \lambda(w) \vee \lambda(v) = \lambda(z))$. The edges are incident to four distinct vertices, with at least three of them having the same color. Both DESs are JDESs, they are not equivalent, and both are moving DESs.

There cannot be more than two JDESs transforming G into H, for $H \neq G$. If there are two, they involve the same pair of edges and are equivalent. All JDESs are reversible: if there are JDESs transforming G into H, then there are JDESs transforming H into G.

4.1 Strong Connectivity and Aperiodicity of $\mathcal{Z}$ via JDESs

In Polaris, the state space $\mathcal{S}$ of the state graph $\mathcal{G} = (\mathcal{S}, \mathcal{E}, \mathrm{w})$ is $\mathcal{Z}$, and the desired probability distribution according to which to sample is π. The edge set $\mathcal{E}$ is defined as follows. For any $G, H \in \mathcal{Z}$, there is an edge $(G,H) \in \mathcal{E}$ if there is a JDES from $G \in \mathcal{Z}$ to $H \in \mathcal{Z}$. Clearly, if that is the case, there is also an edge $(H,G) \in \mathcal{E}$ as all JDESs are reversible. Additionally, there may be a self-loop in G even if there is no JDES from G to G, but G has a neighbor H such that the acceptance probability $\alpha_G(H)$ is strictly less than 1.

We say that G and H are *neighbors* iff there is a JDES transforming G into H. As required by MH, we show that the resulting state graph is strongly connected (Theorem 4.3) and aperiodic (Theorem 4.4) (full proofs in the extended version of the paper [2]), which ensures the chain is ergodic.

Theorem 4.3. *The state space $\mathcal{Z}$ is strongly connected by JDESs.*

The proof of this theorem explicitly builds a sequence of JDESs from any state $G \in \mathcal{Z}$ to any other $H \in \mathcal{Z}$, by first going from G to a $\tilde{G} \in \mathcal{Z}$ such that every vertex has in $\tilde{G}$ exactly the same number of neighbors of each color as it has in H, then going from $\tilde{G}$ to H.

Theorem 4.4. *Given a multigraph G, if either of the following conditions holds, then the state graph $\mathcal{G}$ is aperiodic:*

- *there exist two edges (u,w) and (v,z) that fall in cases 1, 2A, 2B, 2C, 2D, 3A, 3B, 3C, 3D, 3E, or 4C of the classification; or*
- *there exist a color $\ell \in \mathcal{L}$ such that there are bichrome edges (u,v), (u,z), (w,x), with all of u, v, z, w and x distinct, and $\lambda(u) = \lambda(w) = \ell$.*

The proof of Theorem 4.4 involves a case-by-case analysis of the JDES, showing that either there must be a self-loop on a vertex in $\mathcal{G}$, or there are cycles of length 3 and 2, thus ensuring aperiodicity.

The conditions in Theorem 4.4 are extremely mild. For example, the first condition implies that if there is a color such that there are two monochrome edges with that color, then the state graph is aperiodic. Only a (relatively) small class of unusual multigraphs results in periodic state graphs. Additionally, the conditions are not necessary for the graph to be aperiodic: the algorithms we present run Markov chains on a state graph that may have additional self-loops, as they are based on MH.

4.2 A first baseline algorithm

To warm up, we present a baseline algorithm Polaris-B, which is an adaptation of Fosdick et al. [17, Algorithm 3][3] to our task of interest. Fosdick et al. [17] introduced the algorithm to sample uniformly from the space of unlabeled multigraphs with the same degree sequence. Polaris-B is a tailored version of this algorithm to sample according to any distribution π from the space of colored multigraphs with the same degree sequence and the same JCM.

Polaris-B (pseudocode in Algorithm 1) starts by setting the current state G of the Markov chain to the observed multigraph $\mathring{G}$ (Line 1). It then enters a loop for t iterations. At each iteration, it first samples two edges e_1 and e_2 uniformly at random from the population of ordered pairs of distinct edges (Lines 4–5). The algorithm then randomly chooses one of the two possible DESs involving e_1 and e_2 (Line 6). If the selected DES des is not a JDES (Line 7), the algorithm samples a new DES; if it is a no-op (Line 8), the Markov chain stays in G. Otherwise, the algorithm computes a value ρ that depends on properties of the sampled edges, which is used as follows to ensure that the stationary distribution of the Markov chain

[2] https://arxiv.org/abs/2409.01363

[3] This algorithm only appears in the arXiv version of this paper [18].

Algorithm 1: Polaris-B

Input: Observed multigraph $\mathring{G} \doteq (V, E, \mathcal{L}, \lambda)$, distribution π over $\mathcal{Z}$, number of iterations t
Output: Multigraph drawn from $\mathcal{Z}$ according to π

1 $G \leftarrow \mathring{G}$
2 **repeat** t **times**
3 **do**
4 $e_1 = (u, w) \leftarrow$ edge drawn u.a.r. from E
5 $e_2 = (v, z) \leftarrow$ edge drawn u.a.r. from $E \setminus \{e_1\}$
6 des $= (e_1, e_2 \to e_1', e_2') \leftarrow$ DES drawn u.a.r. from $\{(u, w), (v, z) \to (u, z), (v, w),\ (u, w), (v, z) \to (u, v), (w, z)\}$
7 **while** des *is not a JDES* // Case 0
8 **if** des *is a no-op* **then continue** // Cases 1, 2C, 2D, 3D, 3E, and no-ops for other cases
9 **if** $|\{u, w, v, z\}| = 4$ **then** // Cases 4(A,B,C), moving DES
10 $\rho \leftarrow {(\omega_G(e_1') + 1)(\omega_G(e_2') + 1)}\big/{\omega_G(e_1)\omega_G(e_2)}$
11 **else if** $|\{u, w, v, z\}| = 3$ **then**
12 **if** e_1 *is a self-loop* **or** e_2 *is a self-loop* **then** // Case 3A
13 $\rho \leftarrow {(\omega_G(e_1') + 1)(\omega_G(e_2') + 1)}\big/{2\omega_G(e_1)\omega_G(e_2)}$
14 **else** // Cases 3B and 3C, moving DES
15 $\rho \leftarrow {2(\omega_G(e_1') + 1)(\omega_G(e_2') + 1)}\big/{\omega_G(e_1)\omega_G(e_2)}$
16 **else** // i.e., $|\{u, w, v, z\}| = 2$
17 **if** *both* e_1 *and* e_2 *are self-loops* **then** // Case 2A
18 $\rho \leftarrow {(\omega_G(e_1') + 2)(\omega_G(e_1') + 1)}\big/{4\omega_G(e_1)\omega_G(e_2)}$
19 **else** // Case 2B
20 $\rho \leftarrow {4(\omega_G(e_1') + 1)(\omega_G(e_2') + 1)}\big/{\omega_G(e_1)(\omega_G(e_1) - 1)}$
21 $H \leftarrow$ apply des to G
22 **if** $\mathsf{Uniform}(0, 1) < {\rho\pi(H)}\big/{\pi(G)}$ **then** $G \leftarrow H$
23 **return** G

is π. Let $H \neq G$ be the multigraph obtained by applying the JDES 'des' to G (Line 21). Polaris-B checks if $\rho\pi(H)/\pi(G)$ is greater than a real number sampled uniformly at random from $[0, 1]$, and if so, it sets the current state G of the Markov chain to H (Line 22), otherwise the chain remains in G.

The only difference in Polaris-B w.r.t. [17, Algorithm 3] is that it checks if the sampled DES is a JDES, and keeps sampling a new DES until it is a JDES (Line 7).

Theorem 4.5. *The Markov chain run by Polaris-B has stationary distribution π.*

The complete proof is in the extended version of the paper, and shows that Polaris-B follows the MH approach, thus ensuring the thesis.

In practice, the number of steps t must be chosen in such a way that the multigraph returned by Polaris-B is, at least approximately, distributed according to π, i.e., t should be greater or equal to the mixing time for the Markov chain. Theoretical results for the mixing time of these Markov chains are hard to obtain: even in the case of the state space of multigraphs connected by DESs (i.e., when only the degree sequence is preserved), upper bounds on the mixing time are only known in limited cases [14]. Therefore, Section 5 presents an empirical evaluation of the behavior of t.

4.3 An algorithm tailored to the task

We now present Polaris-C, a color-aware algorithm to sample a multigraph from $\mathcal{Z}$ according to π by leveraging the properties of the data and of the task better than Polaris-B (Section 4.2). As the results of our experimental evaluation show (Section 5), this algorithm has higher acceptance probability and converges faster than the baseline presented earlier.

Polaris-C improves over Polaris-B in several ways:

- it avoids sampling pairs of distinct edges falling in Case 0 of the characterization of JDES, i.e., pairs of distinct edges such that neither DES involving them is a JDES;
- if the sampled pair of distinct edges is such that one of the DESs is a no-op or not a JDES and the other is a moving JDES (Cases 2B, 2C, 3B, 3C) Polaris-C always chooses the moving one;
- if the sampled pair of distinct edges is such that the JDESs involving them are equivalent (Cases 1, 2A, 2D, 3A), it deterministically chooses one thus avoiding random choices.

As Polaris-C avoids selecting no-op JDESs in some cases (second bullet point above), one should ask whether the resulting state graph is still aperiodic under the conditions stated in Theorem 4.4. The answer is that the first condition should be modified to hold only for Cases 1, 2C, 2D, 3D, and 4C. The condition for 4C is particularly mild: it only requires the existence of a color $\ell \in \mathcal{L}$ such that there are two non-self-loop, non-copies, monochrome edges with color ℓ, i.e., two edges involving four distinct vertices with the same color.

All the above improvements of Polaris-C over Polaris-B reduce the probability that the Markov chain remains in the current state, while not decreasing (and potentially increasing) the transition probability from a state to any of its different neighbors. In other words, the off-diagonal entries of the transition matrix of the Markov chain realized by Polaris-C are not smaller than the corresponding entries in the transition matrix of the Markov chain realized by Polaris-B. Polaris-C therefore precedes Polaris-B in Peskun's order [38], which implies that it has a smaller mixing time, i.e., requires fewer steps for the state of the chain to be (approximately) distributed according to the stationary distribution.

Polaris-C takes into account the number of *different* colors $\mathsf{nl}(A)$ in a multiset of vertices A to distinguish various cases of the JDES. Formally, w.l.o.g, let $\mathcal{L} = \{0, \ldots, k-1\}$, for some $k > 1$. For any unordered $(\ell, r) \in \mathcal{L} \times \mathcal{L}$, let $E_{G,\ell,r}$ be the multiset of edges incident to one vertex with color ℓ and one vertex with color r. Clearly $E_{G,\ell,\ell}$ is the multiset of the monochrome edges with color ℓ. We also define

$$E_{G,\ell} \doteq \bigcup_{r \in \mathcal{L}} E_{G,\ell,r}$$

as the multiset of edges incident to at least one vertex with color ℓ. Given a multiset A of vertices, let

$$\mathsf{nl}(A) \doteq |\{\lambda(v) : v \in A\}|.$$

Algorithm 2 presents Polaris-C's pseudocode. The algorithm takes as input the observed multigraph $\mathring{G}$, the distribution π according to which one wants to sample from $\mathcal{Z}$, and a number t of iterations. It keeps track of the current state of a Markov chain on $\mathcal{Z}$ in a variable G initialized to $\mathring{G}$ (Line 1). Polaris-C then enters a loop for t iterations. At each iteration, it first samples a color ℓ from $\mathcal{L}$ uniformly at random (Line 3), then it draws two distinct edges (u, w) and (v, z) uniformly at random respectively from $E_{G,\ell}$ and from $E_{G,\ell} \setminus \{(u, w)\}$ (Lines 4–5). It then checks which case of the JDES characterization should be considered. It either sets the variable jdes to be a moving JDES involving (u, w) and (v, z), possibly by choosing it uniformly at random when there are two non-equivalent, moving, JDESs (only happens in Case 4C, Lines 39–43), or keeps the state of the Markov chain to be the current multigraph

G if the JDESs in the considered case are both no-ops (Cases 2C, 2D, 3D). In the cases when jdes is set, the algorithm also sets the variable ρ to a value that, as we discuss in the analysis of Polaris-C (Theorem 4.6), ensures that the multigraph returned by the algorithm is drawn from $\mathcal{Z}$ according to π. Let now H be the multigraph obtained by applying jdes to G. Polaris-C checks whether a value chosen uniformly at random in $[0, 1]$ is smaller than $\rho\pi(H)/\pi(G)$, and if so, updates the state G of the Markov chain to H (Line 46), otherwise the chain remains in the current state. After t iterations, the current state G is returned.

Theorem 4.6. *The Markov chain run by Polaris-C has stationary distribution π.*

The proof can be found in the extended version of the paper. It essentially shows that, no matter into what case of the classification the sampled JDES falls, the algorithm follows the MH approach for choosing the acceptance probability to ensure the thesis of the theorem.

5 Experimental evaluation

Our experimental evaluation has three objectives. First, we demonstrate the qualitative differences between multigraphs sampled using the traditional configuration model and those obtained from Polaris. We focus on the configuration model as it is the standard reference model in network analysis [35] and aligns with the focus of this paper on microcanonical ensembles. Second, we analyze the extent to which the baseline algorithm Polaris-B differs from the color-aware algorithm Polaris-C in their respective movements within the state space. Lastly, we show the scalability of both Polaris-B and Polaris-C, particularly in relation to the number of vertex colors and the number of edges.

Datasets. We consider 11 real-world labeled networks, whose characteristics are summarized in the extended version of the paper.

Experimental Setup. All experiments are run on an Intel Xeon Silver 4210R CPU@2.40GHz running FreeBSD with 383 GiB of RAM. We evaluate three sampling algorithms: the baseline color-agnostic algorithm Polaris-B, the color-aware algorithm Polaris-C, and the traditional configuration model (CM), which samples from the state space of multigraphs with a prescribed degree sequence. The code and the datasets used are available on GitHub.[4]

For the experiments aimed at the first goal, we allow 10 Markov chains to evolve for $4000m$ iterations, where m is the number of multiedges, recording the degree assortativity of the current state every $0.05m$ iterations. For the experiments aimed at the other two goals, we generate 100 independent samples by using each sampler for $m\log(m)$ iterations.

5.1 Comparison with the Configuration Model

Figure 2 shows that the color assortativity values of the multigraphs sampled by CM significantly diverge from those of the corresponding observed multigraphs, with relative errors close to 1. This discrepancy arises because CM disrupts the original correlations in the observed datasets, generating random graphs with low color assortativity. The effect is more pronounced in datasets with

[4] https://github.com/lady-bluecopper/Polaris

Algorithm 2: Polaris-C

Input: Observed multigraph $\mathring{G} \doteq (V, E, \mathcal{L}, \lambda)$, distribution π over $\mathcal{Z}$, number of iterations t
Output: Multigraph drawn from $\mathcal{Z}$ according to π

1 $G \leftarrow \mathring{G}$
2 **repeat** t **times**
3 $\ell \leftarrow$ color drawn u.a.r. from $\mathcal{L}$
4 $(u, w) \leftarrow$ edge drawn u.a.r. from $E_{G,\ell}$
5 $(v, z) \leftarrow$ edge drawn u.a.r. from $E_{G,\ell} \setminus \{(u, w)\}$
6 **if** $|\{u, w, v, z\}| = 1$ **then continue** // Case 1
7 **else if** $|\{u, w, v, z\}| = 2$ **then**
8 **if** *both* (u, w) **and** (v, z) *are self-loops* **then** // Case 2A
9 jdes $\leftarrow (u, u), (v, v) \rightarrow (u, v), (v, u)$
10 $\rho \leftarrow {(\omega_G((u,v))+2)(\omega_G((u,v))+1)}\big/{\omega_G((u,u))\omega_G((v,v))}$
11 **else if** *neither* (u, w) *nor* (v, z) *is a self-loop* **and** $\lambda(u) = \lambda(w)$ **then** // Case 2B
12 W.l.o.g. let $u = z$ (thus $w = v$)
13 jdes $\leftarrow (u, v), (v, u) \rightarrow (u, u), (v, v)$
14 $\rho \leftarrow {(\omega_G((u,u))+1)(\omega_G((v,v))+1)}\big/{\omega_G((u,v))(\omega_G((u,v))-1)}$
15 **else continue** // Case 2C or 2D
16 **else if** $|\{u, w, v, z\}| = 3$ **then**
17 **if** *either* (u, w) *or* (v, z) *is a self-loop* **then** // Case 3A
18 W.l.o.g. let (u, w) be the self-loop
19 jdes $\leftarrow (u, u), (z, v) \rightarrow (u, v), (z, u)$
20 $\rho \leftarrow {(\omega_G((u,v))+1)(\omega_G((u,z))+1)}\big/{\omega_G((u,u))\omega_G((v,z))}$
21 **else** // W.l.o.g. assume $u = v$
22 **if** $\mathsf{nl}(u, w, v, z) = 1$ **then** // Case 3B
23 jdes $\leftarrow (u, w), (z, u) \rightarrow (u, u), (z, w)$
24 $\rho \leftarrow {(\omega_G((u,u))+1)(\omega_G((w,z))+1)}\big/{\omega_G((u,w))\omega_G((u,z))}$
25 **else if** $\lambda(u) = \lambda(w)$ **or** $\lambda(v) = \lambda(z)$ **then** // Case 3C
26 jdes $\leftarrow (u, w), (z, u) \rightarrow (u, u), (z, w)$
27 $\rho \leftarrow {(\omega_G((u,u))+1)(\omega_G((w,z))+1)}\big/{\omega_G((u,w))\omega_G((u,z))}$
28 **else continue** // Case 3D or 3E
29 **else** // i.e., $|\{u, w, v, z\}| = 4$
30 **if** $\mathsf{nl}(u, w, v, z) = 3$ **and** $\lambda(u) \neq \lambda(w)$ **and** $\lambda(v) \neq \lambda(z)$ **then** // Case 4A
31 W.l.o.g. let $\lambda(u) = \lambda(v)$
32 jdes $\leftarrow (u, w), (v, z) \rightarrow (u, z), (v, w)$
33 $\rho \leftarrow {(\omega_G((u,z))+1)(\omega_G((v,w))+1)}\big/{\omega_G((u,w))\omega_G((v,z))}$
34 **else if** $\mathsf{nl}(u, w, v, z) = 2$ **and** $\lambda(u) \neq \lambda(w)$ **and** $\lambda(v) \neq \lambda(z)$ **then** // Case 4B
35 W.l.o.g. assume $\ell = \lambda(u) = \lambda(v)$ and let $\ell' \doteq \lambda(w) = \lambda(z)$ (it holds $\ell \neq \ell'$)
36 jdes $\leftarrow (u, w), (v, z) \rightarrow (u, z), (v, w)$
37 $\rho \leftarrow \dfrac{\frac{(\omega_G((u,z))+1)(\omega_G((v,w))+1)}{|E_{G,\ell}|(|E_{G,\ell}|-1)} + \frac{(\omega_G((u,z))+1)(\omega_G((v,w))+1)}{|E_{G,\ell'}|(|E_{G,\ell'}|-1)}}{\frac{\omega_G((u,w))\omega_G((v,z))}{|E_{G,\ell}|(|E_{G,\ell}|-1)} + \frac{\omega_G((u,w))\omega_G((v,z))}{|E_{G,\ell'}|(|E_{G,\ell'}|-1)}}$
38 **else** // Case 4C
39 **if** `fairCoinFlip()` *is head* **then**
40 jdes $\leftarrow (u, w), (v, z) \rightarrow (u, z), (v, w)$
41 $\rho \leftarrow {(\omega_G((u,z))+1)(\omega_G((v,w))+1)}\big/{\omega_G((u,w))\omega_G((v,z))}$
42 **else**
43 jdes $\leftarrow (u, w), (z, v) \rightarrow (u, v), (z, w)$
44 $\rho \leftarrow {(\omega_G((u,v))+1)(\omega_G((z,w))+1)}\big/{\omega_G((u,w))\omega_G((v,z))}$
45 $H \leftarrow$ apply jdes to G
46 **if** $\mathsf{Uniform}(0, 1) < {\rho\pi(H)}\big/{\pi(G)}$ **then** $G \leftarrow H$
47 **return** G

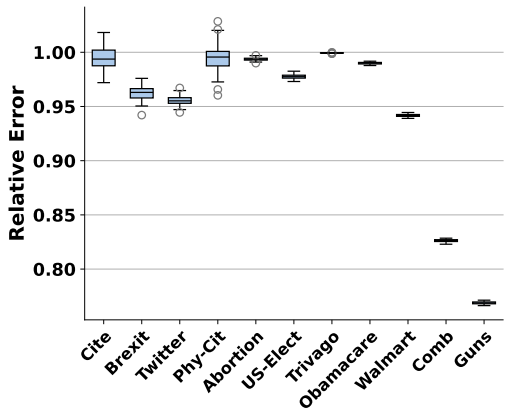

Figure 2: Distribution of the relative errors of color assortativity for samples generated by CM, compared to the color assortativity of the observed datasets, for datasets of increasing size. Results are based on 100 samples. Bars indicate one standard deviation.

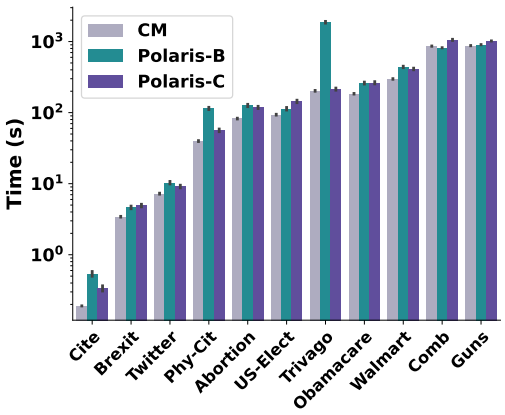

Figure 3: Running time required by each sampler to perform $m\log(m)$ iterations, for datasets of increasing size. Results for 100 samples. Bars indicate one standard deviation.

a larger number of colors or higher color assortativity, where the gap between the observed assortativity and that of the randomized graphs is larger. Consequently, we observe larger relative errors in datasets such as Trivago ($|\mathcal{L}| = 160$), Obamacare, and Abortion (assortativity 0.95), and smaller errors in datasets such as Comb and Guns ($|\mathcal{L}| = 2$ with assortativity values of 0.31 and 0.35, respectively). This result proves that CM does not adequately capture the color assortativity present in the observed data.

Figure 3 presents the running time of each sampler across the different datasets. This plot highlights that, despite Polaris-B and Polaris-C performing more complex operations and needing to update more quantities after each swap operation, their running time is similar to that of CM. However, the differences in performance become especially visible in datasets with a larger number of labels, such as Trivago and Phy-Cit. In these datasets, Polaris-B takes, on average, one order of magnitude longer to generate a sample compared to the other two samplers. As the number of colors increases, the likelihood that the sampled DES is not a JDES also increases, thus increasing the running time. As a consequence, the algorithm must repeatedly sample new DESs until it finds one that is a JDES, which adds considerable overhead to the process.

Figure 4 presents the running time required by Polaris-B and Polaris-C to perform $m\log(m)$ iterations on different versions of the Walmart dataset (left), and the distribution of the relative errors of color assortativity of the samples generated by CM (right). Starting from the 11 available colors (product categories), we cluster these colors to create new realistic sets of 2, 4, and 8 colors.

The running time of CM is not affected by the number of colors, as it samples from a state space that is agnostic to vertex attributes. Therefore we omit it from this plot. Interestingly, the running time of Polaris-C remains consistent across the different numbers of colors. This consistency is likely due to Polaris-C maintaining a high acceptance rate, which produces a similar number of updates regardless of the number of colors.

In contrast, the running time of Polaris-B grows as the number of colors increases. As already mentioned, a higher number of colors increases the probability that a sampled DES is not a JDES.

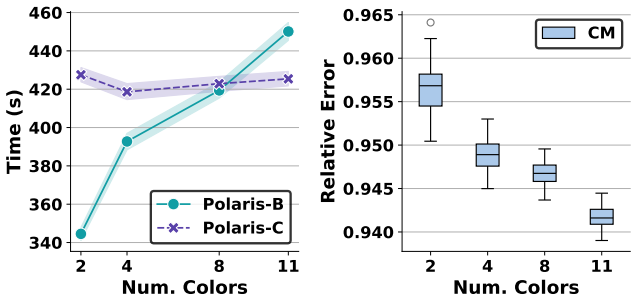

Figure 4: Running time (left) required by Polaris-B and Polaris-C to perform $m\log(m)$ iterations in different versions of Walmart, and distribution of the relative errors of color assortativity of the samples generated by CM (right). Results for 100 samples.

Consequently, more DESs need to be drawn before finding one that is a JDES, which leads to an increased running time.

The figure also shows the distribution of the relative error of color assortativity values for the multigraphs generated by CM. Again, we observe that the color assortativity of the sampled multigraphs significantly diverges from those of the original multigraphs.

5.2 Performance Analysis of Polaris-∗

Figure 5 consists of three panels, each addressing a different aspect of the performance and behavior of Polaris-B and Polaris-C, in three different datasets: Cite, Brexit, and Twitter. The top panel shows the running time as a function of the number of iterations. Polaris-C is faster than Polaris-B in most cases.

The middle panel shows the fraction of iterations with four possible outcomes: (*i*) the sampled DES is not a JDES (*Out of Space*), (*ii*) the DES is a no-op JDES (*Unchanged*), (*iii*) an accepted transition to the next state (*Accepted*), and (*iv*) a rejected transition (*rejected*). Polaris-C avoids sampling DESs that are not JDES, thus having no *Out of Space* outcomes. Additionally, Polaris-C has a higher ratio of accepted transitions than Polaris-B, thus it explores the state space more extensively. Due to frequent *Out of Space* outcomes, Polaris-B has to frequently resample new DESs, leading to increased running

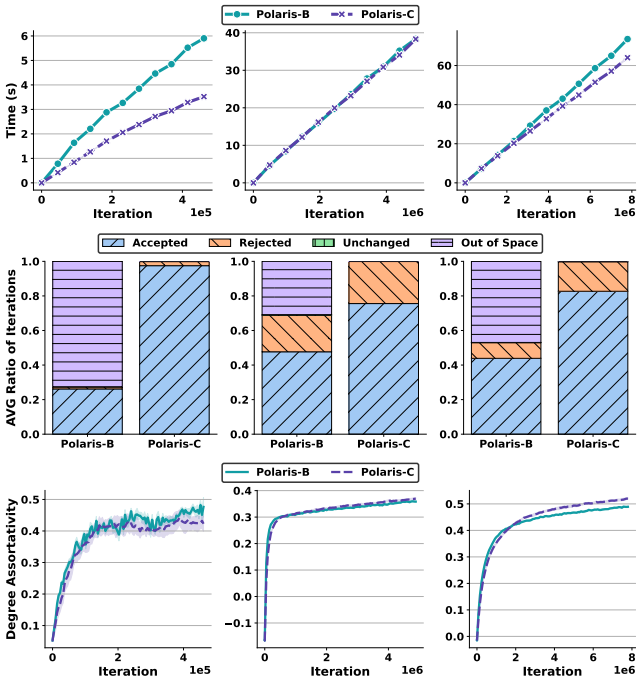

Figure 5: Running time (top), average ratio of iterations for each of the four possible outcomes (mid), and degree assortativity as a function of the number of iterations (bottom) for each sampler on Cite (left), Brexit (middle), and Twitter (right).

times per iteration (as shown in the first panel). This effect becomes particularly evident as the number of colors increases.

The bottom panel illustrates the *degree* assortativity of the states visited in the Markov chains produced by each sampler, which is often used as a measure of convergence for the chain [39]. Both algorithms reach a plateau at nearly the same point, which suggests the states visited start to share similar characteristics. However, as shown in the first plot, Polaris-C achieves this plateau faster.

Figure 6 provides a detailed analysis of the time required to perform a step in the sampling process, categorized by the type of outcome. Specifically, the left chart shows the times for transitions that are accepted, whereas the middle chart illustrates the times for transitions that are rejected. An accepted transition corresponds to the outcome *Accepted*, while a rejected transition include the outcomes *Rejected* and *Unchanged*. The right chart displays the distribution of total time for steps where the sampled DES is not a JDES (i.e., *Out of Space*). On average, accepting a transition is 2× slower than rejecting it, because the algorithms must update the data structures that store the edges and their weights to maintain correctness after an accepted transition.

Polaris-C has higher step times compared to Polaris-B as

- Polaris-C performs more computations even for rejected steps, as it must evaluate several quantities to compute the value of ρ. In contrast, Polaris-B performs fewer computations before rejection, and thus achieves lower running times;
- when a transition is accepted, Polaris-C needs to maintain additional data structures necessary to ensure that the sampled DES is always a JDES.

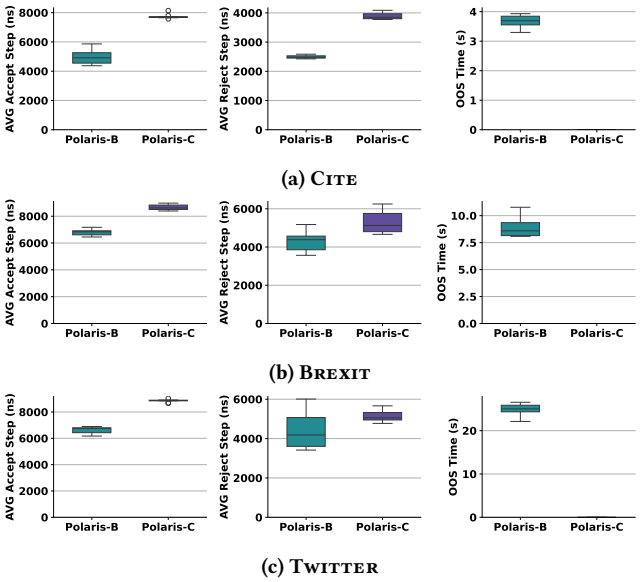

Figure 6: Distribution of the average time required to perform a step where the transition to the next state is accepted (left) or rejected (middle). The right plots show the total time required to find a DES that is a JDES. Results for 10 Markov chains.

Nonetheless, Polaris-B results in longer overall running times because it uses a considerable amount of time to find a DES that is a JDES, especially when the number of colors is higher.

6 Conclusion

We introduced Polaris, an ensemble of colored multigraphs with prescribed Joint Color Matrix. The JCM captures key properties relevant to the study of polarized networks, such as color assortativity. We described two efficient algorithms to sample from the space of such multigraphs according to any user-specified probability distribution over this space. Our algorithms work by running a Markov chain on the multigraph space, following the Metropolis-Hastings approach for determining whether to accept the move to a proposed neighbor of the current state. We conducted an extensive experimental evaluation, showing the shortcomings of existing methods in capturing the color assortativity, and assessing the performance of our algorithms across different datasets in terms of scalability, runtime, and acceptance probability.

This work serves as an important first step toward analyzing polarization in real networks. While a comprehensive study of polarization lies beyond this paper's scope, the tools developed here lay the groundwork for future studies on this subject.

Acknowledgments

MR's work is supported by the National Science Foundation grants IIS-2006765 (https://www.nsf.gov/awardsearch/showAward?AWD_ID=2006765), and CAREER-2238693 (https://www.nsf.gov/awardsearch/showAward?AWD_ID=2238693). AG's work is supported by the ERC Advanced Grant REBOUND (834862), the EC H2020 RIA project SoBigData++ (871042), and the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

References

[1] Yael Artzy-Randrup and Lewi Stone. 2005. Generating uniformly distributed random networks. *Physical Review E* 72, 5 (2005), 056708. (Cited on 3)

[2] Delia Baldassarri and Scott E. Page. 2021. The emergence and perils of polarization. *Proceedings of the National Academy of Sciences* 118, 50 (2021), e2116863118. https://doi.org/10.1073/pnas.2116863118 (Cited on 1)

[3] Albert-László Barabási and Réka Albert. 1999. Emergence of scaling in random networks. *science* 286, 5439 (1999), 509–512. (Cited on 2)

[4] Edward A Bender and E.Rodney Canfield. 1978. The asymptotic number of labeled graphs with given degree sequences. *Journal of Combinatorial Theory, Series A* 24, 3 (May 1978), 296–307. https://doi.org/10.1016/0097-3165(78)90059-6 (Cited on 1, 2)

[5] Béla Bollobás. 1980. A Probabilistic Proof of an Asymptotic Formula for the Number of Labelled Regular Graphs. *European Journal of Combinatorics* 1, 4 (Dec. 1980), 311–316. https://doi.org/10.1016/S0195-6698(80)80030-8 (Cited on 1, 2)

[6] Béla Bollobás, Oliver Riordan, Joel Spencer, and Gábor Tusnády. 2001. The degree sequence of a scale-free random graph process. *Random Structures & Algorithms* 18, 3 (2001), 279–290. (Cited on 2)

[7] Sonia Cafieri, Pierre Hansen, and Leo Liberti. 2010. Loops and multiple edges in modularity maximization of networks. *Physical Review E* 81, 4 (2010), 046102. (Cited on 3)

[8] Matteo Cinelli, Gianmarco De Francisci Morales, Alessandro Galeazzi, Walter Quattrociocchi, and Michele Starnini. 2021. The Echo Chamber Effect on Social Media. *Proceedings of the National Academy of Sciences* 118, 9 (2021), e2023301118. https://doi.org/10.1073/pnas.2023301118 (Cited on 1)

[9] Edward F Connor and Daniel Simberloff. 1979. The assembly of species communities: chance or competition? *Ecology* 60, 6 (1979), 1132–1140. (Cited on 2)

[10] Michael D Conover, Jacob Ratkiewicz, Matthew Francisco, Bruno Gonçalves, Filippo Menczer, and Alessandro Flammini. 2011. Political Polarization on Twitter. In *International AAAI Conference on Weblogs and Social Media (ICWSM)*. 89–96. (Cited on 1)

[11] Colin Cooper, Martin Dyer, and Catherine Greenhill. 2007. Sampling regular graphs and a peer-to-peer network. *Combinatorics, Probability and Computing* 16, 4 (2007), 557–593. (Cited on 2)

[12] Upasana Dutta, Bailey K. Fosdick, and Aaron Clauset. 2021. Sampling random graphs with specified degree sequences. https://doi.org/10.48550/ARXIV.2105.12120 Version Number: 4. (Cited on 2)

[13] Paul Erdős and Alfréd Rényi. 1959. On random graphs. *Publicationes Mathematicate* 6 (1959), 290–297. (Cited on 2)

[14] Péter L. Erdős, Catherine Greenhill, Tamás Róbert Mezei, István Miklós, Daniel Soltész, and Lajos Soukup. 2022. The mixing time of switch Markov chains: a unified approach. *European Journal of Combinatorics* 99 (2022), 103421. (Cited on 5)

[15] Ronald Aylmer Fisher. 1935. The design of experiments. Oliver and Boyd. *Edinburgh* (1935). (Cited on 2)

[16] World Economic Forum. 2024. *The Global Risks Report 2024.* Technical Report. https://www.weforum.org/publications/global-risks-report-2024 (Cited on 1)

[17] Bailey K. Fosdick, Daniel B. Larremore, Joel Nishimura, and Johan Ugander. 2017. Configuring Random Graph Models with Fixed Degree Sequences. arXiv:1608.00607 (Cited on 2, 4, 5)

[18] Bailey K. Fosdick, Daniel B. Larremore, Joel Nishimura, and Johan Ugander. 2018. Configuring Random Graph Models with Fixed Degree Sequences. *Siam Review* 60, 2 (2018), 315–355. (Cited on 1, 3, 4)

[19] Kiran Garimella, Gianmarco De Francisci Morales, Aristides Gionis, and Michael Mathioudakis. 2016. Quantifying Controversy in Social Media. In *ACM International Conference on Web Search and Data Mining (WSDM)*. 33–42. (Cited on 1)

[20] Kiran Garimella, Gianmarco De Francisci Morales, Aristides Gionis, and Michael Mathioudakis. 2017. Reducing Controversy by Connecting Opposing Views. In *ACM International Conference on Web Search and Data Mining (WSDM)*. 81–90. (Cited on 1)

[21] Kiran Garimella, Gianmarco De Francisci Morales, Aristides Gionis, and Michael Mathioudakis. 2018. Political discourse on social media: Echo chambers, gatekeepers, and the price of bipartisanship. In *Proceedings of the World Wide Web Conference (WWW)*. 913–922. (Cited on 1)

[22] Kiran Garimella, Gianmarco De Francisci Morales, Aristides Gionis, and Michael Mathioudakis. 2018. Quantifying Controversy on Social Media. *ACM Transactions on Social Computing* 1, 1 (2018), 3. (Cited on 1)

[23] Andrew Gelman, John B Carlin, Hal S Stern, and Donald B Rubin. 1995. *Bayesian data analysis.* Chapman and Hall/CRC. (Cited on 2)

[24] Matthew Gentzkow and Jesse M. Shapiro. 2011. Ideological Segregation Online and Offline. *The Quarterly Journal of Economics* 126, 4 (2011), 1799–1839. https://doi.org/10.1093/qje/qjr044 (Cited on 1)

[25] Sandra González-Bailón, David Lazer, Pablo Barberá, Meiqing Zhang, Hunt Allcott, Taylor Brown, Adriana Crespo-Tenorio, Deen Freelon, Matthew Gentzkow, Andrew M. Guess, Shanto Iyengar, Young Mie Kim, Neil Malhotra, Devra Moehler, Brendan Nyhan, Jennifer Pan, Carlos Velasco Rivera, Jaime Settle, Emily Thorson, Rebekah Tromble, Arjun Wilkins, Magdalena Wojcieszak, Chad Kiewiet De Jonge, Annie Franco, Winter Mason, Natalie Jomini Stroud, and Joshua A. Tucker. 2023. Asymmetric ideological segregation in exposure to political news on Facebook. *Science* 381, 6656 (2023), 392–398. https://doi.org/10.1126/science.ade7138 (Cited on 1)

[26] Nicholas J. Gotelli and Gary R. Graves. 1996. *Null models in ecology.* Smithsonian Institution Press. (Cited on 3)

[27] Catherine Greenhill. 2014. The switch Markov chain for sampling irregular graphs. In *Proceedings of the 26th annual ACM-SIAM Symposium on Discrete Algorithms*. 1564–1572. (Cited on 2)

[28] Andrew M. Guess, Neil Malhotra, Jennifer Pan, Pablo Barberá, Hunt Allcott, Taylor Brown, Adriana Crespo-Tenorio, Drew Dimmery, Deen Freelon, Matthew Gentzkow, Sandra González-Bailón, Edward Kennedy, Young Mie Kim, David Lazer, Devra Moehler, Brendan Nyhan, Carlos Velasco Rivera, Jaime Settle, Daniel Robert Thomas, Emily Thorson, Rebekah Tromble, Arjun Wilkins, Magdalena Wojcieszak, Beixian Xiong, Chad Kiewiet De Jonge, Annie Franco, Winter Mason, Natalie Jomini Stroud, and Joshua A. Tucker. 2023. Reshares on social media amplify political news but do not detectably affect beliefs or opinions. *Science* 381, 6656 (2023), 404–408. https://doi.org/10.1126/science.add8424 (Cited on 1)

[29] Marilena Hohmann, Karel Devriendt, and Michele Coscia. 2023. Quantifying ideological polarization on a network using generalized Euclidean distance. *Science Advances* 9, 9 (2023), eabq2044. https://doi.org/10.1126/sciadv.abq2044 Publisher: American Association for the Advancement of Science. (Cited on 1)

[30] Miller McPherson, Lynn Smith-Lovin, and James M Cook. 2001. Birds of a feather: Homophily in social networks. *Annual review of sociology* 27, 1 (2001), 415–444. (Cited on 2)

[31] Solomon Messing and Sean J. Westwood. 2014. Selective Exposure in the Age of Social Media: Endorsements Trump Partisan Source Affiliation When Selecting News Online. *Communication Research* 41, 8 (2014), 1042–1063. https://doi.org/10.1177/0093650212466406 (Cited on 1)

[32] Ron Milo, Shai Shen-Orr, Shalev Itzkovitz, Nadav Kashtan, Dmitri Chklovskii, and Uri Alon. 2002. Network motifs: simple building blocks of complex networks. *Science* 298, 5594 (2002), 824–827. (Cited on 2)

[33] Michael Mitzenmacher and Eli Upfal. 2005. *Probability and Computing: Randomized Algorithms and Probabilistic Analysis.* Cambridge University Press. (Cited on 3)

[34] Jacob L Moreno and Helen H Jennings. 1938. Statistics of social configurations. *Sociometry* (1938), 342–374. (Cited on 2)

[35] Mark Newman. 2018. *Networks.* Oxford university press. (Cited on 6)

[36] M. E. J. Newman. 2003. Mixing patterns in networks. *Physical Review E* 67, 2 (2003), 026126. https://doi.org/10.1103/PhysRevE.67.026126 (Cited on 2)

[37] Brendan Nyhan, Jaime Settle, Emily Thorson, Magdalena Wojcieszak, Pablo Barberá, Annie Y. Chen, Hunt Allcott, Taylor Brown, Adriana Crespo-Tenorio, Drew Dimmery, Deen Freelon, Matthew Gentzkow, Sandra González-Bailón, Andrew M. Guess, Edward Kennedy, Young Mie Kim, David Lazer, Neil Malhotra, Devra Moehler, Jennifer Pan, Daniel Robert Thomas, Rebekah Tromble, Carlos Velasco Rivera, Arjun Wilkins, Beixian Xiong, Chad Kiewiet De Jonge, Annie Franco, Winter Mason, Natalie Jomini Stroud, and Joshua A. Tucker. 2023. Like-minded sources on Facebook are prevalent but not polarizing. *Nature* 620, 7972 (2023), 137–144. https://doi.org/10.1038/s41586-023-06297-w (Cited on 1)

[38] Peter H Peskun. 1973. Optimum monte-carlo sampling using markov chains. *Biometrika* 60, 3 (1973), 607–612. (Cited on 2, 5)

[39] Vivekananda Roy. 2019. Convergence diagnostics for Markov chain Monte Carlo. http://arxiv.org/abs/1909.11827 arXiv:1909.11827 [stat]. (Cited on 2, 8)

[40] Ali Salloum, Ted Hsuan Yun Chen, and Mikko Kivelä. 2022. Separating Polarization from Noise: Comparison and Normalization of Structural Polarization Measures. *Proceedings of the ACM on Human-Computer Interaction* 6, CSCW1 (2022), 1–33. https://doi.org/10.1145/3512962 (Cited on 1)

[41] Tom Snijders, et al. 2002. Markov chain Monte Carlo estimation of exponential random graph models. *Journal of Social Structure* 3, 2 (2002), 1–40. (Cited on 2)

[42] Tiziano Squartini and Diego Garlaschelli. 2017. *Maximum-entropy networks: Pattern detection, network reconstruction and graph combinatorics.* Springer. (Cited on 2)

[43] Lewi Stone and Alan Roberts. 1990. The checkerboard score and species distributions. *Oecologia* 85 (1990), 74–79. (Cited on 3)

[44] Richard Taylor. 2006. Contrained switchings in graphs. In *Proceedings of the Eighth Australian Conference on Combinatorial Mathematics.* Springer, 314–336. (Cited on 3)

[45] Petter Törnberg. 2022. How digital media drive affective polarization through partisan sorting. *Proceedings of the National Academy of Sciences* 119, 42 (2022), e2207159119. https://doi.org/10.1073/pnas.2207159119 (Cited on 1)

[46] Norman D. Verhelst. 2008. An efficient MCMC algorithm to sample binary matrices with fixed marginals. *Psychometrika* 73, 4 (2008), 705–728. (Cited on 3)

7 Ethical considerations

Our paper introduces a new method for assessing the statistical significance of the polarization structure discovered in online social networks. The motivation of our work is the study of social phenomena, such as polarization, formation of echo chambers, opinion dynamics, and influence among individuals in online social networks. As such, our work contributes to the growing field of computational social science, which in turn contributes to a better understanding of complex social behavior. Our emphasis in this work is on the design of new algorithms for efficient sampling from a novel network null model and the mathematical analysis of the algorithms and their properties. During our study, we did not perform any data collection, and our empirical evaluation uses benchmark graph datasets that are publicly available. For future studies and researchers who would like to apply our method on new data collected from online social networks, we emphasize the importance of prioritizing ethical considerations to protect the privacy and rights of individuals. This involves obtaining informed consent where possible, anonymizing data to prevent the identification of users, applying the data minimization principle, and being mindful of the potential for harm in the analysis and dissemination of findings. It is also important to comply with platform policies and legal regulations, such as GDPR.