Journal of Machine Learning Research 24 (2023) 1-79 Submitted 5/20; Revised 12/22; Published 1/23

Sampling random graph homomorphisms and applications to network data analysis

Hanbaek Lyu* HLYU@MATH.WISC.EDU
Department of Mathematics
University of Wisconsin - Madison, WI, USA 53706

Facundo Mémoli MEMOLI@MATH.OSU.EDU
Department of Mathematics
The Ohio State University, Columbus, OH 43210, USA

David Sivakoff DSIVAKOFF@STAT.OSU.EDU
Department of Statistics and Department of Mathematics
The Ohio State University, Columbus, OH 43210

Editor: Francis Bach

Abstract

A graph homomorphism is a map between two graphs that preserves adjacency relations. We consider the problem of sampling a random graph homomorphism from a graph into a large network. We propose two complementary MCMC algorithms for sampling random graph homomorphisms and establish bounds on their mixing times and the concentration of their time averages. Based on our sampling algorithms, we propose a novel framework for network data analysis that circumvents some of the drawbacks in methods based on independent and neighborhood sampling. Various time averages of the MCMC trajectory give us various computable observables, including well-known ones such as homomorphism density and average clustering coefficient and their generalizations. Furthermore, we show that these network observables are stable with respect to a suitably renormalized cut distance between networks. We provide various examples and simulations demonstrating our framework through synthetic networks. We also demonstrate the performance of our framework on the tasks of network clustering and subgraph classification on the Facebook100 dataset and on Word Adjacency Networks of a set of classic novels.

Keywords: Networks, sampling, graph homomorphism, MCMC, graphons, stability inequalities, hierarchical clustering, subgraph classification

1. Introduction

Over the past several decades, technological advances in data collection and extraction have fueled an explosion of network data from seemingly all corners of science – from computer science to the information sciences, from biology and bioinformatics to physics, and from economics to sociology. These data sets come with a locally defined pairwise relationship, and the emerging and interdisciplinary field of Network Data Analysis aims at systematic methods to analyze such network data at a systems level, by combining various techniques from probability, statistics, graph theory, geometry, and topology.

∗. All codes are available at https://github.com/HanbaekLyu/motif_sampling

©2023 Hanbaek Lyu and Facundo Mémoli and David Sivakoff.

License: CC-BY 4.0, see https://creativecommons.org/licenses/by/4.0/. Attribution requirements are provided at http://jmlr.org/papers/v24/20-449.html.

Sampling is an indispensable tool in the statistical analysis of large graphs and networks. Namely, we select a typical sample of the network and calculate its graph theoretical properties such as average degree, mean shortest path length, and expansion (see Kolaczyk and Csárdi (2014) for a survey of statistical methods for network data analysis). One of the most fundamental sampling methods, which is called the *independent sampling*, is to choose a fixed number of nodes independently at random according to some distribution on the nodes. One then studies the properties of the subgraph or subnetwork induced on the sample. Independent sampling is suitable for dense graphs, and closely connected to the class of network observables called the *homomorphism density*, which were the central thread in the recent development of the theory of dense graph limits and graphons (Lovász and Szegedy, 2006; Lovász, 2012).

An alternative sampling procedure particularly suitable for sparse networks is called the *neighborhood sampling* (or snowball sampling). Namely, one may pick a random node and sample its entire neighborhood up to some fixed radius, so that we are guaranteed to capture a connected local piece of the sparse network. We then ask what the given network looks like locally. For instance, the average clustering coefficient, first introduced in Watts and Strogatz (1998), is a network observable that measures the extent to which a given network locally resembles complete graphs. Also, neighborhood sampling was used in Benjamini et al. (2001) to define the sampling distance between networks and to define the limit object of sequences of bounded degree networks.

Our primary concern in this work, roughly speaking, is to *sample connected subgraphs from a possibly sparse network in a way such that certain minimal structure is always imposed.* A typical example is to sample k-node subgraphs with uniformly random Hamiltonian path, see Section 2.2. More generally, for a fixed 'template graph' (motif) F of k nodes, we would like to sample k nodes from the network $\mathcal{G}$ so that the induced subnetwork always contains a 'copy' of F. This is equivalent to conditioning the independent sampling to contain a 'homomorphic copy' of F. This conditioning enforces that we are always sampling some meaningful portion of the network, where the prescribed motif F serves as a backbone. One can then study the properties of subnetworks of $\mathcal{G}$ induced on this random copy of F. Clearly, neither independent sampling nor neighborhood sampling serve this purpose, as the former returns disconnected subgraphs with high probability (due to the sparsity of the network) and the latter has no control over the structure of the subgraphs being sampled. We call this sampling scheme *motif sampling* (see Figure 5) and it should not be confused with sampling graphs from a random graph model.

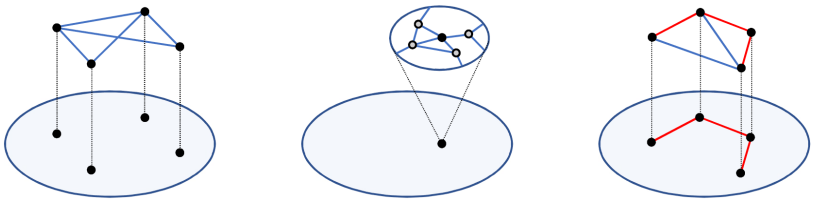

Figure 1: Independent sampling (left), neighborhood sampling (middle), and motif sampling (right).

Once we have developed sufficient mathematical and computational foundations for the motif sampling problem, we will use them to devise computationally efficient and stable

network observables. As the typical size and complexity of network data far exceed the capabilities of human perception, we need some lens through which we can study and analyze network data. Namely, given a network $\mathcal{G}$, we want to associate a much simpler object $f(\mathcal{G})$, which we call a *network observable*, such that it can be computed in a reasonable amount of time even when $\mathcal{G}$ is large and complex, and yet it retains substantial information about $\mathcal{G}$. These two desired properties of network observables are stated more precisely below:

(i) (*Computability*) The observable $f(\mathcal{G})$ is computable in at most polynomial time in the size of the network $\mathcal{G}$.

(ii) (*Stability*) For two given networks $\mathcal{G}_1, \mathcal{G}_2$, we have

$$d(f(\mathcal{G}_1), f(\mathcal{G}_2)) \leq d(\mathcal{G}_1, \mathcal{G}_2), \tag{1}$$

where d on each side denotes a suitable distance metric between observables and between networks, respectively.

An inequality of type (1) is called a 'stability inequality' for the observable $f(\mathcal{G})$, which encodes the property that a small change in the network yields small change in the observable.

1.1 Our approach and contribution

We summarize our approach and contributions in the following bullet points.

- We propose a new network sampling framework based on sampling a graph homomorphism from a small template network F into a large target network $\mathcal{G}$.
- We propose two complementary MCMC algorithms for sampling random graph homomorphisms and establish bounds on their mixing times and concentration of their time averages.
- Based on our sampling algorithms, we propose a number of network observables that are both easy to compute (using our MCMC motif-sampling algorithms) and provably stable.
- We demonstrate the efficacy of our techniques through various synthetic and real-world networks. For instance, for subgraph classification problems on Facebook social networks, our Matrix of Average Clustering Coefficient (MACC) achieves performance better than the benchmark methods (see Figure 2 and Section 6).

The key insight in our approach is to sample adjacency-preserving functions from small graphs to large networks, instead of directly sampling subgraphs. Namely, suppose $\mathcal{G} = (V, E_{\mathcal{G}})$ is a large and possibly sparse graph and $F = (\{1, \dots, k\}, E_F)$ is a k-node template graph. A vertex map $\mathbf{x} : \{1, \dots, k\} \to V$ is said to be a (graph) *homomorphism* $F \to \mathcal{G}$ if it preserves adjacency relations, that is, $\mathbf{x}(i)$ and $\mathbf{x}(j)$ are adjacent in $\mathcal{G}$ if i and j are adjacent in F. Our main goal then becomes the following:

$$\textit{Sample a graph homomorphism } \mathbf{x} : F \to \mathcal{G} \textit{ uniformly at random.} \tag{2}$$

We consider the above problem in the general context where $\mathcal{G}$ is a network with edge weights equipped with a probability distribution on the nodes.

To tackle the homomorphism sampling problem (2), we propose two complementary Markov Chain Monte Carlo algorithms. In other words, the algorithms proceed by sampling a Markov chain of graph homomorphisms $\mathbf{x}_t : F \to \mathcal{G}$ in a way such that the empirical distribution of $\mathbf{x}_t$ converges to the desired target distribution.

Our network observables based on motif sampling will be of the following form:

$$f(\mathcal{G}) := \mathbb{P}(A \textit{ uniformly random homomorphism } \mathbf{x} : F \to \mathcal{G} \textit{ satisfies a property } P). \quad (3)$$

For instance, the well-known *average clustering coefficient* network observable can be realized in the form above (see Example 3.1), which we generalize to *conditional homomorphism densities* (see Section 3.1). By taking the expectation of some function of the random homomorphism $\mathbf{x}$, we can also define not only real-valued network observables, but also function- (see Figure 3), matrix- (see Figure 2), and even network-valued observables. These observables can all be efficiently (and provably) computed by taking suitable time averages along the MCMC trajectory of the MCMC motif sampling procedure (see Theorems 2.6 and 2.7). Furthermore, we establish that these network observables are stable in the sense that a small change in the network results in a small change in their values (see Section 4). Our new network observables are not vanishingly small for sparse networks and are able to capture multi-scale features. Moreover, they can directly be applied to comparing networks with different sizes without node labels (e.g., comparing two social networks with anonymous users or brain networks of two species) with low computational cost.

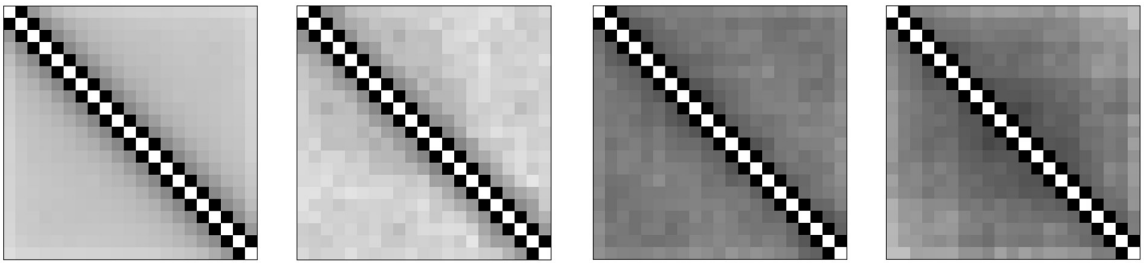

Figure 2: Matrices of Average Clustering Coefficients of the Facebook network corresponding to four schools in the Facebook100 dataset using the chain motif of 21 nodes. The 21×21 matrices are summarizing observables of the corresponding Facebook networks. See Figure 15 for more details.

To demonstrate our new sampling technique and Network Data analysis framework, we apply our framework for network clustering and classification problems using the Facebook100 dataset and Word Adjacency Networks of a set of classic novels. Our new matrix-valued network observable compresses a given network of arbitrary size without node label into a fixed size matrix, which reveals local clustering structures of the network in any desired scale (see Figure 2). We use these low-dimensional representations to perform subgraph classification and hierarchical clustering of the 100 network data. For the former supervised task, our proposed method shows significantly better performance than the baseline methods. On the other hand, we analyze the hierarchical structure of weighted networks representing text

data using our function-valued observable. The obtained observables indicate similar hierarchical structures among different texts by the same author that are distinctive between different authors.

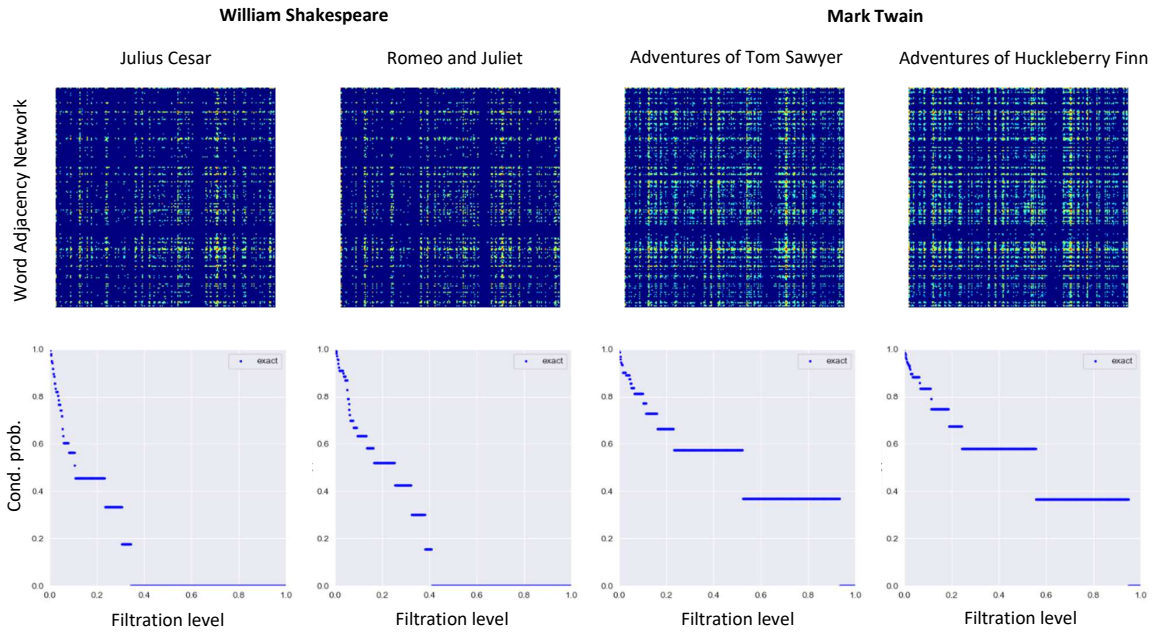

Figure 3: Heat map of the Word Adjacency Networks of four novels and their CHD profiles corresponding to the pair of motifs $(H_{0,0}, H_{0,0})$, $H_{0,0} = (\{0\}, \mathbf{1}_{\{(0,0)\}})$. The non-increasing functions in the second row summarize the observables of the networks shown in the first row. See Section 7.1 for details.

1.2 Background and related work

The motif sampling problem from (2) generalizes the well-known problem of sampling a proper coloring of a given graph uniformly at random. Recall that a proper q-coloring of a simple graph $G = (V, E)$ is an assignment of colors $\mathbf{x} : V \to \{1, \dots, q\}$ such that $\mathbf{x}(u) \neq \mathbf{x}(v)$ whenever nodes u and v are adjacent in G. This is in fact a graph homomorphism $G \to K_q$, where K_q is the complete graph of q nodes. Indeed, in order to preserve the adjacency, any two adjacent nodes in G should *not* be mapped into the same node in K_q. A number of MCMC algorithms and their mixing times to sample a uniform q-coloring of a graph have been studied in the past few decades (Jerrum, 1995; Salas and Sokal, 1997; Vigoda, 2000; Dyer et al., 2002; Frieze and Vigoda, 2007). One of our MCMC motif sampling algorithms, the Glauber chain (see Definition 2.1), is inspired by the standard Glauber dynamics for sampling proper q-coloring of graphs.

There is an interesting change of perspective between the graph coloring problem and motif sampling. Namely, in graph coloring $G \to K_q$, the problem becomes easier for large q and hence the attention is toward sampling a random q-coloring for small q. On the other hand, for motif sampling, our goal is to analyze large network $\mathcal{G}$ through a random homomorphism $F \to \mathcal{G}$ from a relatively small motif F. It will be conceptually helpful to

visualize a homomorphism $F \to \mathcal{G}$ as a graph-theoretic embedding of a motif F into the large network $\mathcal{G}$.

Our work on constructing stable network observables from motif sampling algorithms is inspired by the graph homomorphism and graph limit theory (see, e.g., Lovász and Szegedy (2006); Lovász (2012)), and by methods from Topological Data Analysis (see, e.g., Carlsson (2009); Edelsbrunner and Harer (2010)), which considers the hierarchical structure of certain observables and studies their stability properties.

For an illustrative example, let $G = (V, E)$ be a finite simple graph and let K_3 be a triangle. Choose three nodes x_1, x_2, x_3 independently from V uniformly at random, and define an observable $\mathtt{t}(K_3, G)$, which is called the homomorphism density of K_3 in G, by

$$\mathtt{t}(K_3, G) := \mathbb{P}(\text{there is an edge between } x_i \text{ and } x_j \text{ for all } 1 \le i < j \le 3). \tag{4}$$

In words, this is the probability that three randomly chosen people from a social network are friends of each other. If we replace the triangle K_3 with an arbitrary simple graph F, a similar observable $\mathtt{t}(F, G)$ can be defined. Note that computing such observables can be done by repeated sampling and averaging. Moreover, a fundamental lemma due to Lovász and Szegedy (2006) asserts that the homomorphism densities are stable with respect to the cut distance between graphs (or graphons, in general):

$$|\mathtt{t}(F, G_1) - \mathtt{t}(F, G_2)| \le |E_F| \cdot \delta_\square(G_1, G_2), \tag{5}$$

where G_1, G_2 are simple graphs and E_F is the set of edges in F. Hence by varying F, we obtain a family of observables that satisfy the computability and stability (note that we can absorb the constant $|E_F|$ into the cut distance $\delta_\square$).

However, there are two notable shortcomings of homomorphism densities as network observables. First, they provide no useful information for sparse networks, where the average degree is of order sublinear in the number of nodes (e.g., two-dimensional lattices, trees, most real-world social networks (Barabási, 2013; Newman, 2018a)). This is because for sparse networks the independent sampling outputs a set of non-adjacent nodes with high probability. In terms of the stability inequality (5), this is reflected in the fact that the cut distance $\delta_\square$ between two sparse networks becomes asymptotically zero as the sizes of networks tend to infinity. Second, homomorphism densities do not capture hierarchical features of weighted networks. Namely, we might be interested in how the density of triangles formed through edges of weights at least t changes as we increase the parameter t. But the homomorphism density of triangles aggregates such information into a single numeric value, which is independent of t.

An entirely different approach is taken in the fields of Topological Data Analysis (TDA) in order to capture multi-scale features of data sets (Carlsson, 2009; Edelsbrunner and Harer, 2010). The essential workflow in TDA is as follows. First, a data set X consisting of a finite number of points in Euclidean space $\mathbb{R}^d$ is given. In order to equip the data set with a topological structure, one constructs a filtration of simplicial complexes on top of X by attaching a suitable set of high-dimensional cells according to the filtration parameter (spatial resolution). Then by computing the homology of the filtration (or the persistent homology of X), one can associate X with a topological invariant $f(X)$ called the persistence diagram (Edelsbrunner et al., 2000) (or barcodes (Ghrist, 2008)). The stability of such

observable is well-known (Cohen-Steiner et al., 2007; Chazal et al., 2009). Namely, it holds that

$$d_B(f(X), f(Y)) \leq d_{GH}(X, Y), \tag{6}$$

where the distance metric on the left and right-hand side denotes the bottleneck distance between persistence diagrams and the Gromov-Hausdorff distance between data sets X and Y viewed as finite metric spaces. However, as is well known in the TDA community, computing persistence diagrams for large data sets is computationally expensive (see Edelsbrunner et al. (2000); Zomorodian and Carlsson (2005) for earlier algorithms and Carlsson (2009); Edelsbrunner and Morozov (2012); Otter et al. (2017); Mémoli and Singhal (2019) for recent surveys).

Whereas in the present work we concentrate on *symmetric networks*, where the edge weight between two nodes x and y does not depend on their ordering, we acknowledge that in the context of asymmetric networks, several possible observables f and a suitable metric are studied in (Chowdhury and Mémoli, 2018a, 2017, 2018b; Turner, 2019; Chowdhury and Mémoli, 2018c, 2019).

We also remark that an earlier version of the present work has already found several applications in the literature of network data analysis. The MCMC motif sampling algorithms as well as their theoretical guarantees were used as a key component in the recent network dictionary learning methods of (Lyu et al., 2021, 2020; Peng et al., 2022). Also, a MCMC k-path sampling algorithm was used to generate sub-texts within knowledge graphs for topic modeling applications (Alaverdian et al., 2020). The same algorithm was used to benchmark stochastic proximal gradient descent algorithms for Markovian data in (Alacaoglu and Lyu, 2022).

1.3 Organization

We formally introduce the motif sampling problem on edge and node weighted networks in Section 2.1 and discuss a concrete example of such sampling scheme in the form of subgraph sampling via Hamiltonian paths in Section 2.2. In Section 2.3, we introduce two Markov chain Monte Carlo (MCMC) algorithms for motif sampling. Their convergence is stated in Theorems 2.1 and 2.2 and their mixing time bounds are stated in Theorems 2.4 and 2.5. We also deduce that the expected value of various functions of the random homomorphism can be efficiently computed by time averages of the MCMC trajectory (see Corollary 3.1). Moreover, these estimates are guaranteed to be close to the expected value according to the concentration inequalities that we obtain in Theorems 2.6 and 2.7.

In Section 3.1, we introduce four network observables (Conditional Homomormorphism Density, Matrix of Average Clustering Coefficients, CHD profile, and motif transform) by taking the expected value of suitable functions of random homomorphism $F \to \mathcal{G}$. We also provide some elementary examples. In Section 4, we state stability inequalities (Propositions 4.1, 4.2, and Theorem 4.1) for our network observables using the language of graphons and the cut distance.

Sections 5 and 7 are devoted to examples and applications of our framework. In Section 5, we provide various examples and simulations demonstrating our results on synthetic networks. In Section 6, we apply our methods to the Facebook social network for the tasks of

subgraph classification and hierarchical clustering. In Section 7, we apply our framework to analyze Word Adjacency Networks of a set consisting of 45 novels and propose an authorship attribution scheme using motif sampling and conditional homomorphism profiles.

Finally, we provide additional discussions, examples, proofs, and figures in the appendices. In Appendix A, we discuss the relationship between motif transforms and spectral analysis. In Appendices B and C, we prove convergence, mixing time bounds, and concentration of the MCMC algorithms as well as the stability inequalities of our network observables.

1.4 Notation

For each integer $n \geq 1$, we write $[n] = \{1, 2, \cdots, n\}$. Given a matrix $A : [n]^2 \to [0, \infty)$, we call the pair $G = ([n], A)$ an *edge-weighted graph* with *node set* $[n]$ and edge weight A. When A is 0-1 valued, we call G a *directed graph* and we also write $G = ([n], E)$, where $E = \{(i, j) \in [n]^2 \,|\, A(i, j) = 1\}$ is the set of all directed edges. If A is 0-1 valued, symmetric, and has all diagonal entries equal to 0, then we call G a *simple graph*. Given an edge-weighted graph $G = ([n], A)$, define its *maximum degree* by

$$\Delta(G) = \max_{a \in [n]} \sum_{b \in [n]} \mathbf{1}\big(A(a, b) + A(b, a) > 0\big). \tag{7}$$

A sequence $(x_j)_{j=0}^m$ of nodes in G is called a *walk of length* m if $A(x_j, x_{j+1}) > 0$ for all $0 \leq j < m$. A walk is a *path* if all nodes in the walk are distinct. We define the *diameter* of G, which we denote by $\mathrm{diam}(G)$, by

$$\mathrm{diam}(G) = \max_{a,b \in [n]} \min\{k \geq 0 \,|\, \exists \text{ a path of length } k \text{ between } a \text{ and } b\}. \tag{8}$$

We let $\mathrm{diam}(G) = \infty$ if there is no path between some $x, y \in [n]$.

For an event B, we let $\mathbf{1}_B$ denote the indicator function of B, where $\mathbf{1}_B(\omega) = 1$ if $\omega \in B$ and 0 otherwise. We also write $\mathbf{1}_B = \mathbf{1}(B)$ when convenient. For two real numbers $a, b \in \mathbb{R}$, we write $a \vee b = \max(a, b)$ and $a \wedge b = \min(a, b)$.

2. Motif sampling and MCMC sampling algorithms

2.1 Random homomorphism from motifs into networks

To describe motif sampling, we first give precise definitions of networks and motifs. A *network* as a mathematical object consists of a triple $\mathcal{G} = (X, A, \alpha)$, where X, a finite set, is the node set of individuals, $A : X^2 \to [0, \infty)$ is a matrix describing interaction strength between individuals, and $\alpha : X \to (0, 1]$ is a probability measure on X giving the significance of each individual (cf. Chowdhury and Mémoli (2019)). Any given $(n \times n)$ matrix A taking values from $[0, 1]$ can be regarded as a network $([n], A, \alpha)$ where $\alpha(i) \equiv 1/n$ is the uniform distribution on $[n]$.

Fix an integer $k \geq 1$ and a matrix $A_F : [k]^2 \to [0, \infty)$. Let $F = ([k], A_F)$ denote the corresponding edge-weighted graph, which we also call a *motif*. A motif $F = ([k], A_F)$ is said to be *simple* if A_F is 0-1 valued, has zero diagonal entries (no loops), and $A_F(i, j) + A_F(j, i) \in$

$\{0,1\}$ for each $1 \le i < j \le k$ (see Figure 4 for an illustration). The fact that simple motifs have at most one directed edge between any pair of nodes is crucial in the proof of stability inequalities of the network observables stated in Section 4. A particularly important motif for application purposes is the *k-chain*, which is the pair $([k], \mathbf{1}_{\{(1,2),(2,3),\ldots,(k-1,k)\}})$. It corresponds to the direct path on k nodes, see Figure 4 (c).

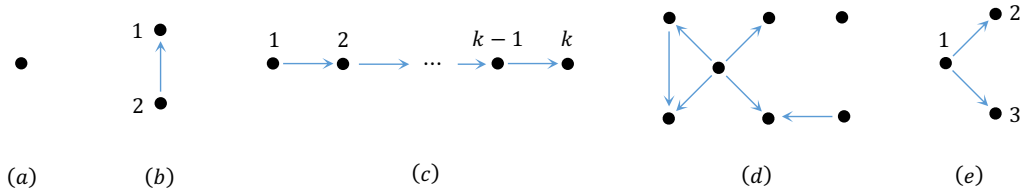

Figure 4: Examples of simple motifs. Motifs may contain no edge (a) or multiple connected components (d). The motif in (c) forms a directed path on k nodes, which we call the 'k-chain'.

For a given motif $F = ([k], A_F)$ and a n-node network $\mathcal{G} = ([n], A, \alpha)$, we introduce the following probability distribution $\pi_{F\to\mathcal{G}}$ on the set $[n]^{[k]}$ of all vertex maps $\mathbf{x} : [k] \to [n]$ by

$$\pi_{F\to\mathcal{G}}(\mathbf{x}) = \frac{1}{\mathtt{Z}} \left(\prod_{1\le i,j\le k} A(\mathbf{x}(i),\mathbf{x}(j))^{A_F(i,j)} \right) \alpha(\mathbf{x}(1)) \cdots \alpha(\mathbf{x}(k)), \tag{9}$$

where the normalizing constant $\mathtt{Z}$ is given by

$$\mathtt{Z} = \mathtt{t}(F,\mathcal{G}) := \sum_{\mathbf{x}:[k]\to[n]} \left(\prod_{1\le i,j\le k} A(\mathbf{x}(i),\mathbf{x}(j))^{A_F(i,j)} \right) \alpha(\mathbf{x}(1)) \cdots \alpha(\mathbf{x}(k)). \tag{10}$$

We call a random vertex map $\mathbf{x} : [k] \to [n]$ distributed as $\pi_{F\to\mathcal{G}}$ a *random homomorphism* from F to $\mathcal{G}$. A vertex map $\mathbf{x} : [k] \to [n]$ is a (graph) *homomorphism* $F \to \mathcal{G}$ if $\pi_{F\to\mathcal{G}}(\mathbf{x}) > 0$. Hence $\pi_{F\to\mathcal{G}}$ is a probability measure on the set of all homomorphisms $F \to \mathcal{G}$. The above quantity $\mathtt{t}(F,\mathcal{G})$ is known as the *homomorphism density* of F in $\mathcal{G}$. We now formally introduce the problem of motif sampling.

Problem 2.1 (Motif sampling from networks) *For a given motif $F = ([k], A_F)$ and an n-node network $\mathcal{G} = ([n], A, \alpha)$, sample a homomorphism $\mathbf{x} : F \to \mathcal{G}$ according to the probability distribution $\pi_{F\to\mathcal{G}}$ in (9).*

An important special case of (2.1) is when $\mathcal{G}$ is a simple graph. Let $G = ([n], A)$ be a simple graph. Then for each vertex map $\mathbf{x} : [k] \to [n]$, note that

$$\prod_{1\le i,j\le k} A(\mathbf{x}(i),\mathbf{x}(j))^{A_F(i,j)} = \mathbf{1}\left(\text{for all } (i,j) \text{ with } A_F(i,j) = 1 \text{ and } A(\mathbf{x}(i),\mathbf{x}(j)) = 1\right). \tag{11}$$

Whenever the indicator on the right-hand side above equals one, we say $\mathbf{x}$ is a homomorphism $F \to G$. That is, $\mathbf{x}$ maps an edge in F to an edge in G. Note that $\mathbf{x}$ need not be injective, so different edges in F can be mapped to the same edge in G. This leads us to the problem of motif sampling from graphs as described below.

Problem 2.2 (Motif sampling from graphs) *For a given motif $F = ([k], A_F)$ and a n-node simple graph $G = ([n], A)$, sample a homomorphism $\mathbf{x} : F \to G$ uniformly at random.*

The Problem 2.2 is indeed a special case Problem 2.1 by identifying the simple graph $G = ([n], A)$ with the network $\mathcal{G} = ([n], A, \alpha)$, where α is the uniform node weight (i.e., $\alpha(i) \equiv 1/n$ for $i = 1, \dots, n$). Then due to (11), the probability distribution $\pi_{F\to\mathcal{G}}$ in (9) becomes the uniform distribution on the set of all homomorphisms $F \to \mathcal{G}$.

2.2 Sampling subgraphs with uniform Hamiltonian path

In order to provide some concrete application contexts for the motif sampling problems posed above, here we consider the problem sampling connected subgraphs from sparse graphs. Computing a large number of k-node subgraphs from a given network is an essential task in modern network analysis, such as in computing 'network motifs' (Milo et al., 2002) and 'latent motifs' (Lyu et al., 2021, 2020; Peng et al., 2022) and in topic modeling on knowledge graphs (Alaverdian et al., 2020).

We consider the random sampling of k-node subgraphs that we obtain by uniformly randomly sampling a 'k-path' from a network and taking the induced subgraph on the sampled nodes. This subgraph sampling procedure is summarized below. (See Figure 5 for an illustration.) Here, a *k-path* is a subgraph that consists of k distinct nodes, with the ith node adjacent to the $(i+1)$th node for all $i \in \{1, \dots, k-1\}$. A path P in a graph G is a *Hamiltonian path* if P contains all nodes of G.

Sampling subgraphs via uniform Hamiltonian paths.

Given a simple graph $G = ([n], A)$ and an integer $1 \le k \le \mathrm{diam}(G)$:

(1) Sample a k-path $P \subseteq G$ uniformly at random;

(2) Return the k-node induced subgraph H of G on the nodes in P.

Above, sampling a subgraph induced by a k-path serves two purposes: (1) It ensures that the sampled k-node induced subgraph is connected with the minimum number of imposed edges; and (2) it induces a natural node ordering of the k-node induced subgraph. When applied to sparse networks, such k-node subgraphs are not likely to possess many other Hamiltonian paths, so ordering the nodes using the sampled Hamiltonian path provides a canonical representation of the subgraphs as their $k \times k$ adjacency matrices (e.g., see Figure 5 (c)). This is an important computational advantage of sampling k-node subgraphs via Hamiltonian paths over neighborhood sampling. In the latter, there is no canonical choice of node ordering out of $k!$ ways so there is a large number of equivalent adjacency matrix representations for the same subgraph.

The k-node subgraph induced on such a uniformly random k-path is guaranteed to be connected and can exhibit diverse connection patterns (see Figure 6), depending on the structure of the original network.

The key sampling problem in the above subgraph sampling scheme is to sample a k-path uniformly at random from a graph G. A naive way to do so is to use rejection sampling together with independent sampling. That is, one can repeatedly sample a set $\{x_1, \dots, x_k\}$ of k distinct nodes in G independently and uniformly at random until there is a path on the

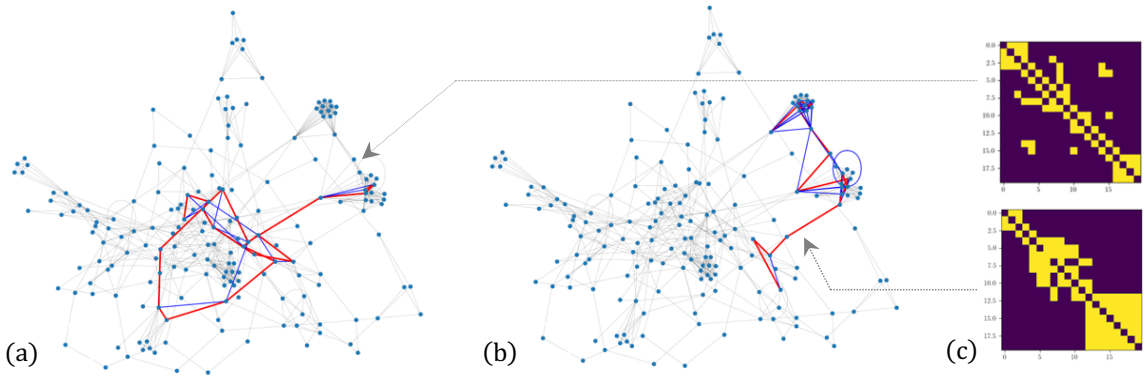

Figure 5: Illustration of motif sampling with chain motif of $k = 20$ nodes. Two instances of injective homomorphisms from a path of 20 nodes into the same network are shown in panels (a) and (b), which are depicted as paths of k nodes with red edges. Panel (c) shows the $k \times k$ adjacency matrix of the induced subgraph on these k-paths.

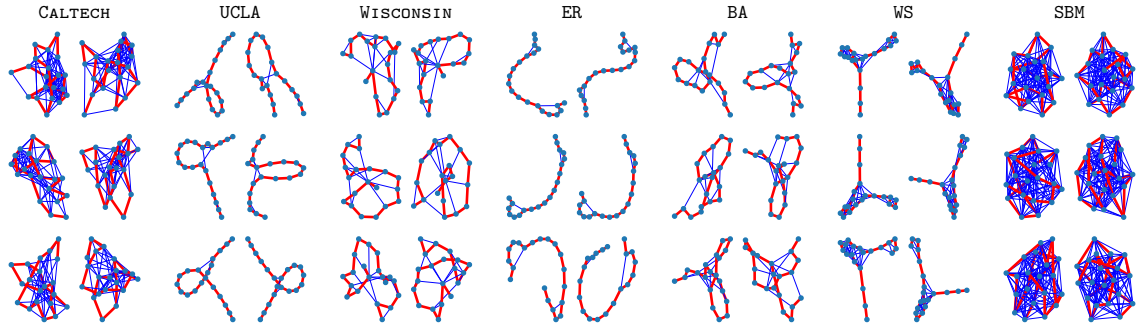

Figure 6: Examples of 20-node subgraphs induced through Hamiltonian paths on three Facebook social networks (Traud et al., 2012) and on synthetic networks generated according to the following models: the Erdős–Rényi (ER) (Erdős and Rényi, 1959), Barabási–Albert (BA) (Barabási and Albert, 1999) Watts–Strogatz (WS) (Watts and Strogatz, 1998), and stochastic-block-model (SBM) (Holland et al., 1983). For each subgraph, its Hamiltonian path (with red edges) is sampled uniformly at random by using the Glauber chain algorithm (see Def. 2.1).

sequence $(x_1, \ldots, x_k)$ (i.e., x_i and x_i are adjacent for $i = 1, \ldots, k-1$). However, when G is sparse (i.e., when the number of edges in G is much less than n^2), the probability that a randomly chosen node set $(x_1, \ldots, x_k)$ forms a k-path is extremely small, so this procedure might suffer a large number of rejections until finding a k-path.

We propose to use motif sampling with k-chain motifs to address this problem of sampling a k-path uniformly at random. Let $F = ([k], \mathbf{1}_{\{(1,2),(2,3),\ldots,(k-1,k)\}})$ denote a k-chain motif. Consider the problem sampling a homomorphism $\mathbf{x} : F \to G$ uniformly at random with the additional constraint that $\mathbf{x}$ be injective, that is, the nodes $\mathbf{x}(1), \ldots, \mathbf{x}(k)$ are distinct. When $\mathbf{x} : F \to \mathcal{G}$ is an injective homomorphism, we denote $\mathbf{x} : F \hookrightarrow \mathcal{G}$. This would give us a uniformly random k-path on the node set $\{\mathbf{x}(1), \ldots, \mathbf{x}(k)\}$. Letting $\pi_{F \hookrightarrow \mathcal{G}}$ denote the probability distribution on the set of all injective homomorphisms $F \to G$, we can write

$$\pi_{F \hookrightarrow \mathcal{G}}(\mathbf{x}) := C\, \pi_{F \to \mathcal{G}}(\mathbf{x}) \cdot \mathbf{1}(\mathbf{x}(1), \ldots, \mathbf{x}(k) \text{ are distinct}), \tag{12}$$

where $C > 0$ is a normalization constant. The probability distribution (12) is well-defined as long as there exists an injective homomorphism $\mathbf{x} : F \to \mathcal{G}$. For instance, if F is a k-chain motif for $k \geq 4$ and if $\mathcal{G}$ is a star graph, then there is no injective homomorphism $\mathbf{x} : F \to \mathcal{G}$ and the probability distribution (12) is not well-defined.

The identity (12) suggests that, if we can sample a homomorphism $\mathbf{x} : F \to \mathcal{G}$ uniformly at random efficiently, then we can sample a sequence of homomorphisms $\mathbf{x}_1, \dots, \mathbf{x}_m : F \to G$ uniformly at random until the first time m such that $\mathbf{x}_m$ is injective. Note that the probability of uniformly random homomorphism $\mathbf{x} : F \to G$ being injective is not vanishingly small even if G is sparse. In Section 2.3, we provide two MCMC sampling algorithms for sampling a homomorphism $F \to G$ uniformly at random. We remark that this sampling scheme is a crucial component in the recent development of network dictionary learning methods (Lyu et al., 2021, 2020; Peng et al., 2022).

2.3 MCMC algorithms for motif sampling

Note that computing the measure $\pi_{F\to\mathcal{G}}$ according to its definition is computationally expensive, especially when the network $\mathcal{G}$ is large. In this subsection, we give efficient randomized algorithms to sample a random homomorphism $F \to \mathcal{G}$ from the measure $\pi_{F\to\mathcal{G}}$ by a Markov chain Monte Carlo method. Namely, we seek for a Markov chain $(\mathbf{x}_t)_{t\geq 0}$ evolving in the space $[n]^{[k]}$ of vertex maps $[k] \to [n]$ such that each $\mathbf{x}_t$ is a homomorphism $F \to \mathcal{G}$ and the chain $(\mathbf{x}_t)_{t\geq 0}$ has a unique stationary distribution given by (9). We call such a Markov chain a *dynamic embedding* of F into $\mathcal{G}$. We propose two complementary dynamic embedding schemes.

Observe that equation (9) suggests considering a spin model on F where each site $i \in [k]$ takes a discrete spin $\mathbf{x}(i) \in [n]$ and the probability of such discrete spin configuration $\mathbf{x} : [k] \to [n]$ is given by (9). This spin model interpretation naturally leads us to the following dynamic embedding in terms of the Glauber chain. See Figure 7 for an illustration.

Definition 2.1 (Glauber chain) *Let $F = ([k], A_F)$ be a simple motif and $\mathcal{G} = ([n], A, \alpha)$ be a network. Suppose $\mathtt{t}(F, \mathcal{G}) > 0$ and fix a homomorphism $\mathbf{x}_0 : F \to \mathcal{G}$. Define a Markov chain $\mathbf{x}_t$ of homomorphisms $F \to \mathcal{G}$ as below.*

(i) Choose a node $i \in [k]$ of F uniformly at random.

(ii) Set $\mathbf{x}_{t+1}(j) = \mathbf{x}_t(j)$ for $j \neq i$. Update $\mathbf{x}_t(i) = a$ to $\mathbf{x}_{t+1}(i) = b$ according to the transition kernel

$$G(a,b) = \frac{\left(\prod_{j\neq i} A(\mathbf{x}_t(j), b)^{A_F(j,i)} A(b, \mathbf{x}_t(j))^{A_F(i,j)}\right) A(b,b)^{A_F(i,i)} \alpha(b)}{\sum_{1\leq c\leq n} \left(\prod_{j\neq i} A(\mathbf{x}_t(j), c)^{A_F(j,i)} A(c, \mathbf{x}_t(j))^{A_F(i,j)}\right) A(c,c)^{A_F(i,i)} \alpha(c)}, \quad (13)$$

where the product is overall $1 \leq j \leq k$ such that $j \neq i$.

Note that in the case of the Glauber chain, since all nodes in the motif try to move in all possible directions within the network, one can expect that it might take a long time to converge to its stationary distribution, $\pi_{F\to\mathcal{G}}$. To break the symmetry, we can designate a special node in the motif F as the 'pivot', and let it 'carry' the rest of the homomorphism

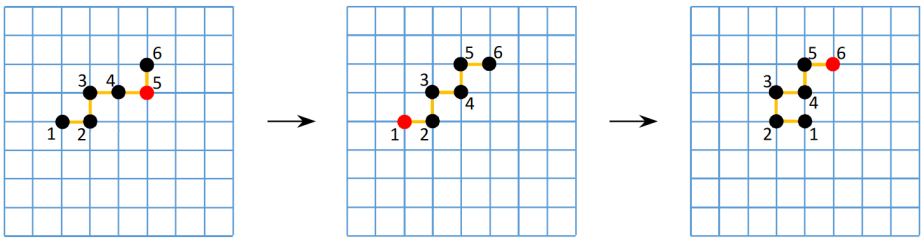

Figure 7: Glauber chain of homomorphisms $\mathbf{x}_t : F \to \mathcal{G}$, where $\mathcal{G}$ is the (9×9) grid with uniform node weights and $F = ([6], \mathbf{1}_{\{(1,2),(2,3),\cdots,(5,6)\}})$ is a 6-chain. The orientations of the edges $(1,2),\ldots,(5,6)$ are suppressed in the figure. During the first transition, node 5 is chosen with probability $1/6$ and $\mathbf{x}_t(5)$ is moved to the top left common neighbor of $\mathbf{x}_t(4)$ and $\mathbf{x}_t(6)$ with probability $1/2$. During the second transition, node 1 is chosen with probability $1/6$ and $\mathbf{x}_{t+1}(1)$ is moved to the right neighbor of $\mathbf{x}_{t+1}(2)$ with probability $1/4$.

as it performs a simple random walk on $\mathcal{G}$. A canonical random walk kernel on $\mathcal{G}$ can be modified by the Metropolis-Hastings algorithm (see, e.g., (Levin and Peres, 2017, Sec. 3.2)) so that its unique stationary distribution agrees with the correct marginal distribution from the joint distribution $\pi_{F\to\mathcal{G}}$. We can then successively sample the rest of the embedded nodes (see Figure 8) after each move of the pivot. We call this alternative dynamic embedding the *pivot chain*.

To make a precise definition of the pivot chain, we restrict the motif $F = ([k], A_F)$ to be an edge-weighted directed tree rooted at node 1 without loops. More precisely, suppose $A_F = 0$ if $k = 1$ and for $k \geq 2$, we assume that for each $2 \leq i \leq k$, $A_F(j,i) > 0$ for some unique $1 \leq j \leq k$, $j \neq i$. In this case, we denote $j = i^-$ and call it the *parent* of i. We may also assume that the other nodes in $\{2,\cdots,k\}$ are in a depth-first order, so that $i^- < i$ for all $2 \leq i \leq k$. We can always assume such ordering is given by suitably permuting the vertices, if necessary. In this case, we call F a *rooted tree motif*.

Now we introduce the pivot chain. See Figure 8 for an illustration.

Definition 2.2 (Pivot chain) *Let $F = ([k], A_F)$ be a rooted tree motif and let $\mathcal{G} = ([n], A, \alpha)$ be a network such that for each $i \in [n]$, $A(i,j) > 0$ for some $j \in [n]$. Let $\mathbf{x}_0 : [k] \to [n]$ be an arbitrary homomorphism. Define a Markov chain $\mathbf{x}_t$ of homomorphisms $F \to \mathcal{G}$ as follows.*

(i) Given $\mathbf{x}_t(1) = a$, sample a node $b \in [n]$ according to the distribution $\Psi(a,\cdot)$, where the kernel $\Psi : [n]^2 \to [0,1]$ is defined by

$$\Psi(a,b) := \frac{\alpha(a)\max(A(a,b),A(b,a))\alpha(b)}{\sum_{c\in[n]}\alpha(a)\max(A(a,c),A(c,a))\alpha(c)} \qquad a,b \in [n]. \tag{14}$$

(ii) Let $\pi^{(1)}$ denote the projection of the probability distribution $\pi_{F\to\mathcal{G}}$ (defined at (9)) onto the location of node 1. Then accept the update $a \mapsto b$ and set $\mathbf{x}_{t+1}(1) = b$ or reject the update and set $\mathbf{x}_{t+1}(1) = a$ independently with probability λ or $1-\lambda$, respectively, where

$$\lambda := \left[\frac{\pi^{(1)}(b)}{\pi^{(1)}(a)}\frac{\Psi(b,a)}{\Psi(a,b)} \wedge 1\right]. \tag{15}$$

(iii) Having sampled $\mathbf{x}_{t+1}(1), \cdots, \mathbf{x}_{t+1}(i-1) \in [n]$*, inductively, sample* $\mathbf{x}_{t+1}(i) \in [n]$ *according to the following conditional probability distribution*

$$\mathbb{P}(\mathbf{x}_{t+1}(i) = x_i \,|\, \mathbf{x}_{t+1}(1) = x_1, \cdots, \mathbf{x}_{t+1}(i-1) = x_{i-1}) \tag{16}$$

$$= \frac{\left(\prod_{2\le j<i} A(x_{j^-}, x_j)\alpha(j)\right) A(x_{i^-}, x_i)\alpha(x_i)}{\sum_{c\in[n]} \left(\prod_{2\le j<i} A(x_{j^-}, x_j)\alpha(j)\right) A(x_{i^-}, c)\alpha(c)}. \tag{17}$$

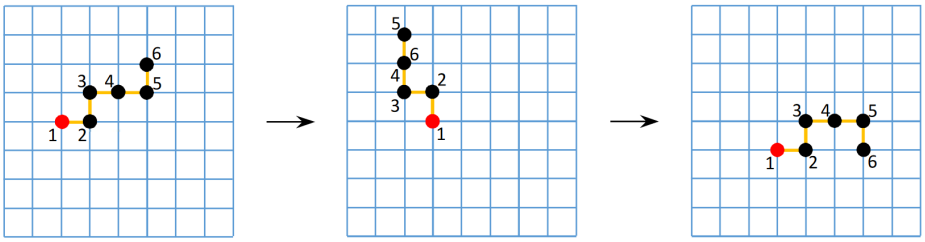

Figure 8: Pivot chain of homomorphisms $\mathbf{x}_t : F \to \mathcal{G}$, where $\mathcal{G}$ is the (9×9) grid with uniform node weight and $F = ([6], \mathbf{1}_{\{(1,2),(2,3),\cdots,(5,6)\}})$ is a 6-chain. The orientations of the edges $(1,2), \ldots, (5,6)$ are suppressed in the figure. During the first transition, the pivot $\mathbf{x}_t(1)$ moves to its right neighbor with probability $1/4$, and $\mathbf{x}_{t+1}(i)$ is sampled uniformly among the four neighbors of $\mathbf{x}_{t+1}(i-1)$ for $i = 2$ to 6. Note that $\mathbf{x}_{t+1}(4) = \mathbf{x}_{t+1}(6)$ in the middle figure. In the second transition, the pivot moves down with probability $1/4$, and again $\mathbf{x}_{t+1}(i)$ is sampled uniformly among the four neighbors of $\mathbf{x}_{t+1}(i-1)$ for $i = 2$ to 6.

The tree structure of the motif F is crucially used both in steps (ii) and (iii) of the pivot chain. Namely, computing the acceptance probability λ in step (ii) involves computing the marginal distribution $\pi^{(1)}$ on the location of the pivot from the joint distribution $\pi_{F\to\mathcal{G}}$. This can be done recursively due to the tree structure of F, admitting a particularly simple formula when F is a star or a path, see Examples 2.1 and 2.2.

In order to explain the construction of the pivot chain, we first note that the simple random walk on $\mathcal{G}$ with kernel Ψ defined at (14) has the following canonical stationary distribution

$$\pi_{\mathcal{G}}(a) := \frac{\sum_{c\in[n]} \Psi(a,c)}{\sum_{b,c\in[n]} \Psi(b,c)} \qquad a \in [n]. \tag{18}$$

When this random walk is irreducible, $\pi_{\mathcal{G}}$ is its unique stationary distribution. If we draw a random homomorphism $\mathbf{x} : F \to \mathcal{G}$ from a rooted tree motif $F = ([k], A_F)$ into a network $\mathcal{G} = ([n], A, \alpha)$ according to the distribution $\pi_{F\to\mathcal{G}}$, then for each $x_1 \in [n]$,

$$\pi^{(1)}(x_1) := \mathbb{P}_{F\to\mathcal{G}}(\mathbf{x}(1) = x_1) \tag{19}$$

$$= \frac{1}{\mathtt{t}(F,\mathcal{G})} \sum_{1\le x_2,\cdots,x_k\le n} A(x_{2^-}, x_2)\cdots A(x_{k^-}, x_k)\alpha(x_1)\alpha(x_2)\cdots\alpha(x_k). \tag{20}$$

Hence, we may use Metropolis-Hastings algorithm (Liu, 2008; Levin and Peres, 2017) to modify the random walk kernel Ψ to P so that its stationary distribution becomes $\pi^{(1)}$,

where

$$P(a,b) = \begin{cases} \Psi(a,b)\left[\frac{\pi^{(1)}(b)\Psi(b,a)}{\pi^{(1)}(a)\Psi(a,b)} \wedge 1\right] & \text{if } b \neq a \\ 1 - \sum_{c:c\neq a} \Psi(a,c)\left[\frac{\pi^{(1)}(c)\Psi(c,a)}{\pi^{(1)}(a)\Psi(a,c)} \wedge 1\right] & \text{if } b = a. \end{cases} \tag{21}$$

This new kernel P can be executed by steps (i)-(ii) in the definition of the pivot chain.

In the following examples, we consider particular instances of the pivot chain for embedding paths and stars and compute the corresponding acceptance probabilities.

Example 2.1 (Pivot chain for embedding stars) Consider the following 'star' motif $F = ([k], \mathbf{1}_{\{(1,2),(1,3),\cdots,(1,k)\}})$ centered at node 1 (e.g., (a)-(c) in Figure 4). Embedding a star into a network gives important network observables such as the transitivity ratio and average clustering coefficient (see Example 3.1). In this case, the marginal distribution $\pi^{(1)}$ of the pivot in (19) simplifies into

$$\pi^{(1)}(x_1) = \frac{\alpha(x_1)}{\mathtt{t}(F,\mathcal{G})}\left(\sum_{c\in[n]} A(x_1,c)\alpha(c)\right)^{k-1}. \tag{22}$$

Accordingly, the acceptance probability λ in (15) becomes

$$\lambda = \left[\frac{\alpha(b)\left(\sum_{c\in[n]} A(b,c)\alpha(c)\right)^{k-1}}{\alpha(a)\left(\sum_{c\in[n]} A(a,c)\alpha(c)\right)^{k-1}}\frac{\Psi(b,a)}{\Psi(a,b)} \wedge 1\right]. \tag{23}$$

For a further simplicity, suppose that the network $\mathcal{G} = ([n], A, \alpha)$ is such that A is symmetric and $\alpha \equiv 1/n$. In this case, the random walk kernel Ψ and the acceptance probability λ for the pivot chain simplify as

$$\Psi(a,b) = \frac{A(a,b)}{\sum_{c\in[n]} A(a,c)} \quad a,b \in [n], \qquad \lambda = \left[\frac{\left(\sum_{c\in[n]} A(b,c)\right)^{k-2}}{\left(\sum_{c\in[n]} A(a,c)\right)^{k-2}} \wedge 1\right]. \tag{24}$$

In particular, if $F = ([2], \mathbf{1}_{\{(0,1)\}})$, then $\lambda \equiv 1$ and the pivot $\mathbf{x}_t(1)$ performs the simple random walk on $\mathcal{G}$ given by the kernel $\Psi(a,b) \propto A(a,b)$ without rejection. ▲

Example 2.2 (Pivot chain for embedding paths) Suppose for simplicity that the node weight α on the network $\mathcal{G} = ([n], A, \alpha)$ is uniform and let $F = ([k], \mathbf{1}_{\{(1,2),(2,3),\cdots,(k-1,k)\}})$ be a k-chain motif. Draw a random homomorphism $\mathbf{x} : F \to \mathcal{G}$ from the distribution $\pi_{F\to\mathcal{G}}$. Then the marginal distribution $\pi^{(1)}$ of the pivot in (19) simplifies into

$$\pi^{(1)}(x_1) = \frac{n^{-k}}{\mathtt{t}(F,\mathcal{G})}\sum_{c\in[n]} A^{k-1}(x_1,c). \tag{25}$$

Hence the acceptance probability in step (ii) of the pivot chain becomes

$$\lambda = \left[\frac{\sum_{c\in[n]} A^{k-1}(b,c)}{\sum_{c\in[n]} A^{k-1}(a,c)}\frac{\Psi(b,a)}{\Psi(a,b)} \wedge 1\right], \tag{26}$$

which involves computing powers of the matrix A up to the length of the path F. ▲

Remark 2.1 (Comparison between the Glauber and the pivot chains) Here we compare various aspects of the Glauber and the pivot chains.

(Per-iteration complexity) The Glauber chain is much cheaper than the pivot chain per iteration for bounded degree networks. Note that in each step of the Glauber chain, the transition kernel in (13) can be computed in at most $O(\Delta(\mathcal{G})k^2)$ steps in general, where $\Delta(\mathcal{G})$ denotes the 'maximum degree' of $\mathcal{G}$, which we understand as the maximum degree of the edge-weighted graph $([n], A)$ as defined at (7).

For the pivot chain, from the computations in Examples 2.1 and 2.2, one can easily generalize the formula for the acceptance probability λ recursively when F is a general directed tree motif. This will involve computing powers of A up to the depth of the tree. More precisely, the computational cost of each step of the pivot chain is of order $\Delta(\mathcal{G})^{\ell\Delta(F)}$, where $\Delta(\mathcal{G})$ and $\Delta(F)$ denote the maximum degree of $\mathcal{G}$ and F (defined at (7)) and ℓ denotes the depth of F. Unlike the Glauber chain, this could be exponentially large in the depth of F even when $\mathcal{G}$ and F have bounded maximum degrees.

(Iteration complexity (or mixing time)) The pivot chain requires much less iterations to mix to the stationary distribution than the Glauber chain for sparse networks. In Theorem 2.5, we show that the mixing time of the pivot chain is about the same as the standard random walk on networks. In Theorem 2.4, we show that the Glauber chain mixes fast for dense networks. However, if $\mathcal{G}$ is sparse, we do not have a good mixing bound and we expect the chain may mix slowly.

(Sampling k-chain motifs from sparse networks) For the problem of sampling k-chain motifs from sparse networks, we recommend using the pivot chain but with an approximate computation of the acceptance probability. For instance, taking only a bounded number of powers of the weight matrix A in (26) seems to work well in practice.

(Sampling motifs from dense networks) For sampling general motifs (not necessarily trees) from dense networks, we recommend to use the Glauber chain.

2.4 Convergence and mixing of Glauber/pivot chains

In this subsection, we state convergence results for the Glauber and pivot chains.

We say a network $\mathcal{G} = ([n], A, \alpha)$ is *irreducible* if the random walk on $\mathcal{G}$ with kernel Ψ defined at (14) visits all nodes in $\mathcal{G}$ with positive probability. Note that since $\Psi(a, b) > 0$ if and only if $\Psi(b, a) > 0$, each proposed move $a \mapsto b$ is never rejected with probability 1. Hence $\mathcal{G}$ is irreducible if and only if the random walk on $\mathcal{G}$ with the modified kernel P is irreducible. Moreover, we say $\mathcal{G}$ is *bidirectional* if $A(i, j) > 0$ if and only if $A(j, i) > 0$ for all $i, j \in [n]$. Lastly, we associate a simple graph $G = ([n], A_G)$ with the network $\mathcal{G}$, where A_G is its adjacency matrix given by $A_G(i, j) = \mathbf{1}(\min(A(i, j), A(j, i)) > 0)$. We call G the *skeleton* of $\mathcal{G}$.

Theorem 2.1 (Convergence of Glauber chain) *Let $F = ([k], A_F)$ be a motif and $\mathcal{G} = ([n], A, \alpha)$ be an irreducible network. Suppose $\mathtt{t}(F, \mathcal{G}) > 0$ and let $(\mathbf{x}_t)_{t\geq 0}$ be the Glauber chain $F \to \mathcal{G}$.*

(i) *$\pi_{F\to\mathcal{G}}$ is a stationary distribution for the Glauber chain.*

(ii) *Suppose F is a rooted tree motif, $\mathcal{G}$ is bidirectional. Then the Glauber chain is irreducible if and only if $\mathcal{G}$ is not bipartite. If $\mathcal{G}$ is not bipartite, then $\pi_{F\to\mathcal{G}}$ is the unique stationary distribution for the Glauber chain.*

The proof of Theorem 2.1 (i) uses a straightforward computation. For (ii), since F is a rooted tree, one can argue that for the irreducibility of the Glauber chain for homomorphisms $F \to \mathcal{G}$, it suffices to check irreducibility of the Glauber chain $\mathbf{x}_t : K_2 \to \mathcal{G}$, where K_2 is the 2-chain motif, which has at most two communicating classes depending on the 'orientation' of x_t. Recall that $\mathcal{G}$ is not bipartite if and only if its skeleton G contains an odd cycle. An odd cycle in G can be used to construct a path between arbitrary two homomorphisms $K_2 \to \mathcal{G}$. See Appendix B for more details.

We also have the corresponding convergence results for the pivot chain in Theorem 2.2 below.

Theorem 2.2 (Convergence of pivot chain) *Let $\mathcal{G} = ([n], A, \alpha)$ be an irreducible network with $A(i,j) > 0$ for some $j \in [n]$ for each $i \in [n]$. $F = ([k], A_F)$ be a rooted tree motif. Then pivot chain $F \to \mathcal{G}$ is irreducible with unique stationary distribution $\pi_{F\to\mathcal{G}}$.*

Since both the Glauber and pivot chains evolve in the finite state space $[n]^{[k]}$, when given the irreducibility condition, both chains converge to their unique stationary distribution $\pi_{F\to\mathcal{G}}$. Then the Markov chain ergodic theorem implies the following corollary.

Theorem 2.3 (Computing stationary mean by ergodic mean) *Let $F = ([k], A_F)$ be a rooted tree motif and $\mathcal{G} = ([n], A, \alpha)$ be an irreducible network. Let $g : [n]^{[k]} \to \mathbb{R}^d$ be any function for $d \geq 1$. Let $\mathbf{x} : [k] \to [n]$ denote a random homomorphism $F \to \mathcal{G}$ drawn from $\pi_{F\to\mathcal{G}}$.*

(i) *If $(\mathbf{x}_t)_{t\geq 0}$ denotes the pivot chain $F \to \mathcal{G}$, then*

$$\mathbb{E}[g(\mathbf{x})] = \lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} g(\mathbf{x}_t). \tag{27}$$

(ii) *If $\mathcal{G}$ is bidirectional and its skeleton is not bipartite, then (27) also holds for the Glauber chain $(\mathbf{x}_t)_{t\geq 0} : F \to \mathcal{G}$.*

Next, we address the question of how long we should run the Markov chain Monte Carlo in order to get a precise convergence to the target measure $\pi_{F\to\mathcal{G}}$. Recall that the *total deviation distance* between two probability distributions μ, ν on a finite set Ω is defined by

$$\|\mu - \nu\|_{\mathrm{TV}} := \frac{1}{2} \sum_{x\in\Omega} |\mu(x) - \nu(x)|. \tag{28}$$

If $(X_t)_{t\geq 0}$ is any Markov chain on finite state space Ω with transition kernel P and unique starionay distribution π, then its *mixing time* t_{mix} is defined to be the function

$$t_{mix}(\varepsilon) = \inf \left\{ t \geq 0 \,:\, \max_{x\in\Omega} \|P^t(x,\cdot) - \pi\|_{\mathrm{TV}} \leq \varepsilon \right\}. \tag{29}$$

In Theorems 2.4 and 2.5 below, we give bounds on the mixing times of the Glauber and pivot chains when the underlying motif F is a tree. For the Glauber chain, let $\mathbf{x} : F \to \mathcal{G}$ be a homomorphism and fix a node $j \in [k]$. Define a probability distribution $\mu_{\mathbf{x},j}$ on $[n]$ by

$$\mu_{\mathbf{x},i}(b) = \frac{\left(\prod_{j\neq i} A(\mathbf{x}(j),b)^{A_F(j,i)} A(b,\mathbf{x}(j))^{A_F(i,j)}\right) A(b,b)^{A_F(i,i)}\alpha(b)}{\sum_{1\le c\le n}\left(\prod_{j\neq i} A(\mathbf{x}(j),c)^{A_F(j,i)} A(c,\mathbf{x}(j))^{A_F(i,j)}\right) A(c,c)^{A_F(i,i)}\alpha(c)}, \tag{30}$$

This is the conditional distribution that the Glauber chain uses to update $\mathbf{x}(j)$.

For each integer $d \ge 1$ and network $\mathcal{G} = ([n], A, \alpha)$, define the following quantity

$$c(d,\mathcal{G}) := \max_{\substack{\mathbf{x},\mathbf{x}':S_d\to\mathcal{G}\\ \mathbf{x}\sim\mathbf{x}' \text{ and } \mathbf{x}(1)=\mathbf{x}'(1)}} \left(1 - 2d\|\mu_{\mathbf{x},1} - \mu_{\mathbf{x}',1}\|_{\mathrm{TV}}\right), \tag{31}$$

where $S_d = ([d+1], E)$ is the star with d leaves where node 1 is at the center, and $\mathbf{x} \sim \mathbf{x}'$ means that they differ by at most one coordinate. For a motif $F = ([k], A_F)$, we also recall its *maximum degree* $\Delta(F)$ defined in (7).

Theorem 2.4 (Mixing time of Glauber chain) *Suppose $F = ([k], A_F)$ is a rooted tree motif and $\mathcal{G}$ is an irreducible and bidirectional network. Further, assume that the skeleton G of $\mathcal{G}$ contains an odd cycle. If $c(\Delta(F),\mathcal{G}) > 0$, then the mixing time $t_{mix}(\varepsilon)$ of the Glauber chain $(\mathbf{x}_t)_{t\ge0}$ of homomorphisms $F \to \mathcal{G}$ satisfies*

$$t_{mix}(\varepsilon) \le \lceil 2c(\Delta(F),\mathcal{G})k\log(2k/\varepsilon)(\mathrm{diam}(G)+1)\rceil. \tag{32}$$

On the other hand, we show that the pivot chain mixes at the same time that the single-site random walk on network $\mathcal{G}$ does. An important implication of this fact is that the mixing time of the pivot chain does not depend on the size of the motif. However, the computational cost of performing each step of the pivot chain does increase in the size of the motif (see Remark 2.1).

It is well-known that the mixing time of a random walk on $\mathcal{G}$ can be bounded by the absolute spectral gap of the transition kernel in (21) (see (Levin and Peres, 2017, Thm. 12.3, 12.4)). Moreover, a standard coupling argument shows that the mixing time is bounded above by the meeting time of two independent copies of the random walk. Using a well-known cubic bound on the meeting times due to Coppersmith et al. (1993), we obtain the following result.

Theorem 2.5 (Mixing time of pivot chain) *Let $F = ([k], E_F)$ be a directed rooted tree and $\mathcal{G} = ([n], A, \alpha)$ be an irreducible network. Further assume that for each $i \in [n]$, $A(i,j) > 0$ for some $j \in [n]$. Let P denote the transition kernel of the random walk on $\mathcal{G}$ defined at (21). Then the mixing time $t_{mix}(\varepsilon)$ of the pivot chain $(\mathbf{x}_t)_{t\ge0}$ of homomorphisms $F \to \mathcal{G}$ satisfies the following.*

(i) *Let $t^{(1)}_{mix}(\varepsilon)$ be the mixing time of the pivot with kernel P. Then*

$$t_{mix}(\varepsilon) = t^{(1)}_{mix}(\varepsilon). \tag{33}$$

(ii) *Let $\lambda_\star$ be the eigenvalue of P with largest modulus that is less than 1. Then*

$$\frac{\lambda_\star \log(1/2\varepsilon)}{1-\lambda_\star} \le t_{mix}(\varepsilon) \le \max_{x\in[n]} \frac{\log(1/\alpha(x)\varepsilon)}{1-\lambda_\star}. \tag{34}$$

(iii) *Suppose $n \ge 13$, A is the adjacency matrix of some simple graph, and $\alpha(i) \propto \deg(i)$ for each $i \in [n]$. Then*

$$t_{mix}(\varepsilon) \le \log_2(\varepsilon^{-1}) \left(\frac{4}{27}n^3 + \frac{4}{3}n^2 + \frac{2}{9}n - \frac{296}{27} \right). \tag{35}$$

2.5 Concentration and statistical inference

Suppose $(\mathbf{x}_t)_{t\ge 0}$ is the pivot chain of homomorphisms $F \to \mathcal{G}$, and let $g : [k]^{[n]} \to \mathbb{R}^d$ be a function for some $d \ge 1$. In the previous subsection, we observed that various observables on the network $\mathcal{G}$ can be realized as the expected value $\mathbb{E}[g(\mathbf{x})]$ under the stationary distribution $\pi_{F\to\mathcal{G}}$, so according to Corollary 3.1, we can approximate them by time averages of increments $g(\mathbf{x}_t)$ for a suitable choice of g. A natural question to follow is that if we take the time average for the first N steps, is it possible to infer the stationary expectation $\mathbb{E}[g(\mathbf{x})]$?

The above question can be addressed by applying McDiarmid's inequality for Markov chains (see, e.g., (Paulin et al., 2015, Cor. 2.11)) together with the upper bound on the mixing time of pivot chain provided in Theorems 2.5.

Theorem 2.6 (Concentration bound for real-valued observables) *Let $F = ([k], E_F)$, $\mathcal{G} = ([n], A, \alpha)$, $(\mathbf{x}_t)_{t\ge 0}$, and $t^{(1)}_{mix}(\varepsilon)$ be as in Theorem 2.5. Let $g : [k]^{[n]} \to \mathbb{R}$ be any functional. Then for any $\delta > 0$,*

$$\mathbb{P}\left(\left| \mathbb{E}_{\pi_{F\to\mathcal{G}}}[g(\mathbf{x})] - \frac{1}{N}\sum_{t=1}^{N} g(\mathbf{x}_t) \right| \ge \delta \right) < 2\exp\left(\frac{-2\delta^2 N}{9 t^{(1)}_{mix}(1/4)} \right). \tag{36}$$

A similar result for the Glauber chain (with $t^{(1)}_{mix}(1/4)$ at (36) replaced by $t_{mix}(1/4)$) can be derived from the mixing bounds provided in Theorem 2.4.

Remark 2.2 *One can reduce the requirement for running time N in Theorem 2.6 by a constant factor in two different ways. First, if the random walk of pivot on $\mathcal{G}$ exhibits a cutoff, then the factor of 9 in (36) can be replaced by 4 (see (Paulin et al., 2015, Rmk. 2.12)). Second, if we take the partial sum of $g(\mathbf{x}_t)$ after a 'burn-in period' a multiple of mixing time of the pivot chain, then thereafter we only need to run the chain for a multiple of the relaxation time $1/(1-\lambda_\star)$ of the random walk of pivot (see (Levin and Peres, 2017, Thm. 12.19)).*

Next, we give a concentration inequality for the vector-valued partial sums process. This will allow us to construct confidence intervals for CHD profiles and motif transforms. The key ingredients are the use of burn-in period as in (Levin and Peres, 2017, Thm. 12.19) and a concentration inequality for vector-valued martingales (Hayes, 2005).

Theorem 2.7 (Concentration bound for vector-valued observables) *Suppose* $F = ([k], A_F)$, $\mathcal{G} = ([n], A, \alpha)$, $(\mathbf{x}_t)_{t\geq 0}$, *and* $t^{(1)}_{mix}(\varepsilon)$ *be as in Theorem 2.5. Let* $\mathcal{H}$ *be any Hilbert space and let* $g : [n]^{[k]} \to \mathcal{H}$ *be any function such that* $\|g\|_\infty \leq 1$. *Then for any* $\varepsilon, \delta > 0$,

$$\mathbb{P}\left(\left\|\mathbb{E}_{\pi_{F\to\mathcal{G}}}[g(\mathbf{x})] - \frac{1}{N}\sum_{t=1}^{N} g(\mathbf{x}_{r+t})\right\| \geq \delta\right) \leq 2\exp\left(2 - \frac{-\delta^2 N}{2}\right) + \varepsilon, \tag{37}$$

provided $r \geq t^{(1)}_{mix}(\varepsilon)$.

3. Network observables based on motif sampling

In Section 2, we introduced the motif sampling problem and proposed MCMC algorithms for the efficient computational solution of this problem. In that section we also established various theoretical guarantees. Specifically, we have shown that the stationary expectation of an arbitrary vector-valued function of a random homomorphism can be computed through an ergodic average along MCMC trajectories (see Theorems 2.6 and 2.7). In this section, we introduce specific network observables that can be efficiently computed in this way and also establish their stability properties.

3.1 Definitions and computation

In this section, we introduce four network observables based on the random embedding of motif F into a network $\mathcal{G}$. The first one is a conditional version of the well-known homomorphism density Lovász (2012).

Definition 3.1 (Conditional homomorphism density) *Let* $\mathcal{G} = ([n], A, \alpha)$ *be a network and fix two motifs* $H = ([k], A_H)$ *and* $F = ([k], A_F)$. *Let* $H + F$ *denote the motif* $([k], A_H + A_F)$. *We define the conditional homomorphism density (CHD) of* H *in* $\mathcal{G}$ *given* F *by*

$$\mathtt{t}(H, \mathcal{G}|F) = \frac{\mathtt{t}(H + F, \mathcal{G})}{\mathtt{t}(F, \mathcal{G})}, \tag{38}$$

which is set to zero when the denominator is zero.

When $\mathcal{G}$ is a simple graph with uniform node weight, the above quantity equals the probability that all edges in H are preserved by a uniform random homomorphism $\mathbf{x} : F \to \mathcal{G}$. As a notable special case, we describe a quantity closely related to the the average clustering coefficient as a conditional homomorphism density.

Example 3.1 (Average clustering coefficient) A notable special case is when F is the wedge motif $W_3 = ([3], \mathbf{1}_{\{(1,2),(1,3)\}})$ (see Figure 4 (e)) and $H = ([3], \mathbf{1}_{\{(2,3)\}})$ and $\mathcal{G}$ is a simple graph. Then $\mathtt{t}(H, \mathcal{G} \,|\, W_3)$ is the conditional probability that a random sample of three nodes x_1, x_2, x_3 in $\mathcal{G}$ induces a copy of the triangle motif K_3, given that there are edges from x_1 to each of x_2 and x_3 in $\mathcal{G}$. If all three nodes are required to be distinct, such a conditional probability is known as the *transitivity ratio* (Luce and Perry, 1949).

A similar quantity with different averaging leads to the *average clustering coefficient*, which was introduced to measure how a given network locally resembles a complete graph

and used to define small-world networks by Watts and Strogatz (1998). Namely, we may write

$$\mathtt{t}(H,\mathcal{G}\,|\,W_3)=\sum_{x_1\in[n]}\frac{\sum_{x_2,x_3\in[n]}A(x_1,x_2)A(x_2,x_3)A(x_1,x_3)\alpha(x_2)\alpha(x_3)}{\left(\sum_{x_2\in[n]}A(x_1,x_2)\alpha(x_2)\right)^2}\frac{\alpha(x_1)}{\sum_{x_1\in[n]}\alpha(x_1)}. \tag{39}$$

If $\mathcal{G}$ is a simple graph with uniform node weight $\alpha\equiv 1/n$, then we can rewrite the above equation as

$$\mathtt{t}(H,\mathcal{G}\,|\,W_3)=\sum_{x_1\in[n]}\frac{\#(\text{edges between neighbors of } x_1 \text{ in } \mathcal{G})}{\deg_{\mathcal{G}}(x_1)(\deg_{\mathcal{G}}(x_1)-1)/2}\frac{\deg_{\mathcal{G}}(x_1)-1}{n\deg_{\mathcal{G}}(x_1)}. \tag{40}$$

If the second ratio in the above summation is replaced by $1/n$, then it becomes the average clustering coefficient of $\mathcal{G}$ (Watts and Strogatz, 1998). Hence the conditional homomorphism density $\mathtt{t}(H,\mathcal{G}\,|\,W_3)$ can be regarded as a a variant of the generalized average clustering coefficient, which lower bounds the average clustering coefficient of $\mathcal{G}$ when it is a simple graph. We also remark that a direct generalization of the average clustering coefficient in terms of higher-order cliques was introduced recently by Yin et al. (2018). See also Cozzo et al. (2015) for a related discussion for multiplex networks. ▲

Motivated by the connection between the conditional homomorphism density and the average clustering coefficient discussed above, we introduce the following generalization of the average clustering coefficient. (See Figures 2 and 15 for examples.)

Definition 3.2 (Matrix of Average Clustering Coefficients) *Let $\mathcal{G}=([n],A,\alpha)$ be a network and fix a motif $F=([k],A_F)$. For each $1\le i\le j\le k$, let $H_{ij}=([k],A_F+\mathbf{1}_{\{(i,j)\}}\mathbf{1}(A_F(i,j)=0))$ be the motif obtained by 'adding' the edge (i,j) to F. We define the Matrix of Average Clustering Coefficient (MACC) of $\mathcal{G}$ given F by the $k\times k$ matrix whose (i,j) coordinate is given by*

$$\mathtt{MACC}(\mathcal{G}|F)(i,j)=\frac{\mathtt{t}(H_{ij},\mathcal{G})}{\mathtt{t}(F,\mathcal{G})}, \tag{41}$$

which is set to zero when the denominator is zero.

Next, instead of looking at the conditional homomorphism density of H in $\mathcal{G}$ given F at a single scale, we could look at how the conditional density varies at different scales as we threshold $\mathcal{G}$ according to a parameter $t\ge 0$. Namely, we draw a random homomorphism $\mathbf{x}:F\to\mathcal{G}$, and ask if all the edges in H have weights $\ge t$ in $\mathcal{G}$. This naturally leads to the following function-valued observable.

Definition 3.3 (CHD profile) *Let $\mathcal{G}=([n],A,\alpha)$ be a network and fix two motifs $H=([k],A_H)$ and $F=([k],A_F)$. We define the CHD (Conditional Homomorphism Density) profile of a network $\mathcal{G}$ for H given F by the function $\mathtt{f}(H,\mathcal{G}\,|\,F):[0,1]\to[0,1]$,*

$$\mathtt{f}(H,\mathcal{G}\,|\,F)(t)=\mathbb{P}_{F\to\mathcal{G}}\left(\min_{1\le i,j\le k}A(\mathbf{x}(i),\mathbf{x}(j))^{A_H(i,j)}\ge t\right), \tag{42}$$

where $\mathbf{x}:F\to\mathcal{G}$ is a random embedding drawn from the distribution $\pi_{F\to\mathcal{G}}$ defined at (9).

We give examples of CHD profiles involving two-armed paths, singleton, and self-loop motifs.

Example 3.2 (CHD profiles involving two-armed paths) For integers $k_1, k_2 \geq 0$, we define a *two-armed path motif* $F_{k_1,k_2} = (\{0, 1, \cdots, k_1+k_2\}, \mathbf{1}(E))$ where its set E of directed edges are given by

$$E = \left\{ \begin{matrix} (0,1), (1,2), \cdots, (k_1-1, k_1), \\ (0, k_1+1), (k_1+1, k_1+2), \cdots, (k_1+k_2-1, k_1+k_2) \end{matrix} \right\}. \tag{43}$$

This is also the rooted tree consisting of two directed paths of lengths k_1 and k_2 from the root 0. Also, we denote $H_{k_1,k_2} = (\{0, 1, \cdots, k_1+k_2\}, \mathbf{1}_{\{(k_1,k_1+k_2)\}})$. This is the motif on the same node set as F_{k_1,k_2} with a single directed edge between the ends of the two arms. (See Figure 9.)

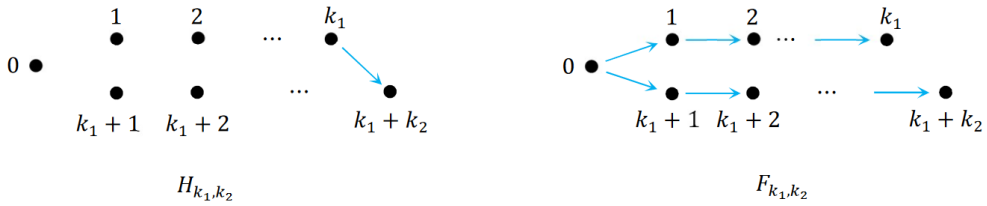

Figure 9: Plots of the motif H_{k_1,k_2}, which contains a single directed edge from k_1 to k_1+k_2 (left), and the two-armed path motif F_{k_1,k_2} on the right.

When $k_1 = k_2 = 0$, then $F_{0,0}$ and $H_{0,0}$ become the 'singleton motif' $([0], \mathbf{0})$ and the 'self-loop motif' $([0], \mathbf{1}_{(0,0)})$, respectively. In this case the corresponding homomorphism and conditional homomorphism densities have simple expressions involving only the diagonal entries of the edge weight matrix of the network. Namely, for a given network $\mathcal{G} = ([n], A, \alpha)$, we have

$$\mathtt{t}(H_{0,0}, \mathcal{G}) = \sum_{k=1}^{n} A(k,k)\alpha(k), \qquad \mathtt{t}(F_{0,0}, \mathcal{G}) = \sum_{k=1}^{n} \alpha(k) = 1. \tag{44}$$

The former is also the weighted average of the diagonal entries of A with respect to the node weight α. For the conditional homomorphism densities, observe that

$$\mathtt{t}(H_{0,0}, \mathcal{G} \,|\, F_{0,0}) = \sum_{k=1}^{n} A(k,k)\alpha(k), \qquad \mathtt{t}(H_{0,0}, \mathcal{G} \,|\, H_{0,0}) = \frac{\sum_{k=1}^{n} A(k,k)^2\alpha(k)}{\sum_{k=1}^{n} A(k,k)\alpha(k)}. \tag{45}$$

The latter is also the ratio between the first two moments of the diagonal entries of A. The corresponding CHD profile is given by

$$\mathtt{f}(H_{0,0}, \mathcal{G} \,|\, F_{0,0})(t) = \sum_{k=1}^{n} \mathbf{1}(A(k,k) \geq t)\alpha(k), \tag{46}$$

$$\mathtt{f}(H_{0,0}, \mathcal{G} \,|\, H_{0,0})(t) = \frac{\sum_{k=1}^{n} \mathbf{1}(A(k,k) \geq t)A(k,k)\alpha(k)}{\sum_{k=1}^{n} A(k,k)\alpha(k)}. \tag{47}$$

The above two quantities can be interpreted as the probability that the self-loop intensity $A(k,k)$ is at least t, when $k \in [n]$ is chosen with probability proportional to $\alpha(k)$ or $A(k,k)\alpha(k)$, respectively. ▲

Lastly, we define network-valued observables from motif sampling. Recall that motif sampling gives the k-dimensional probability measure $\pi_{F\to\mathcal{G}}$ on the set $[n]^{[k]}$. Projecting this measure onto the first and last coordinates gives a probability measure on $[n]^{\{1,k\}}$. This can be regarded as the weight matrix $A^F : [n]^2 \to [0,1]$ of another network $\mathcal{G}^F := ([n], A^F, \alpha)$. The precise definition is given below.

Definition 3.4 (Motif transform) *Let $F = ([k], A_F)$ be a motif for some $k \geq 2$ and $\mathcal{G} = ([n], A, \alpha)$ be a network. The motif transform of $\mathcal{G}$ by F is the network $\mathcal{G}^F := ([n], A^F, \alpha)$, where*

$$A^F(x,y) = \mathbb{P}_{F\to\mathcal{G}}\left(\mathbf{x}(1) = x,\, \mathbf{x}(k) = y\right), \tag{48}$$

where $\mathbf{x} : F \to \mathcal{G}$ is a random embedding drawn from the distribution $\pi_{F\to\mathcal{G}}$ defined at (9).

Motif transforms can be used to modify a given network so that certain structural defects are remedied without perturbing the original network aggressively. For instance, suppose $\mathcal{G}$ consists two large cliques C_1 and C_2 connected by a thin path P. When we perform the single-linkage clustering on $\mathcal{G}$, it will perceive $C_1 \cup P \cup C_2$ as a single cluster, even though the linkage P is not significant. To overcome such an issue, we could instead perform single-linkage clustering on the motif transform $\mathcal{G}^F$ where F is a triangle. Then the thin linkage P is suppressed by the transform, and the two cliques C_1 and C_2 will be detected as separate clusters. See Example 5.4 for more details.

Remark 3.1 Transformations of networks analogous to motif transforms have been studied in the context of clustering of metric spaces and networks by Carlsson and Mémoli (2013); Carlsson et al. (2017, 2016) and Mémoli and Pinto (2020).

Next, we discuss how to compute the network observables we introduced in this section. Given a motif F and network $\mathcal{G}$, the CHD, CHD profile, and motif transform are all defined by the expectation of a suitable function of a random homomorphism $\mathbf{x} : F \to \mathcal{G}$. While computing this expectation directly is computationally challenging, we can efficiently compute them by taking time averages along MCMC trajectory $\mathbf{x}_t : F \to \mathcal{G}$ of a dynamic embedding (see Theorems 2.6, 2.7, and 2.6). This is more precisely stated in the following corollary.

Corollary 3.1 *Let $F = ([k], A_F)$ be a rooted tree motif, $H = ([k], A_H)$ another motif, and $\mathcal{G} = ([n], A, \alpha)$ an irreducible network. Let $(\mathbf{x}_t)_{t\geq 0}$ be the pivot chain $F \to \mathcal{G}$. Then the*

followings hold:

$$\mathtt{t}(H,\mathcal{G}|F) = \lim_{N\to\infty} \frac{1}{N}\sum_{t=1}^{N} \prod_{1\le i,j\le k} A(\mathbf{x}_t(i),\mathbf{x}_t(j))^{A_H(i,j)}, \tag{49}$$

$$\mathtt{f}(H,\mathcal{G}\,|\,F)(t) = \lim_{N\to\infty} \frac{1}{N}\sum_{t=1}^{N} \prod_{1\le i,j\le k} \mathbf{1}\left(A(\mathbf{x}_t(i),\mathbf{x}_t(j))^{A_H(i,j)} \ge t\right) \quad t\in[0,1], \tag{50}$$

$$\mathtt{t}(H,\mathcal{G}\,|,F)A^H = \lim_{N\to\infty} \frac{1}{N}\sum_{t=1}^{N} \left(\prod_{1\le i,j\le k} A(\mathbf{x}_t(i),\mathbf{x}_t(j))^{A_H(i,j)}\right) E_{\mathbf{x}_t(1),\mathbf{x}_t(k)}, \tag{51}$$

where $E_{i,j}$ denotes the $(n\times n)$ matrix with zero entries except 1 at (i,j) entry. Furthermore, $\mathcal{G}$ is bidirectional and its skeleton contains an odd cycle, then the above equations also hold for the Glauber chain $(\mathbf{x}_t)_{t\ge 0} : F \to \mathcal{G}$.

Remark 3.2 When we compute A^H using (51), we do not need to approximate the conditional homomorphism density $\mathtt{t}(H,\mathcal{G},|\,F)$ separately. Instead, we compute the limiting matrix on the right-hand side of (51) and normalize by its 1-norm so that $\|A^H\|_1 = 1$.

Remark 3.3 We emphasize that all the network observables that we introduced in this section can be expressed as the expected value of some function of a random homomorphism $F\to\mathcal{G}$, and that any network observable defined in this manner can be computed efficiently by taking suitable time averages along MCMC trajectory of homomorphisms $\mathbf{x}_t : F\to\mathcal{G}$ as in Corollary 3.1. It would be interesting to investigate other network observables that can be expressed as the expectation of some function of a random homomorphism $F\to\mathcal{G}$.

4. Stability inequalities

In this section, we establish stability properties of the network observables we introduced in Section 3.1. Roughly speaking, our aim is to show that these observable change little when we change the underlying network little. In order to do so, we need to introduce a notion of distance between networks.

We introduce two commonly used notions of distance between networks as viewed as 'graphons'. A *kernel* is a measurable integrable function $W : [0,1]^2 \to [0,\infty)$. We say a kernel W is a *graphon* if it takes values from $[0,1]$. Note that we do not require the kernels and graphons to be symmetric, in contrast to the convention use in Lovász (2012). For a given network $\mathcal{G} = ([n], A, \alpha)$, we define a 'block kernel' $U_{\mathcal{G}} : [0,1]^2 \to [0,1]$ by

$$U_{\mathcal{G}}(x,y) = \sum_{1\le i,j\le n} A(i,j)\mathbf{1}(x\in I_i, y\in I_j), \tag{52}$$

where $[0,1] = I_1 \sqcup I_2 \sqcup \cdots \sqcup I_n$ is a partition such that each I_i is an interval with Lebesgue measure $\mu(I_i) = \alpha(i)$. (For more discussion on kernels and graphons, see (Lovász, 2012).)

For any integrable function $W : [0,1]^2 \to \mathbb{R}$, we define its p-norm by

$$\|W\|_p = \left(\int_0^1 \int_0^1 |W(x,y)|^p \, dx \, dy \right)^{1/p}, \tag{53}$$

for any real $p \in (0, \infty)$, and its *cut norm* by

$$\|W\|_\square = \sup_{A,B \subseteq [0,1]} \left| \int_A \int_B W(x,y) \, dx \, dy \right|, \tag{54}$$

where the supremum is taken over Borel-measurable subsets of $A, B \subseteq [0,1]$. Now for any two networks $\mathcal{G}_1$ and $\mathcal{G}_2$, we define their *p-distance* by

$$\delta_p(\mathcal{G}_1, \mathcal{G}_2) = \inf_\varphi \|U_{\mathcal{G}_1} - U_{\varphi(\mathcal{G}_2)}\|_p, \tag{55}$$

where the infimum is taken over all bijections $\varphi : [n] \to [n]$ and $\varphi(\mathcal{G}_2)$ is the network $([n], A^\varphi, \alpha \circ \varphi)$, $A^\varphi(x,y) = A(\varphi(x), \varphi(y))$. Taking infimum over φ ensures that the similarity between two networks does not depend on relabeling of nodes. We define *cut distance* between $\mathcal{G}_1$ and $\mathcal{G}_2$ similarly and denote it by $\delta_\square(\mathcal{G}_1, \mathcal{G}_2)$. We emphasize that the cut norm and cut distance are well-defined for possibly asymmetric kernels.

The cut distance is more conservative than the 1-norm in the sense that

$$\delta_\square(W_1, W_2) \le \delta_1(W_1, W_2) \tag{56}$$

for any two kernels W_1 and W_2. This follows from the fact that

$$\|W\|_\square \le \| \, |W| \, \|_\square = \|W\|_1. \tag{57}$$

for any kernel W.

Now we state stability inequalities for the network observables we introduced in Section 3.1 in terms of kernels and graphons. The homomorphism density of a motif $F = ([k], A_F)$ in a kernel U is defined by (see, e.g., Lovász and Szegedy (2006, Subsection 7.2))

$$\mathtt{t}(F, U) = \int_{[0,1]^k} \prod_{1 \le i,j \le k} U(x_i, x_j)^{A_F(i,j)} \, dx_1 \cdots dx_k. \tag{58}$$

For any other motif $H = ([k], A_H)$, we define the conditional homomorphism density of H in U given F by $\mathtt{t}(H, U \,|\, F) = \mathtt{t}(H + F, U)/\mathtt{t}(F, U)$, where $F + E = ([k], A_E + A_F)$ and we set $\mathtt{t}(H, U \,|\, F) = 0$ if $\mathtt{t}(F, U) = 0$. It is easy to check that the two definitions of conditional homomorphism density for networks and graphons agree, namely $\mathtt{t}(H, \mathcal{G} \,|\, F) = \mathtt{t}(H, U_\mathcal{G} \,|\, F)$. Also, CHD for kernels is defined analogously to (42). That is, we define the *CHD profile* of a kernel $U : [0,1]^2 \to [0,1]$ for H given F by the function $\mathtt{f}(H, U \,|\, F) : [0,1] \to [0,1]$,

$$\mathtt{f}(H, U \,|\, F)(t) = \int_{[0,1]^k} \mathbf{1} \left(\min_{1 \le i,j \le k} U(\mathbf{x}(i), \mathbf{x}(j))^{A_H(i,j)} \ge t \right) dx_1, \ldots, dx_k. \tag{59}$$

Finally, we define the motif transform $U^F : [0,1]^2 \to [0,\infty)$ of a kernel U by a motif $F = ([k], A_F)$ for $k \geq 2$ by

$$U^F(x_1, x_k) = \frac{1}{\mathtt{t}(F,U)} \int_{[0,1]^{k-2}} \prod_{1 \leq i,j \leq k} U(x_i, x_j)^{A_F(i,j)} \, dx_2 \cdots dx_{k-1}. \tag{60}$$

The well-known stability inequality for homomorphism densities is due to Lovász and Szegedy (2006), which reads

$$|\mathtt{t}(F,U) - \mathtt{t}(F,W)| \leq |E_F| \cdot \delta_\square(U,W) \tag{61}$$

for any two graphons $U, W : [0,1]^2 \to [0,1]$ and a motif $F = ([k], E_F)$. A simple application of this inequality shows that conditional homomorphism densities are also stable with respect to the cut distance up to normalization.

Proposition 4.1 *Let $H = ([k], A_H)$ and $F = ([k], A_F)$ be motifs such that $H + F = ([k], A_H + A_F)$ is simple. Let $U, V : [0,1]^2 \to [0,1]$ be graphons. Then*

$$|\mathtt{t}(H, U|F) - \mathtt{t}(H, W|F)| \leq \frac{2|E_H| \cdot \delta_\square(U,W)}{\max(\mathtt{t}(F,U), \mathtt{t}(F,W))}. \tag{62}$$

As a corollary, this also yields a similar stability inequality for the MACC (see Definition 3.2). A similar argument shows that motif transforms are also stable with respect to cut distance.

Proposition 4.2 *Let $F = ([k], A_F)$ be a simple motif and let $U, W : [0,1]^2 \to [0,1]$ be graphons. Then*

$$\delta(U^F, W^F)_\square \leq \left(1 + \frac{1}{\max(\mathtt{t}(F,U), \mathtt{t}(F,W))}\right) |E(F)| \cdot \delta_\square(U,W) \tag{63}$$

Lastly, we state a stability inequality for the CHD profiles in Theorem 4.1. While the proof of Propositions 4.1 and 4.2 is relatively straightforward using the stability inequality for the homomorphism density (61), the proof of Theorem 4.1 is more involved and requires new analytical tools. The main idea is to define a notion of cut distance between 'filtrations of graphons' and to show that this new distance interpolates between the cut distance and the 1-norm-distance (see Proposition C.1). See Appendix C for more details.

Theorem 4.1 *Let $H = ([k], A_H)$ and $F = ([k], A_F)$ be simple motifs such that $H + F = ([k], A_H + A_F)$ is simple. Then for any graphons $U, W : [0,1]^2 \to [0,1]$,*

$$\|\mathtt{f}(H, U \,|\, F) - \mathtt{f}(H, W \,|\, F)\|_1 \leq \frac{2\|A_F\|_1 \cdot \delta_\square(U,W) + \|A_H\|_1 \cdot \delta_1(U,W)}{\max(\mathtt{t}(F,U), \mathtt{t}(F,W))}. \tag{64}$$

5. Examples

In this section, we provide various computational examples to demonstrate our techniques and results. Throughout this section we use the motifs H_{k_1,k_2} and F_{k_1,k_2} introduced in Example 3.2. In Subsection 5.1, we compute explicitly and numerically various homomorphism densities for the network given by a torus graph plus some random edges. In Subsection 5.2, we compute various CHD profiles for stochastic block networks. Lastly, in Subsection 5.3, we discuss motif transforms in the context of hierarchical clustering of networks and illustrate this using a barbell network.

5.1 Conditional homomorphism densities

Example 5.1 (Torus) Let $\mathcal{G}_n = ([n] \times [n], A, \alpha)$ be the $(n \times n)$ torus $\mathbb{Z}_n \times \mathbb{Z}_n$ with nearest neighbor edges and uniform node weight $\alpha \equiv 1/n^2$. Consider the conditional homomorphism density $\mathtt{t}(H_{k,0}, \mathcal{G}_n \,|\, F_{k,0})$. Since A binary and symmetric, note that $\mathbb{P}_{F_{k,0} \to \mathcal{G}_n}$ is the uniform probability distribution on the sample paths of simple symmetric random walk on $\mathcal{G}_n$ for the first k steps. Hence if we denote this random walk by $(X_t)_{t \geq 0}$, then

$$\mathtt{t}(H_{k,0}, \mathcal{G}_n \,|\, F_{k,0}) = \mathbb{P}(\|X_k - (0,0)\|_\infty = 1 \,|\, X_0 = (0,0)) \tag{65}$$

$$= 4\mathbb{P}(X_{k+1} = (0,0) \,|\, X_0 = (0,0)) \tag{66}$$

$$= \frac{1}{4^k} \sum_{\substack{a,b \geq 0 \\ 2(a+b)=k+1}} \frac{(k+1)!}{a!a!b!b!}. \tag{67}$$

For instance, we have $\mathtt{t}(H_{3,0}, \mathcal{G}_n \,|\, F_{3,0}) = 9/16 = 0.5625$ and

$$\mathtt{t}(H_{9,0}, \mathcal{G}_n \,|\, F_{9,0}) = \frac{2 \cdot 10!}{4^9} \left(\frac{1}{5!5!} + \frac{1}{4!4!} + \frac{1}{3!3!2!2!} \right) = \frac{3969}{16384} \approx 0.2422. \tag{68}$$

See Figure 25 for a simulation of Glauber and Pivot chains $F_{k,0} \to \mathcal{G}_n$. As asserted in Corollary 3.1, time averages of these dynamic embeddings converge to the correct values of the conditional homomorphism density $\mathtt{t}(H_{k,0}, \mathcal{G}_n \,|\, F_{k,0})$. The simulation indicates that for sparse networks like the torus, the Glauber chain takes longer to converge than Pivot chain does. ▲

Example 5.2 (Torus with long-range edges) Fix parameters $p \in [0,1]$ and $\alpha \in [0, \infty)$. Let $\mathcal{G}_n = \mathcal{G}_n^{p,\alpha}$ be the $n \times n$ torus $\mathbb{Z}_n \times \mathbb{Z}_n$ with additional edges added randomly to each non-adjacent pair (a,b) and (c,d), independently with probability $p(|a-c| + |b-d|)^{-\alpha}$. When $\alpha = 0$, this reduces to the standard Watts-Strogatz model Watts and Strogatz (1998).

See Figure 25 for some simulation of Glauber and Pivot chains $F_{k,0} \to \mathcal{G}_{50}$ for $p = 0.1$ and $\alpha = 0$. Time averages of these dynamic embeddings converge to the correct values of the conditional homomorphism density $\mathtt{t}(H_{k,0}, \mathcal{G}_n \,|\, F_{k,0})$, which is approximately the ambient edge density 0.1. This is because if we sample a copy of $F_{k,0}$, it is likely to use some ambient 'shortcut' edges so that the two ends of $F_{k,0}$ are far apart in the usual shortest path metric on the torus. Hence the chance that these two endpoints are adjacent in the network $\mathcal{G}_n^{p,0}$ is roughly p.

In the next example, we use the tree motif F on six nodes and H is obtained from F by adding two extra edges, as described in Figure 29. A similar reasoning to the one used above tells us that the probability that a random copy of F from $\mathcal{G}_n^{p,0}$ has edges $(2,5)$ and $(3,6)$ should be about p^2. Indeed, both the Glauber and Pivot chains in Figure 29 converge to 0.01. ▲

5.2 CHD profiles of stochastic block networks

Let $\mathcal{G} = ([n], A, \alpha)$ be a network. For each integer $r \geq 1$ and a real number $\sigma > 0$, we will define a 'stochastic block network' $\mathfrak{X} = ([nr], B^{(r)}(A, \sigma^2), \beta)$ by replacing each node of $\mathcal{G}$ by a community with r nodes. The node weight $\beta : [nr] \to [0,1]$ of the block network

is inherited from α by the relation $\beta(x) = \alpha(\lfloor x/r \rfloor + 1)$. For the edge weight, we define $B^{(r)}(A, \sigma^2) = \Gamma^{(r)}(A, \sigma^2)/\max(\Gamma^{(r)}(A, \sigma^2))$, where $\Gamma^{(r)}(A, \sigma^2)$ is the $(nr \times nr)$ random matrix obtained from A by replacing each of its positive entries $a_{ij} > 0$ by an $(r \times r)$ matrix of i.i.d. entries following a Gamma distribution with mean a_{ij} and variance σ^2. Recall that the Gamma distribution with parameters α and β has the following probability distribution function

$$f_{\alpha,\beta}(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x} \mathbf{1}(x \geq 0). \tag{69}$$

Since the mean and variance of the above distribution are given by α/β and α/β^2, respectively, we may set $\alpha = a_{ij}^2/\sigma^2$ and $\beta = a_{ij}/\sigma^2$ for the $(r \times r)$ block corresponding to a_{ij}.

For instance, consider two networks $\mathcal{G}_1 = ([6], A_1, \alpha)$, $\mathcal{G}_2 = ([6], A_2, \alpha)$ where $\alpha \equiv 1/6$ and

$$A_1 = \begin{bmatrix} 5 & 1 & 1 & 1 & 1 & 1 \\ 1 & 5 & 1 & 1 & 1 & 1 \\ 1 & 1 & 5 & 1 & 1 & 1 \\ 1 & 1 & 1 & 5 & 1 & 1 \\ 1 & 1 & 1 & 1 & 5 & 1 \\ 1 & 1 & 1 & 1 & 1 & 5 \end{bmatrix}, \qquad A_2 = \begin{bmatrix} 1 & 1 & 1 & 5 & 5 & 1 \\ 1 & 1 & 1 & 1 & 1 & 5 \\ 5 & 1 & 1 & 5 & 1 & 5 \\ 5 & 1 & 1 & 1 & 1 & 2 \\ 1 & 5 & 1 & 1 & 1 & 1 \\ 1 & 1 & 5 & 10 & 1 & 1 \end{bmatrix}. \tag{70}$$

Let $B_1 = B^{(10)}(A_1, 1)$, $B_2 = B^{(10)}(A_2, 1.5)$, and $B_3 = B^{(10)}(A_2, 0.5)$. Consider the stochastic block networks $\mathfrak{X}_1 = ([60], B_1, \beta), \mathfrak{X}_2 = ([60], B_2, \beta)$, and $\mathfrak{X}_3 = ([60], B_3, \beta)$. The plots of matrices B_1 and B_2 are given in Figure 27.

In Figure 10 below, we plot the CHD profiles $\mathtt{f} := \mathtt{f}(H_{k_1,k_2}, \mathfrak{X} \,|\, F_{k_1,k_2})$ for $\mathfrak{X} = \mathfrak{X}_1, \mathfrak{X}_2$, and $\mathfrak{X}_3$. The first row in Figure 10 shows the CHD profiles for $k_1 = k_2 = 0$. At each filtration level $t \in [0, 1]$, the value $\mathtt{f}(t)$ of the profile, in this case, means the proportion of diagonal entries in B_i at least t (see Example 3.2). The CHD profiles for $\mathfrak{X}_2$ and $\mathfrak{X}_3$ drop quickly to zero by level $t = 0.3$, as opposed to the profile for $\mathfrak{X}_1$, which stays close to height 1 and starts dropping around level $t = 0.4$. This is because, as can be seen in Figure 27, entries in the diagonal blocks of the matrix B_1 are large compared to that in the off-diagonal blocks, whereas for the other two matrices B_1 and B_2, diagonal entries are essentially in the order of the Gamma noise with standard deviation σ.

For $\max(k_1, k_2) \geq 1$, note that the value of the profile $\mathtt{f}(t)$ at level t equals the probability that the extra edge in H_{k_1,k_2} has weight $\geq t$ in $\mathfrak{X}$, when we sample a random copy of F_{k_1,k_2} from $\mathfrak{X}$. For instance, if $(k_1, k_2) = (0, 1)$, this quantity is almost the density of edges in $\mathfrak{X}$ whose weights are at least t. But since the measure of random homomorphism $\mathbf{x} : F_{1,0} \to \mathfrak{X}$ is proportional to the edge weight $B_i(\mathbf{x}(0), \mathbf{x}(1))$, we are in favor of sampling copies of $F_{0,1}$ with large edge weight.

In the second row of Figure 10, the profile for $\mathfrak{X}_3$ differs drastically from the other two, which gradually decays to zero. The small variance in the Gamma noise for sampling B_3 makes the two values of 5 and 10 in A_2 more pronounced with respect to the 'ground level' 1. Hence we see two plateaus in its profile. As noted in the previous paragraph, the height of the first plateau (about 0.7), is much larger than the actual density (about 0.25) of the edges sample from blocks of value 5. A similar tendency could be seen in the third row of

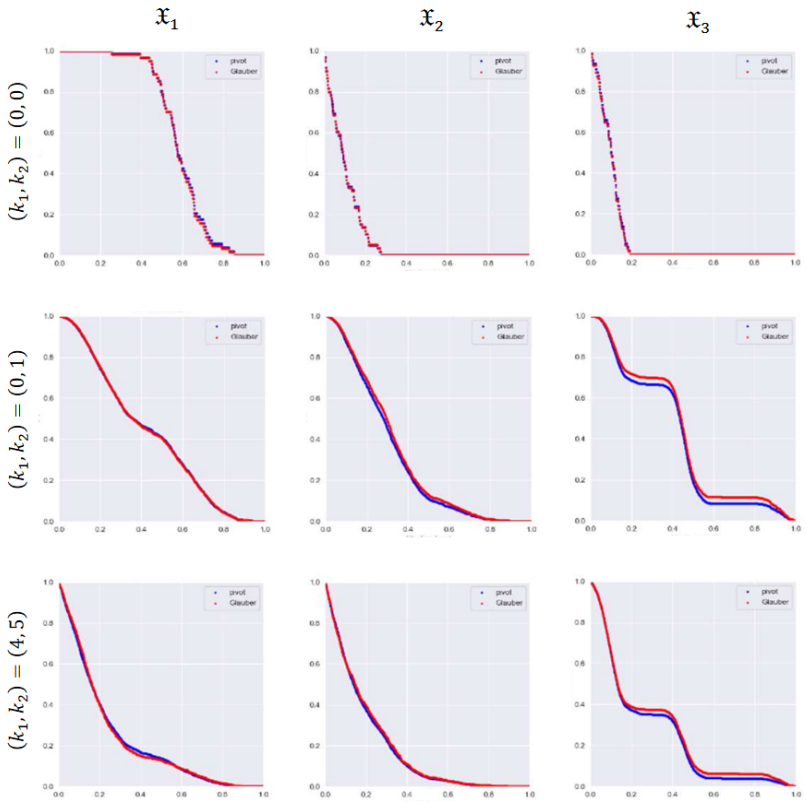

Figure 10: Plots of CHD profiles $\mathtt{f}(H_{k_1,k_2}, \mathfrak{X} \,|\, F_{k_1,k_2})$ for $\mathfrak{X} = \mathfrak{X}_1$ (first row), $\mathfrak{X}_2$ (second row), and $\mathfrak{X}_3$ (third row). To compute each profile, both the Glauber (red) and pivot (blue) chains are run up to 10^5 iterations.

Figure 10, which shows the CHD profiles for $(k_1, k_2) = (4, 5)$. Note that the first plateau in the profile for $\mathfrak{X}$ now appears at a lower height (about 0.4). This indicates that sampling a copy of $F_{4,5}$ is less affected by the edge weights than sampling a copy of $F_{0,1}$.

5.3 Hierarchical clustering for networks and motif transform

In this section, we discuss the application of the motif transform to the setting of hierarchical clustering of networks.

Hierarchical clustering and dendrogram. Suppose we are given a finite set X and fixed 'levels' $0 = t_0 < t_1 < \cdots < t_m$ for some integer $m \geq 0$. Let $\mathcal{H} := (\mathcal{F}_t)_{t \in \{t_0, \ldots, t_m\}}$ be a sequence of collection of subsets of X, that is, $\mathcal{F}_{t_k} \subseteq 2^X$ with 2^X denoting the power set of X. We call $\mathcal{H}$ a *hierarchical clustering* of X if (1) $\mathcal{F}_0 = X$ and $\mathcal{F}_{t_m} = \{X\}$ and (2) for each $0 \leq a \leq b \leq m$ and for each $A \in \mathcal{F}_{t_a}$, there exists a unique $B \in \mathcal{F}_{t_b}$ with $A \subseteq B$. For each $t \in \{t_0, \ldots, t_M\}$, we call each $A \in \mathcal{F}_t$ a *cluster* of X at *level* t. One can associate a tree $T = (V, E)$ to $\mathcal{H}$ by setting $V = \bigsqcup_{k=0}^{m} \mathcal{F}_{t_k}$ and letting the edge set E consist of all pairs (A, B) such that $A \in \mathcal{F}_{t_k}$, $B \in \mathcal{F}_{t_{k+1}}$, and $A \subseteq B$ for some $k = 0, 1, \ldots, m-1$. The graph $T = (V, E)$ defined in this way is indeed a tree with root $\{X\}$ at level t_m and a set of leaves

X at level 0. We call T a *dendrogram* of $\mathcal{H}$. See (Jardine and Sibson, 1971; Carlsson et al., 2010) for more details on hierarchical clustering and dendrograms.

Single-linkage hierarchical clustering for finite metric spaces. *Single-linkage hierarchical clustering* is a standard mechanism to obtaining a hierarchical clustering of a finite matrix space. Suppose (X, d) is a finite metric space. Let $0 = t_0 < t_1 < \cdots < t_m$ be the result of ordering all distinct values of the pairwise distances $d(x, y)$ for $x, y \in X$. For each $t \geq 0$, define the equivalence relation $\simeq_t$ on X as the *transitive closure* of the relation $d(x, y) \leq t$, that is,

$$x \simeq_t y \iff \begin{array}{c} \text{exists an integer } m \geq 1 \text{ and } z_0, \ldots, z_m \in X \text{ s.t.} \\ d(z_i, z_{i+1}) \leq t \text{ for } i = 0, \ldots, m-1 \text{ and } z_0 = x,\ z_m = y. \end{array} \tag{71}$$

Then $\mathcal{H} := (U_{t_k})_{0 \leq k \leq m}$ is a hierarchical clustering of X. The associated dendrogram T of $\mathcal{H}$ is called the *single-linkage (hierarchical clustering) dendrogram* of (X, d).

Single-linkage hierarchical clustering for networks. We introduce a method for computing the hierarchical clustering of networks based on a metric on the node set. Let $\mathcal{G} = ([n], A, \alpha)$ be a network. We view the weight $A(x, y)$ between distinct nodes as representing the magnitude of the relationship between them, so the larger $A(x, y)$ is, the stronger the nodes x, y are associated. Hence it would be natural to interpret the off-diagonal entries of A as a measure of similarity between the nodes, as opposed to a metric d on a finite metric space, which represents 'dissimilarity' between points.

In order to define a metric d_A on the node set $[n]$, first transform the pairwise similarity matrix A into a pairwise *dissimilarity* matrix A' as

$$A'(x, y) = \begin{cases} 0 & \text{if } x = y \\ \infty & \text{if } A(x, y) = 0 \text{ and } x \neq y \\ \max(A) - A(x, y) & \text{otherwise.} \end{cases} \tag{72}$$

We then define a metric d_A on the node set $[n]$ by letting $d_A(x, y)$ be the smallest sum of all A'-edge weights of any sequence of nodes starting from x and ending at y:

$$d_A(x, y) := \inf \left\{ \sum_{i=1}^{m} A'(x_i, x_{i+1}) \,\middle|\, x_1 = x,\ x_{m+1} = y,\ x_1, \ldots, x_{m+1} \in [n] \right\}. \tag{73}$$

This defines a metric space $([n], d_A)$ associated with the network $\mathcal{G} = ([n], A, \alpha)$. We let $\mathcal{H} = \mathcal{H}(\mathcal{G})$ to denote the hierarchical clustering of $([n], d_A)$. We call the dendrogram $T = (V, E)$ of $\mathcal{H}(\mathcal{G})$ the *single-linkage heirarchical clustering dendrogram* of the network $\mathcal{G}$, or simply the *dendrogram* of $\mathcal{G}$. Computing the metric d_A in (73) can be easily accomplished in $O(n^3)$ time by using the Floyd-Warshall algorithm (Floyd, 1962; Warshall, 1962). See Figures 13, 14, and 33 for network dendrograms computed in this way.

The hierarchical clustering $\mathcal{H}$ of $\mathcal{G}$ defined above is closely related to the following notion of network capacity function. Given a network $\mathcal{G} = ([n], A, \alpha)$, consider the 'capacity function' $T_{\mathcal{G}} : [n]^2 \to [0, \infty)$ defined by

$$T_{\mathcal{G}}(x, y) = \sup_{t \geq 0} \left\{ t \geq 0 \,\middle|\, \begin{array}{c} \exists x_0, x_1, \cdots, x_m \in [n] \text{ s.t. } (x_0, x_m) = (x, y) \text{ or } (y, x) \\ \text{and } \min_{0 \leq i < m} A(x_i, x_{i+1}) > t. \end{array} \right\}. \tag{74}$$

That is, $T_{\mathcal{G}}(x, y)$ is the minimum edge weight of all possible walks from x to y in $\mathcal{G}$, where a *walk* in $\mathcal{G}$ is a sequence of nodes $x_0, \ldots, x_m$ such that $A(x_i, x_{i+1}) > 0$ for $i = 1, \ldots, m-1$. Let $\simeq_t$ denote the equivalence relation induced by d_A as in (71). Then one can see that

$$x \simeq_t y \iff T_{\mathcal{G}}(x, y) \geq \max(A) - t \quad \text{or} \quad x = y. \tag{75}$$

Thus, x and y merge into the same cluster in $\mathcal{H}$ at level $\max(A) - T_{\mathcal{G}}(x, y)$. Furthermore, it is easy to see that $T_{\mathcal{G}}$ satisfies the following 'dual' to ultrametric condition (see Carlsson and Mémoli (2010) for how the ultrametric condition relates to dendrograms) for distinct nodes:

$$T_{\mathcal{G}}(x, y) \geq \min(T_{\mathcal{G}}(x, z), T_{\mathcal{G}}(z, y)) \qquad \forall\ x, y, z \in [n] \text{ s.t. } x \neq y. \tag{76}$$

Note that $T_{\mathcal{G}}(x, x) = A(x, x)$ for all $x \in [n]$. Hence (76) may not hold if $x = y$, as $T_{\mathcal{G}}(x, y) = A(x, x)$ could be less than the minimum of $T_{\mathcal{G}}(x, z)$ and $T_{\mathcal{G}}(z, x))$ (e.g., $\mathcal{G}$ a simple graph). If we modify the capacity function on the diagonal by setting $T_{\mathcal{G}}(x, x) \equiv \max(A)$ for all x, then (76) is satisfied for all choices $x, y, z \in [z]$. This modification corresponds to setting $A'(x, x) = 0$ in (72).

The above construction of the hierarchical clustering $\mathcal{H}$ of $\mathcal{G} = ([n], A, \alpha)$ does not use diagonal entries of A. One can slightly modify the definition of hierarchical clustering of $\mathcal{G}$ in a way that it also uses the diagonal entries of A by allowing each node x to 'appear' in the dendrogram at different times depending on its 'self-similarity' $A(x, x)$. More precisely, define a relation $\simeq'_t$ on the node set $[n]$ by $x \simeq'_t y$ if and only if $T_{\mathcal{G}}(x, y) \geq \max(A) - t$ for all $x, y \in [n]$ (not necessarily for distinct x, y). Then $x \simeq'_t x$ if and only if $t \geq \max(A) - A(x, x)$. Hence in order for the relation $\simeq'_t$ to be an equivalence relation, we need to restrict its domain to $\{x \in [n] \,|\, \max(A) - A(x, x) \leq t\}$ at each filtration level t. The resulting dendrogram is called a *treegram*, since its leaves may appear at different heights (Smith et al., 2016).

Note that the capacity function in (74) can be defined for graphons U instead of networks. Hence by using (75), we can also define hierarchical clustering dendrogram for graphons in a similar manner. The following example illustrates single-linkage hierarchical clustering of the three-block graphons from Example A.4.

Example 5.3 Recall the graphons U, $U^{\circ 2}$, and $U \cdot U^{\circ 2}$ discussed in Example A.4. Note that U is the graphon $U_{\mathcal{G}}$ associated to the network $\mathcal{G} = ([3], A, \alpha)$ in Example A.1. Single-linkage hierarchical clustering dendrograms of the three networks corresponding to the three graphons are shown in Figure 11 (in solid + dotted blue lines), which are solely determined by the off-diagonal entries. Truncating each vertical line below the corresponding diagonal entry (dotted blue lines), one obtains the treegrams for the three networks (solid blue lines).

Furthermore, one can also think of hierarchical clustering of the graphons by viewing them as networks with continuum node set $[0, 1]$. The resulting dendrogram is shown in Figure 11 (solid blue lines + shaded rectangles) ▲.

In the following example, we illustrate how motif transforms can be used to suppress weak connections between two communities in order to improve the recovery of the hierarchical clustering structure of a given network.

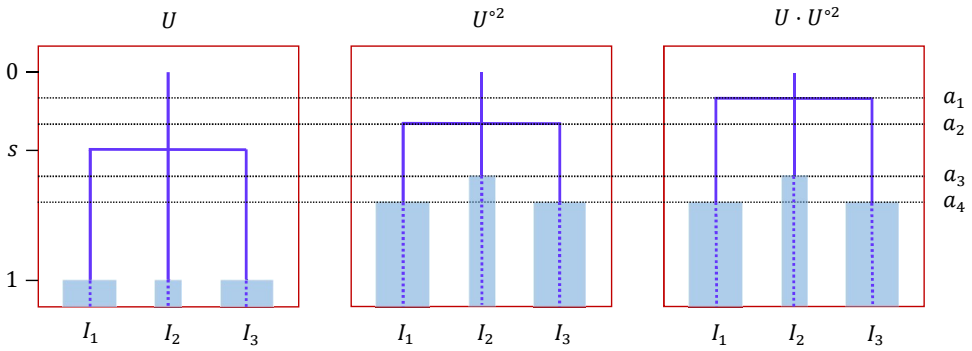

Figure 11: Dendrograms and treegrams of the networks associated to the graphons $U = U_{\mathcal{G}}$ (left), $U^{\circ 2}$ (middle), and $U \cdot U^{\circ 2}$ (right) in Example A.4. The vertical axis show the values of the capacity function from (74). $a_1 = s^2(1+\epsilon)/2$, $a_2 = s(1+\epsilon)/2$, $a_3 = s^2(1-\epsilon/4)+\epsilon$, and $a_4 = (1/2)+(s^2-1/2)\epsilon$. See the text in Example 5.3 for more details.

Example 5.4 (Barbell networks) In this example, we consider 'barbell networks', which are obtained by connecting two networks by a single edge of weight 1. When the two networks are $\mathcal{H}_1$ and $\mathcal{H}_2$, we denote the resulting barbell network by $\mathcal{H}_1 \oplus \mathcal{H}_2$, and we say $\mathcal{H}_1$ and $\mathcal{H}_2$ are the two components of $\mathcal{H}_1 \oplus \mathcal{H}_2$.

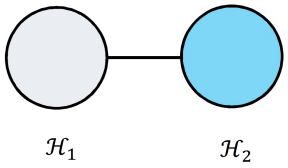

Figure 12: Depiction of a barbell network.

Recall the network $\mathcal{G}_n^{p,\alpha}$ defined in Example 5.2, which is the $(n \times n)$ torus with long-range edges added according to the parameters p and α. Also let $\mathfrak{X} = ([nr], B_r(A, \sigma^2), \beta)$ denote the stochastic block network constructed from a given network $\mathcal{G} = ([n], A, \alpha)$ (see Subsection 5.2). Denote the stochastic block network corresponding to $\mathcal{G}_n^{p,\alpha}$ with parameters r and σ by $\mathcal{G}_n^{p,\alpha}(r, \sigma)$.

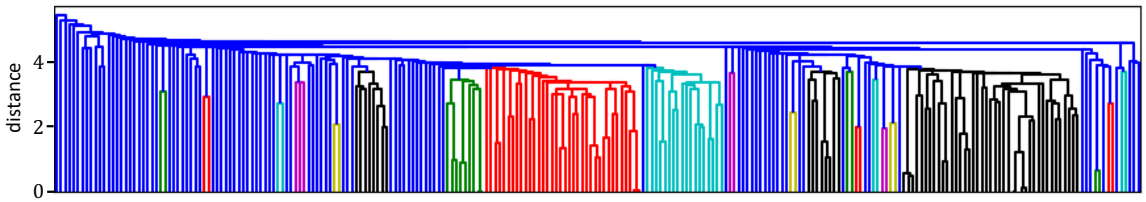

Figure 13: Single-linkage dendrogram for the barbell network $\mathcal{G}_2$.

Now define barbell networks $\mathcal{G}_1 := \mathcal{G}_{10}^{0,0} \oplus \mathcal{G}_{10}^{0.2,0}$ and $\mathcal{G}_2 = \mathcal{G}_5^{0,0}(5, 0.6) \oplus \mathcal{G}_5^{0.2,0}(5, 0.2)$. Also, let $\mathcal{G}_3 := \mathcal{G}_2^{C_3}$ be the network obtained from $\mathcal{G}_2$ by the motif transform using the triangle motif $C_3 := ([3], \mathbf{1}_{\{(1,2),(2,3),(3,1)\}})$ (here the orientation of the edges of C_3 is irrelevant since the networks are symmetric). In each barbell network, the two components are connected

by the edge between node 80 and node 53 in the two components. For each $i \in \{1, 2, 3\}$, let A_i denote the edge weight matrix corresponding to $\mathcal{G}_i$. The plots for A_i's are given in Figure 28.

We are going to consider the single-linkage dendrograms of each barbell network for their hierarchical clustering analysis. We omit the dendrogram of the simple graph $\mathcal{G}_1$. For $\mathcal{G}_2$, the Gamma noise prevents all nodes from merging at the same level. Instead, we expect to have multiple clusters forming at different levels and they all merge into one cluster at some positive level $t > 0$. Indeed, in the single-linkage dendrogram for $\mathcal{G}_2$ shown in Figure 13, we do observe such hierarchical clustering structure of $\mathcal{G}_2$.

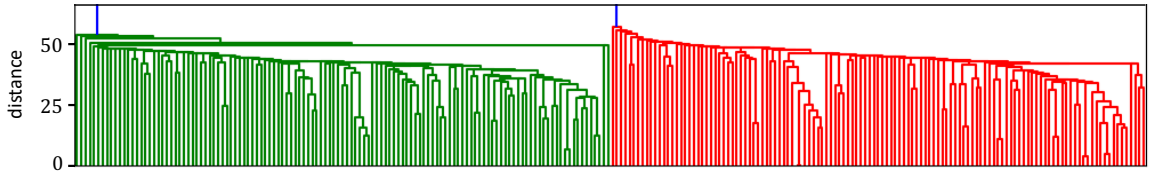

Figure 14: Single-linkage dendrogram for barbell network $\mathcal{G}_3$, which is obtained by motif-transforming $\mathcal{G}_2$ using a triangle C_3.

However, the 'single linkage' between the two main components of $\mathcal{G}_2$ is very marginal compared to the substantial interconnection within the components. We may use motif transforms prior to single-linkage clustering in order to better separate the two main components. The construction of $\mathcal{G}_3 = \mathcal{G}_2^{C_3}$ using triangle motif transform and its dendrogram in Figure 14 demonstrate this point.

In the dendrogram of $\mathcal{G}_3$ shown in Figure 14, we see that the two main clusters still maintain internal hierarchical structure, but they are separated at all levels $t \geq 0$. A similar motif transform may be used to suppress weak connections in the more general situation in order to emphasize the clustering structure within networks, but without perturbing the given network too much. ▲

6. Application I: Subgraph classification and Network clustering with Facebook networks

In this section, we apply our methods to a problem consisting of clustering given a data set of networks. In our experiment, we use the *Facebook100 dataset*, which consists of the snapshots of the Facebook network of 100 schools in the US in Sep. of 2005. Since it was first published and analyzed by Traud, Mucha, and Porter in Traud et al. (2012), it has been regarded as a standard data set in the field of social network analysis. In the dataset, each school's social network is encoded as a simple graph of anonymous nodes corresponding to the users, and nodes i and j are adjacent if and only if the corresponding users have a friendship relationship on the Facebook network. The networks have varying sizes: Caltech36 is the smallest with 769 nodes and 16,656 edges, whereas Texas84 is the largest with 36,371 nodes and 1,590,655 edges. The lack of shared node labels and varying network sizes make it difficult to directly compare the networks and perform clustering tasks. For instance, for directly computing a distance between two networks of different sizes without a shared node labels, one needs to find optimal correspondence between the node sets (as in (55)), which

is computationally very expensive. We overcome this difficulty by using our motif sampling for computing the Matrix of Average Clustering Coefficients (MACC) (see Definition 3.2) for each network. This the compressed representation of social networks will then be used for performing hierarchical clustering on the whole collection of 100 networks.

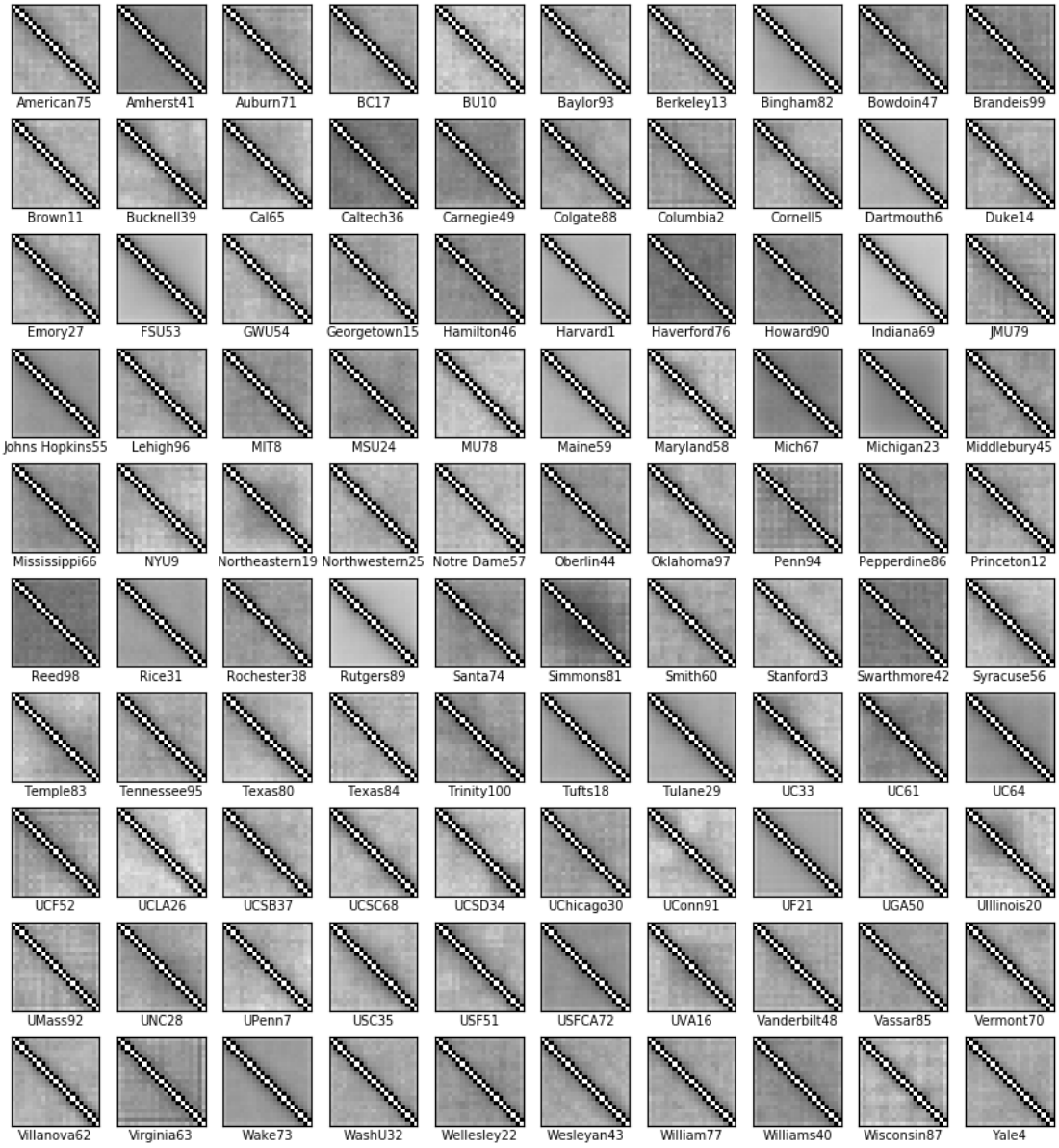

Figure 15: Matrices of Average Clustering Coefficients (MACC) for the 100 Facebook network data set using the chain motif $F = ([21], \mathbf{1}_{\{(1,2),(2,3),\cdots,(20,21)\}})$. Values of the entries are shown in greyscale with black = 1 and white = 0. The two main diagonals correspond to the edges in the motif $F_{0,20}$ and always have a value of 1. Each entry (i, j) for $|i - j| > 1$ equals to the probability of seeing the corresponding 'long-range' edge along a uniformly sampled copy of the chain motif.

6.1 MACCs for the Facebook100 dataset

The full MACCs of size 21 for the 100 facebook networks are shown in Figure 15. We used the chain motif $F = ([21], \mathbf{1}_{\{(1,2),(2,3),\cdots,(20,21)\}})$ of 20 edges, which we sampled from each network by Glauber chain (see Definition 2.1) for $2n \log n$ iterations, where n denotes the number of nodes in the given network, which we denote by $\mathcal{G}$. Each entry (i, j) of the MACC is computed by taking the time average in (49) with motifs F and $H = H_{ij} := ([21], \mathbf{1}_{\{(i,j)\}})$. This time average along the Glauber chain $F \to \mathcal{G}$ converges to a 21×21 matrix as shown in Figure 15. Note that the two main diagonals on $|i - j| = 1$ are always 1 as they correspond to the edges of the chain motif F embedded in the network. For $|i - j| > 1$, the (i, j) entry equals the conditional homomorphism density $\mathtt{t}(H_{ij}, \mathcal{G} \,|\, F)$, which is the probability of seeing the corresponding 'long-range' edge (i, j) along a uniformly sampled copy of the chain motif F from $\mathcal{G}$. We note that in Figure 15, in order to emphasize the off-diagonal entries, MACCs are plotted after the square-root transform.

MACC gives a natural and graphical generalization of the network clustering coefficient (see Example 3.1). For instance, consider the MACCs of Caltech, Harverford, Reed, Simmons, and Swarthmore in Figure 15. These are relatively small private or liberal arts schools, so one might expect to see stronger clustering among a randomly sampled chain of 21 users in their Facebook network. In fact, their MACCs show large values (dark) off of the two main diagonals, indicating that it is likely to see long-range connections along a randomly sampled chain F of 21 friends. On the other hand, the MACCs of Indiana, Rutgers, and UCLA show relatively small (light) values away from the two main diagonals, indicating that it is not very likely to see long-range friendships among a chain of 21 friends in their Facebook network. Indeed, they are large public schools so it is reasonable to see less clustered friendships in their social network.

6.2 Subgraph classification

In this section, we consider the subgraph classification problem in order to compare the performance of MACCs to that of classical network summary statistics such as edge density, diameter, and average clustering coefficient.

The problem setting is as follows. Suppose we have two networks $\mathcal{G}_1$ and $\mathcal{G}_2$, not necessarily of the same size. From each network $\mathcal{G}_i$, we sample 100 connected subgraphs of 30 nodes by running a simple symmetric random walk on $\mathcal{G}_i$ until it visits 30 distinct nodes and then taking the induced subgraph on the sampled nodes. Subgraphs sampled from network $\mathcal{G}_i$ get label i (see Figure 16 for examples of subgraphs subject to classification). Out of the total of 200 labeled subgraphs, we use 100 (50 from each network) to train several logistic regression classifiers corresponding to the input features consisting of various network summary statistics of the subgraphs — edge density, minimum degree, maximum degree, (shortest-path) diameter, degree assortativity coefficient (Newman, 2002), number of cliques, and average clustering coefficient — as well as MACCs at four scales $k \in \{5, 10, 15, 20\}$. Each trained logistic classifier is used to classify the remaining 100 labeled subgraphs (50 from each network). The performance is measured by using the area-under-curve (AUC) metric for the receiver-operating characteristic (ROC) curves.

We compare the performance of a total of 11 logistic classifiers trained on the various summary statistics of subgraphs described above using the seven Facebook social networks

Caltech, Simmons, Reed, NYU, Virginia, UCLA, and Wisconsin. There are total 21 pairs of distinct networks $(\mathcal{G}_1, \mathcal{G}_2)$ we consider for the subgraph classification task. For each pair of distinct networks, we repeated the same experiment ten times and reported the average AUC scores together with their standard deviations.

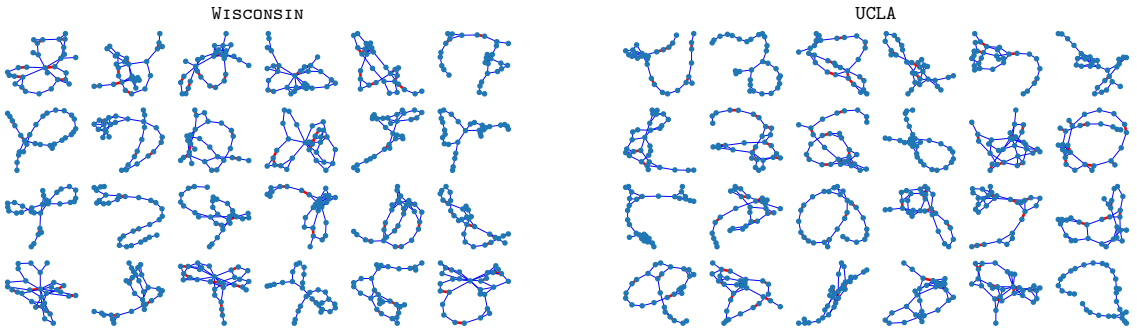

Figure 16: Examples of 30-node connected subgraphs from two Facebook social networks Wisconsin and UCLA Each subgraph is sampled by running a simple symmetric random walk on the network until visiting 30 distinct nodes and then taking the induced subgraph on the sampled nodes.

As we can see from the results reported in Table 17, classification using MACCs achieves the best performance in all 21 cases This indicates that MACCs are network summary statistics that are more effective in capturing structural information shared among subgraphs from the same network than the benchmark network statistics. Furthermore, observe that the classification performance using MACCs is mostly improved by increasing the scale parameter k. This show that MACCs do capture not only local scale information (recall that the average clustering coefficient is closely related to MACC with $k = 2$, see Example 3.1), but also the mesoscale (intermediate between local and global scales) structure of networks (Milo et al., 2002; Alon, 2007; Schwarze and Porter, 2020).

6.3 Clustering the Facebook networks via MACCs

For a more quantitative comparison of the MACCs in Figure 15, we show a multi-dimensional scaling of the MACCs together with cluster labels obtained by the k-means algorithm with $k = 5$ in Figure 18 Each school's Facebook network is represented by its 21×21 MACC, and mutual distance between two networks are measured by the Frobenius distance between their MACCs. Note that, as we can see from the corresponding MACCs, the five schools in the top left cluster in Figure 18 are private schools with a high probability of long-range connections, whereas all schools including UCLA, Rutgers, and Indiana in the bottom right cluster in Figure 18 have relatively sparse long-range edges. For a baseline comparison, we show the result of the k-means clustering of the same dataset only using the number of nodes and average degree for each network in Figure 19. The two clustering results have some similarities but also some interesting differences: The cluster that contains small private schools Caltech and Reed; The location of UCLA and USFCA with respect to other clusters. This shows qualitatively different clustering results can be obtained by using the local clustering structure of the networks encoded in their MACCs instead of the macroscopic information counting nodes and edges.

Networks	*edeg density*	*min degree*	*max degree*	*diameter*	*degree assortativity coef*	*# cliques*	*Avg clustering coeff*	*MACC (k=5)*	*MACC (k=10)*	*MACC (k=15)*	*MACC (k=20)*
Caltech36-Simmons81	0.914	0.618	0.774	0.863	0.638	0.738	0.762	0.932	0.947	0.966	**0.969**
Caltech36-Reed98	0.889	0.623	0.833	0.848	0.670	0.786	0.809	0.902	0.916	0.913	**0.924**
Caltech36-NYU9	0.902	0.666	0.795	0.831	0.408	0.699	0.775	0.923	**0.959**	0.950	0.944
Caltech36-Virginia63	0.822	0.55	0.704	0.822	0.724	0.715	0.813	0.900	0.924	0.928	**0.948**
Caltech36-UCLA26	0.873	0.594	0.783	0.848	0.631	0.724	0.770	0.888	0.943	0.946	**0.958**
Caltech36-Wisconsin87	0.749	0.713	0.687	0.784	0.527	0.663	0.697	0.794	0.873	0.896	**0.921**
Simmons81-Reed98	0.840	0.573	0.769	0.796	0.725	0.689	0.761	0.897	0.967	**0.977**	0.976
Simmons81-NYU9	0.922	0.596	0.841	0.884	0.693	0.770	0.833	0.934	0.954	**0.965**	0.952
Simmons81-Virginia63	0.802	0.448	0.691	0.766	0.664	0.603	0.731	0.802	0.812	**0.838**	0.826
Simmons81-UCLA26	0.856	0.566	0.755	0.812	0.669	0.753	0.821	0.910	0.919	**0.938**	0.935
Simmons81-Wisconsin87	0.877	0.645	0.819	0.896	0.651	0.745	0.754	0.934	0.959	**0.960**	**0.960**
Reed98-NYU9	0.822	0.607	0.672	0.787	0.562	0.712	0.724	0.896	0.934	**0.940**	0.926
Reed98-Virginia63	0.836	0.621	0.737	0.794	0.510	0.782	0.701	0.897	0.934	0.934	**0.941**
Reed98-UCLA26	0.822	0.628	0.697	0.770	0.521	0.786	0.747	0.891	0.958	**0.978**	0.974
Reed98-Wisconsin87	0.884	0.684	0.770	0.836	0.620	0.776	0.807	0.886	0.955	**0.956**	0.946
NYU9-Virginia63	0.858	0.568	0.817	0.79	0.655	0.807	0.797	0.868	0.897	**0.915**	0.880
NYU9-UCLA26	0.877	0.579	0.801	0.833	0.645	0.780	0.776	0.942	0.954	**0.964**	0.963
NYU9-Wisconsin87	0.889	0.641	0.793	0.855	0.673	0.744	0.780	0.932	0.954	0.957	**0.960**
Virginia63-UCLA26	0.914	0.604	0.839	0.822	0.640	0.782	0.801	0.927	0.947	**0.952**	0.946
Virginia63-Wisconsin87	0.839	0.574	0.783	0.850	0.637	0.684	0.730	0.888	0.904	0.891	**0.919**
UCLA26-Wisconsin87	0.820	0.588	0.766	0.810	0.653	0.699	0.724	0.866	**0.920**	0.913	0.912

Table 17: Performance benchmark on subgraph classification tasks. From the two Facebook social networks mentioned in in each row, 100 subgraphs of 30 nodes are sampled. Different logistic regression classifiers are then trained using eleven different statistics of the sampled subgraphs on a 50% training set. The classification performance on the other 50% test set is reported as the area-under-curve (AUC) for the receiver-operating characteristic (ROC) curves. The table shows the mean AUC over ten independent trials. The best performance in each case is marked in bold. The standard deviations are reported in Table 31 in the appendix.

We also show a single-linkage hierarchical clustering dendrogram of the MACCs in Figure 21, where two schools whose MACCs are Frobenius distance d away from each other merge into the same cluster at height d. Similarly, as we have seen in Figure 18, the rightmost cluster consisting of the private schools SIMMONS, HAVERFORD, CALTECH, and REED is separated by any other schools by distance as least 4; In the middle, we also observe the cluster of public schools including MARYLAND, RUTGERS, and UCONN. Lastly, we also provide a dendrogram using the baseline network metric using the normalized number of nodes and average degrees as used in Figure 19.

Lastly, we also remark that an entirely different approach for network data clustering as well as an application to the Facebook100 dataset is presented in Onnela et al. (2012). There, a given network's community structure is encoded as a profile of a 3-dimensional 'mesoscopic response function' (MRF), which is computed by the multiresolution Potts model for community detection with varying scaling parameters. MRFs encode the global community structure of a network, whereas MACCs capture local community structure at a chosen scale.

6.4 Computational complexity and remarks

We can estimate the the complexity of the MACC-based method as follows: each step of the Glauber chain update $\mathbf{x} \mapsto \mathbf{x}'$ has the complexity of order $O(k\Delta(F)\Delta(\mathbf{x}))$, where k

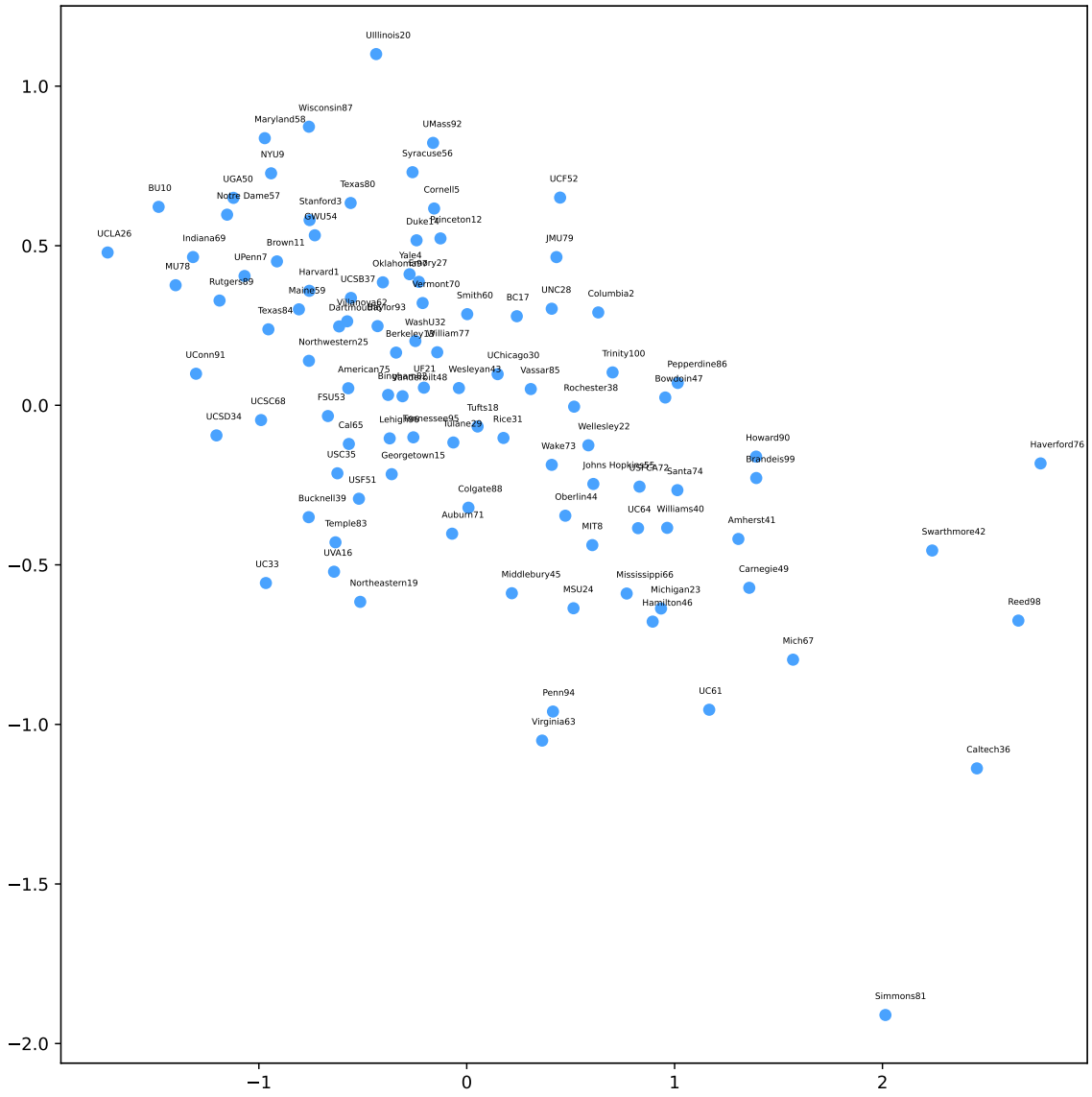

Figure 18: Multi-dimensional scaling plot of the MACCs of the Facebook100 dataset in Figure 15. We measured distance between two MACCs using the matrix Frobenius norm.

denotes the number of nodes in the motif F, and $\Delta(\cdot)$ denotes the maximum degree of a simple graph, and $\Delta(\cdot)$ denotes the maximum degree of the nodes in the image of the homomorphism $\mathbf{x}$ in $\mathcal{G}$. Note the trivial bound $\Delta(\mathbf{x}) \leq \Delta(\mathcal{G})$. By adding up these terms for a given number of iterations, the average time complexity of a single Glauber chain update is approximately $O(k\Delta(F) \cdot \texttt{avgdeg}(\mathcal{G}))$, where $\texttt{avgdeg}(\mathcal{G})$ denotes the average degree of $\mathcal{G}$. For dense networks, avgdeg($\mathcal{G}$) maybe large but the mixing time of the Glauber chain is small (see Theorem 2.4); for sparse networks, the Glauber chain takes longer to converge but $\texttt{avgdeg}(\mathcal{G})$ is small. In our experiments, we used $2n \log n$ steps of the Glauber chain for each

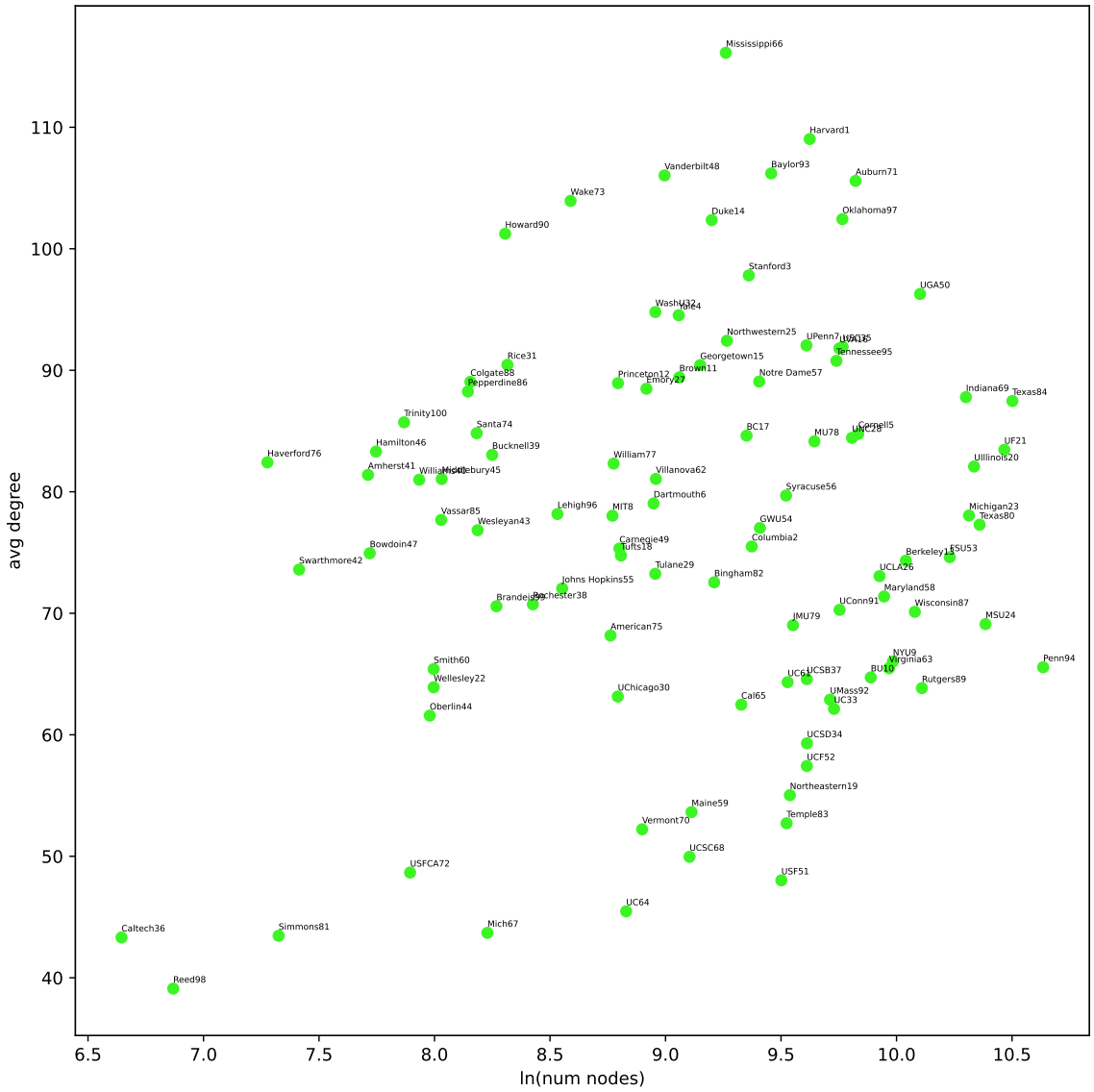

Figure 19: A two-dimensional scatter plot of the Facebook100 dataset using the average degree and the natural logarithm of the number of nodes.

network $\mathcal{G}$ with n nodes resulting in the total computational cost of $O(n \log n \cdot \text{avgdeg}(\mathcal{G}))$. One could use fewer iterations to quickly get a crude estimate.

We used a modest computational resource for our experiments: A quad-core 10th Gen. Intel Core i5-1035G7 Processor and 8 GB LPDDR4x RAM with Windows 10 operating system. The actual times for computing the MACCs shown in Figure 15 are shown in Table 30. The computation can easily be parallelized even for computing MACC for a single network. Indeed, since the MACC of a given network is computed by a Monte Carlo integration, one can use multiple Glauber chains on different cores and average the individual results to reduce the computation time by a large factor. All scripts for replicating the ex-

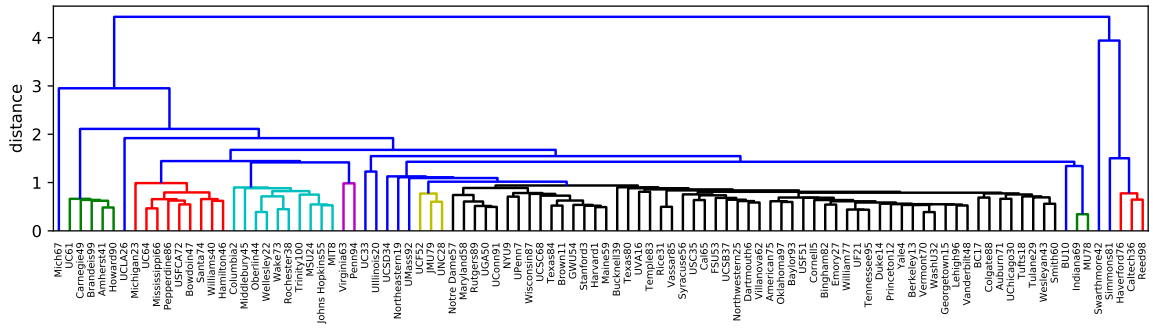

Figure 20: Single-linkage hierarchical clustering dendrogram of the Facebook100 dataset using the 21×21 matrices of average clustering coefficients (MACC) shown in Figure 15. Two schools with similar MACCs merge early in the dendrogram. Clusters emerging before level 1 are shown in non-blue colors.

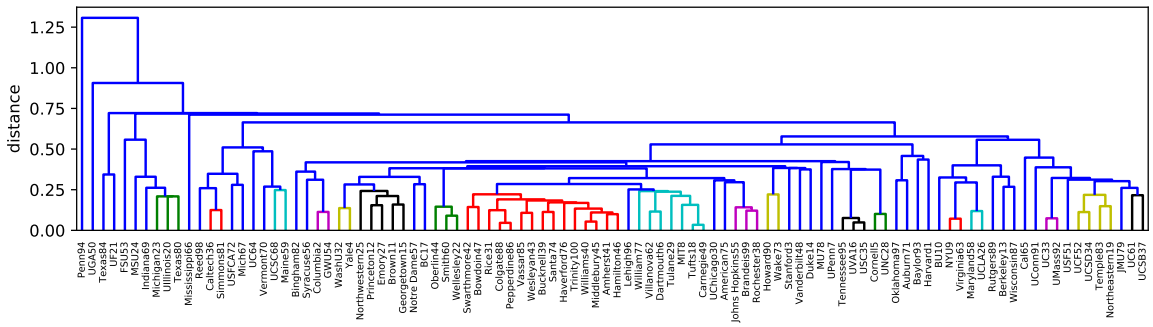

Figure 21: Single-linkage hierarchical clustering dendrogram of the Facebook100 dataset using the normalized L_2-distance using the number of nodes and average degree used in Figure 19. Clusters emerging before level 0.25 are shown in non-blue colors.

periments can be obtained from our GitHub repository https://github.com/HanbaekLyu/motif_sampling.

We close this section by pointing out some of the key advantages of our method for the network clustering problem. Our method can efficiently handle networks of different sizes without node labels. Indeed, note that the MACC of a given network is invariant under node relabeling, and regardless of the size of the network, we obtain a low-dimensional representation in the form of a MACC of fixed size, which can be tuned as a user-defined parameter (by making different choices of the underlying motif F). Also, as we have discussed in the previous paragraph, MACCs are interpretable in terms of the average clustering structure of the network so we can interpret the result of a clustering algorithm based on MACCs.

7. Application II: Textual analysis and Word Adjacency Networks

Function word adjacency networks (WANs) are weighted networks introduced by Segarra, Eisen, and Ribeiro in the context of authorship attribution (Segarra et al., 2015). Function words are words that are used for the grammatical purpose and do not carry lexical meaning

on their own, such as *the*, *and*, and *a* (see Segarra et al. (2015) for the full list of 211 function words). After fixing a list of n function words, for a given article $\mathcal{A}$, we construct a $(n \times n)$ *frequency matrix* $M(\mathcal{A})$ whose (i, j) entry m_{ij} is the number of times that the ith function word is followed by the jth function word within a forward window of $D = 10$ consecutive words (see Segarra et al. (2015) for details). For a given article $\mathcal{A}$, we associate a network $\mathcal{G}(\mathcal{A}) = ([n], A, \alpha)$, where $\alpha \equiv 1/n$ is the uniform node weight on the function words and A is a suitable matrix obtained from normalizing the frequency matrix $M(\mathcal{A})$. In Segarra et al. (2015), the authors used row-wise normalization to turn the frequency matrix into a Markov transition kernel and then used Kullback–Leibler (KL) divergence to compare them for a classification task. Use the same normalization for the same purpose (see Table 23). In all other simulations, we use the global normalization $A = M(\mathcal{A})/\max(M(\mathcal{A}))$ as it leads to more visually distinctive CHD profiles among different authors (see, e.g., Figure 35).

The particular data set we will analyze in this section consists of the following 45 novels of the nine authors listed below:

1. *Jacob Abbott*: Caleb in the Country, Charles I, Cleopatra, Cyrus the Great, and Darius the Great

2. *Thomas Bailey Aldrich*: Marjorie Daw, The Cruise of the Dolphin, The Little Violinist, Mademoiselle Olympe Zabriski, and A Midnight Fantasy

3. *Jane Austen*: Northanger Abbey, Emma, Mansfield Park, Pride and Prejudice, and Sense and Sensibility

4. *Grant Allen*: The British Barbarians, Biographies of Working Men, Anglo-Saxon Britain, Charles Darwin, and An African Millionaire

5. Charles Dickens: A Christmas Carol, David Copperfield, Bleak House, Oliver Twist, and Holiday Romance

6. *Christopher Marlowe*: Edward the Second, The Tragical History of Doctor Faustus, The Jew of Malta, Massacre at Paris, and Hero and Leander and Other Poems

7. *Herman Melville*: Israel Potter, The Confidence-Man, Moby Dick; or The Whale, Omoo: Adventures in the South Seas, and Typee

8. *William Shakespeare*: Hamlet, Henry VIII, Julius Cesar, King Lear, and Romeo and Juliet

9. *Mark Twain*: Adventures of Huckleberry Finn, A Horse's Tale, The Innocents Abroad, The Adventures of Tom Sawyer, and A Tramp Abroad

The frequency matrices corresponding to the above novels are recorded using a list of $n = 211$ function words (see supplementary material of Segarra et al. (2015)). These matrices are sparse and spiky, meaning that most entries are zero and that there are a few entries that are very large compared to the others. For their visual representation, in the first row in Figure 3, we plot the heat map of some of the frequency matrices after a double 'log transform' $A \mapsto \log(A + \mathbf{1})$ and then normalization $B \mapsto B/\max(B)$.

Next, we find that the corresponding WANs contain one large connected component and a number of isolated nodes. This can be seen effectively by performing single-linkage hierarchical clustering on these networks. In Figure 33 we plot the resulting single-linkage dendrograms for two novels: "Jane Austen - Pride and Prejudice" and "William Shakespeare - Hamlet". In both novels, the weight between the function words "of" and "the" is the maximum and they merge at level 1 (last two words in Figure 33 top and the fourth and fifth to last in Figure 33 bottom). On the other hand, function words such as "yet" and "whomever" are isolated in both networks (first two words in Figure 33 top and bottom).

7.1 CHD profiles of the novel data set

We compute various CHD profiles of the WANs corresponding to our novel data set. We consider the following three pairs of motifs: (see Example 3.2)

$$(H_{0,0}, H_{0,0}): \qquad H_{0,0} = (\{0\}, \mathbf{1}_{\{(0,0)\}}) \tag{77}$$

$$(F_{0,1}, F_{0,1}): \qquad F_{0,1} = (\{0,1\}, \mathbf{1}_{\{(0,1)\}}) \tag{78}$$

$$(H_{1,1}, F_{1,1}): \qquad H_{1,1} = (\{0,1,2\}, \mathbf{1}_{\{(1,2)\}}), \qquad F_{1,1} = (\{0,1,2\}, \mathbf{1}_{\{(0,1),(0,2)\}}). \tag{79}$$

The CHD profiles of WANs corresponding to the 45 novels are given in Figures 35, 37, and 39.

Figure 35 as well as the second row of Figure 3 below show the CHD profiles $\mathtt{f}(H_{0,0}, \mathcal{G} \,|\, H_{0,0})$ for the pair of 'self-loop' motifs. At each filtration level $t \in [0,1]$, the value $\mathtt{f}(t)$ of the profile, in this case, means roughly the density of self-loops in the network $\mathcal{G}_{\mathcal{A}}$ whose edge weight exceed t. In terms of the function words, the larger value of $\mathtt{f}(t)$ indicates that more function words are likely to be repeated in a given $D = 10$ chunk of words. All of the five CHD profiles for Jane Austen drop to zero quickly and vanishes after $t = 0.4$. This means that in her five novels, function words are not likely to be repeated frequently in a short distance. This is in a contrast to the corresponding five CHD profiles for Mark Twain. The rightmost long horizontal bars around height 0.4 indicate that, among the function words that are repeated within a 10-ward window at least once, at least 40% of them are repeated almost with the maximum frequency. In this regard, from the full CHD profiles given in Figure 35, the nine authors seem to divide into two groups. Namely, Jane Austen, Christopher Marlowe, and William Shakespeare have their $(0,0)$ CHD profiles vanishing quickly (less frequent repetition of function words), and the other five with persisting $(0,0)$ CHD profiles (more frequent repetition of function words).

Figure 37 shows the CHD profiles $\mathtt{f}(F_{0,1}, \mathcal{G} \,|\, F_{0,1})$. The value $\mathtt{f}(t)$ of the CHD profile in this case can be viewed as the tail probability of a randomly chosen edge weight in the network, where the probability of each edge (i,j) is proportional to the weight $A(i,j)$. The CHD profiles for Mark Twain seem to persist longer than that of Jane Austen as in the self-loop case, the difference is rather subtle in this case.

Lastly, Figure 39 shows the CHD profiles $\mathtt{f}(H_{1,1}, \mathcal{G} \,|\, F_{1,1})$. The value $\mathtt{f}(t)$ of the CHD profile in this case can be regarded as a version of the average clustering coefficient for the corresponding WAN (see Example 3.1). Namely, the value $\mathtt{f}(t)$ of the profile at level t is the conditional probability that two random nodes with a common neighbor are connected by an edge with intensity $\geq t$. In terms of function words, this is the probability that if we randomly choose three function words x, y, and z such that x and y are likely to appear

shortly after z, then y also appear shortly after x with more than a proportion t of all times. The corresponding profiles suggest that for Jane Austen, two function words with commonly associated function words are likely to have a very weak association. On the contrary, for Mark Twain, function words tend to be more strongly clustered. From Figure 39, one can see that the $(1,1)$ CHD profile of Shakespeare exhibits fast decay in a manner similar to Jane Austen's CHD profiles. While the five CHD profiles of most authors are similar, Grant Allan and Christopher Marlowe show somewhat more significant differences in their CHD profiles among different novels.

7.2 Authorship attribution by CHD profiles

In this subsection, we analyze the CHD profiles of the dataset of novels more quantitatively by computing the pairwise L^1-distances between the CHD profiles. Also, we discuss an application in authorship attribution.

In order to generate the distance matrices, we partition the 45 novels into 'validation set' and 'reference set' of sizes 9 and 36, respectively, by randomly selecting a novel for each author. Note that there are a total 5^9 such partitions. For each article i in the validation set, and for each of the three pairs of motifs, we compute the L^1-distance between the corresponding CHD profile of the article i and the mean CHD profile of each of the nine authors, where the mean profile for each author is computed by averaging the four profiles in the reference set. This will give us a 9×9 matrix of L^1-distances between the CHD profiles of the nine authors. We repeat this process for 10^4 iterations to obtain a $9 \times 9 \times 10^4$ array. The average of all 10^4 distance matrices for each of the three pairs of motifs are shown in Figure 32.

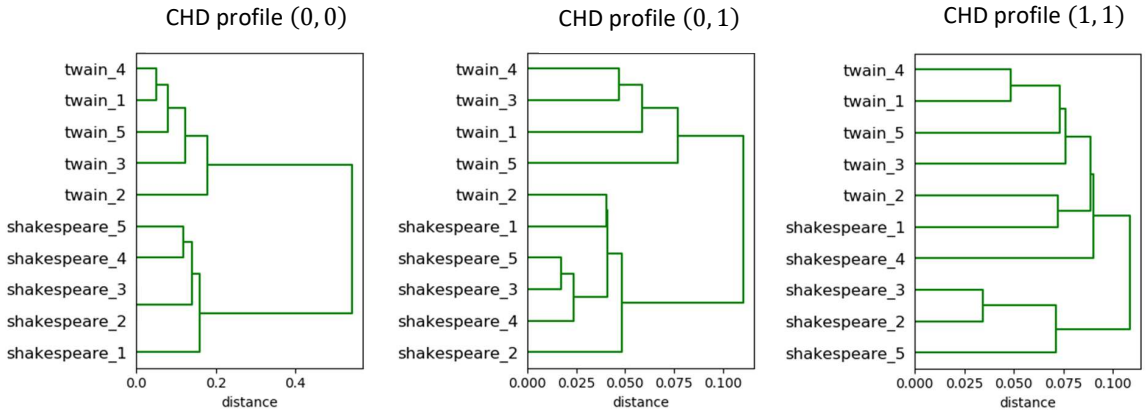

Figure 22: Single-linkage dendrogram of the L^1-distance matrices between the CHD profiles of the 10 novels of Shakespeare and Twain for the pair of motifs (H_{00}, F_{00}) (left), (H_{01}, F_{01}) (middle), and (H_{11}, F_{11}) (right). Most texts of the same author fall into the same cluster.

For instance, consider the middle plot in Figure 32. The plot suggests that Jane Austen and William Shakespeare have small L^1-distance with respect to their CHD profiles $\mathtt{f}(F_{0,1}, \mathcal{G} \,|\, F_{0,1})$. From the full list of CHD profiles given in Figure 37, we can see 'why' this is so: while their CHD profiles drop to zero quickly around filtration level 0.5, all the other authors have more persisting CHD profiles.

Next, note that if we restrict the three distance matrices to the last two authors (Twain and Shakespeare), then the resulting 2×2 matrices have small diagonal entries indicating that the two authors are well-separated by the three profiles. Indeed, in Figure 22, we plot the single-linkage hierarchical clustering dendrogram for the L^1-distances across the 10 novels by the two authors. In the first dendrogram in Figure 22, we see the five novels from each author form perfect clusters according to the $(0,0)$-CDH profile. For the other two dendrograms, we observe near-perfect clustering using $(0,1)$- and $(1,1)$-CHD profiles.

Lastly, in Table 23, we apply our method as a means of authorship attribution and compare its correct classification rate with other baseline methods. We choose five novels for each author as described at the beginning of this section. In this experiment, for each article $\mathcal{A}$, we normalize its frequency matrix $M(\mathcal{A})$ row-wise to make it a Markov transition kernel and then calculate pairwise distances between them by three methods – L^1-the distance between $(0,0)$-CHD profiles, the KL-divergence, and the Frobenius distance. This normalization is used in the original article (Segarra et al., 2015), and we find that this generally leads to a higher classification rate than the global normalization $M(\mathcal{A}) \mapsto M(\mathcal{A})/\max(M(\mathcal{A}))$. For the classification test, we first choose $k \in \{1,2,3,4\}$ known texts and one text with unknown authorship from each author. For each unknown text X, we compute its distance from the $5k$ texts of known authorship and attribute X to the author of known texts of the minimum distance. The classification rates after repeating this experiment 1000 times are reported in Table 23.

	CHD profile (0,0)				KL-divergence				Frobenius			
# known texts	1	2	3	4	1	2	3	4	1	2	3	4
Abbott	0.87	0.92	0.95	1.00	0.61	0.81	0.79	0.81	0.72	0.76	0.76	0.85
Austen	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.75	0.80	0.81	0.86
Marlowe	0.52	0.74	0.74	0.72	0.54	0.61	0.66	0.71	0.24	0.26	0.33	0.36
Shakespeare	0.35	0.46	0.61	0.68	0.92	0.95	0.97	1.00	0.47	0.66	0.85	1.00
Twain	0.53	0.67	0.76	0.80	0.61	0.78	0.79	0.81	0.33	0.36	0.32	0.33
Average	**0.65**	**0.76**	**0.81**	**0.84**	**0.73**	**0.83**	**0.84**	**0.87**	**0.50**	**0.57**	**0.61**	**0.68**

Table 23: Success rate of authorship attribution by using CHD profiles, the KL divergence, and the Frobenius metric for various numbers of known texts per author.

The table above summarizes classification rates among the five authors - Abbott, Austen, Marlowe, Shakespeare, and Twain. For four known texts per author, the CHD profile gives 84% success rate, which outperforms the Frobenius distance (68%) and shows similar performance as the KL divergence (87%). It is also interesting to note that different metric shows complementary classification performance for some authors. For instance, for four known tests, Abbott is perfectly classified by the CHD profile, whereas KL-divergence has only %81 success rate; on the other hand, Shakespeare is perfectly classified by the KL-divergence but only with %68 accuracies with the CHD profile. We also report the average classification rates for all nine authors: CHD profile $(0,0)$ – 53%, KL-divergence – 77%, and Frobenius – 41%. The $(0,0)$-profile loses the classification score mainly for Aldrich (author index 1), Dickens (author index 5), and Melville (author index 7). Indeed, in Figure 32

left, we see the diagonal entries of these authors are not the smallest in the corresponding rows. A WAN is already a compressed mathematical summary of text data, so running an additional MCMC motif sampling algorithm and further compressing it to a profile may lose information that could simply directly be processed. We emphasize that, as we have seen in Section 6 as well as in Subsection 7.1, our method is more suitable for extracting interpretable and low-dimensional information from large networks.

Acknowledgments

This work has been partially supported by NSF projects DMS-1723003 and CCF-1740761. HL is partially supported by NSF DMS-2206296 and DMS-2010035. The authors are grateful to Mason Porter for sharing the Facebook100 dataset, and also to Santiago Segarra and Mark Eisen for sharing the Word Adjacency Network dataset and original codes.

References

Ahmet Alacaoglu and Hanbaek Lyu. Convergence and complexity of stochastic subgradient methods with dependent data for nonconvex optimization. *arXiv preprint arXiv:2203.15797*, 2022.

Mariam Alaverdian, William Gilroy, Veronica Kirgios, Xia Li, Carolina Matuk, Daniel McKenzie, Tachin Ruangkriengsin, Andrea L Bertozzi, and P Jeffrey Brantingham. Who killed lilly kane? a case study in applying knowledge graphs to crime fiction. In *2020 IEEE International Conference on Big Data (Big Data)*, pages 2508–2512. IEEE, 2020.

Uri Alon. Network motifs: Theory and experimental approaches. *Nature Reviews Genetics*, 8(6):450–461, 2007.

Albert-László Barabási. Network science. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 371(1987):20120375, 2013.

Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.

Itai Benjamini, Oded Schramm, et al. Recurrence of distributional limits of finite planar graphs. *Electronic Journal of Probability*, 6, 2001.

Gunnar Carlsson. Topology and data. *Bulletin of the American Mathematical Society*, 46 (2):255–308, 2009.

Gunnar Carlsson and Facundo Mémoli. Characterization, stability and convergence of hierarchical clustering methods. *Journal of machine learning research*, 11(Apr):1425–1470, 2010.

Gunnar Carlsson, Facundo Mémoli, Alejandro Ribeiro, and Santiago Segarra. Excisive hierarchical clustering methods for network data. *arXiv preprint arXiv:1607.06339*, 2016.

Gunnar Carlsson, Facundo Mémoli, Alejandro Ribeiro, and Santiago Segarra. Representable hierarchical clustering methods for asymmetric networks. In Francesco Palumbo, Angela Montanari, and Maurizio Vichi, editors, *Data Science*, pages 83–95, Cham, 2017. Springer International Publishing. ISBN 978-3-319-55723-6.

Gunnar E. Carlsson and Facundo Mémoli. Classifying clustering schemes. *Foundations of Computational Mathematics*, 13(2):221–252, 2013. doi: 10.1007/s10208-012-9141-9. URL https://doi.org/10.1007/s10208-012-9141-9.

Gunnar E Carlsson, Facundo Mémoli, et al. Characterization, stability and convergence of hierarchical clustering methods. *J. Mach. Learn. Res.*, 11(Apr):1425–1470, 2010.

Frédéric Chazal, David Cohen-Steiner, Leonidas J Guibas, Facundo Mémoli, and Steve Y Oudot. Gromov-Hausdorff stable signatures for shapes using persistence. In *Computer Graphics Forum*, volume 28, pages 1393–1403. Wiley Online Library, 2009.

Samir Chowdhury and Facundo Mémoli. Distances and isomorphism between networks and the stability of network invariants. *arXiv preprint arXiv:1708.04727*, 2017.

Samir Chowdhury and Facundo Mémoli. The metric space of networks. *arXiv preprint arXiv:1804.02820*, 2018a.

Samir Chowdhury and Facundo Mémoli. A functorial dowker theorem and persistent homology of asymmetric networks. *Journal of Applied and Computational Topology*, 2(1-2): 115–175, 2018b.

Samir Chowdhury and Facundo Mémoli. Persistent path homology of directed networks. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1152–1169. SIAM, 2018c.

Samir Chowdhury and Facundo Mémoli. The Gromov-Wasserstein distance between networks. *Information and Inference (to appear)*, 2019.

David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Stability of persistence diagrams. *Discrete & Computational Geometry*, 37(1):103–120, 2007.

Don Coppersmith, Prasad Tetali, and Peter Winkler. Collisions among random walks on a graph. *SIAM Journal on Discrete Mathematics*, 6(3):363–374, 1993.

Emanuele Cozzo, Mikko Kivelä, Manlio De Domenico, Albert Solé-Ribalta, Alex Arenas, Sergio Gómez, Mason A Porter, and Yamir Moreno. Structure of triadic relations in multiplex networks. *New Journal of Physics*, 17(7):073029, 2015.

Martin Dyer, Catherine Greenhill, and Mike Molloy. Very rapid mixing of the glauber dynamics for proper colorings on bounded-degree graphs. *Random Structures & Algorithms*, 20(1):98–114, 2002.

Herbert Edelsbrunner and John Harer. *Computational topology: an introduction*. American Mathematical Soc., 2010.

Herbert Edelsbrunner and Dmitriy Morozov. Persistent homology: theory and practice. Technical report, Lawrence Berkeley National Lab.(LBNL), Berkeley, CA (United States), 2012.

Herbert Edelsbrunner, David Letscher, and Afra Zomorodian. Topological persistence and simplification. In *Foundations of Computer Science, 2000. Proceedings. 41st Annual Symposium on*, pages 454–463. IEEE, 2000.

Paul Erdős and Alfréd Rényi. On random graphs. I. *Publicationes Mathematicae*, 6(18): 290–297, 1959.

Robert W Floyd. Algorithm 97: shortest path. *Communications of the ACM*, 5(6):345, 1962.

Alan Frieze and Eric Vigoda. A survey on the use of markov chains to randomly sample colourings. *Oxford Lecture Series in Mathematics and its Applications*, 34:53, 2007.

Robert Ghrist. Barcodes: the persistent topology of data. *Bulletin of the American Mathematical Society*, 45(1):61–75, 2008.

Thomas P Hayes. A large-deviation inequality for vector-valued martingales. *Combinatorics, Probability and Computing*, 2005.

Paul W. Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels: First steps. *Social Networks*, 5(2):109–137, 1983.

Nicholas Jardine and Robin Sibson. Mathematical taxonomy. Technical report, 1971.

Mark Jerrum. A very simple algorithm for estimating the number of k-colorings of a low-degree graph. *Random Structures & Algorithms*, 7(2):157–165, 1995.

Olav Kallenberg and Rafal Sztencel. Some dimension-free features of vector-valued martingales. *Probability Theory and Related Fields*, 88(2):215–247, 1991.

Tosio Kato. *Perturbation theory for linear operators*, volume 132. Springer Science & Business Media, 2013.

Eric D Kolaczyk and Gábor Csárdi. *Statistical analysis of network data with R*, volume 65. Springer, 2014.

David A Levin and Yuval Peres. *Markov chains and mixing times*, volume 107. American Mathematical Soc., 2017.

Jun S Liu. *Monte Carlo strategies in scientific computing*. Springer Science & Business Media, 2008.

László Lovász. *Large networks and graph limits*, volume 60. American Mathematical Soc., 2012.

László Lovász and Balázs Szegedy. Limits of dense graph sequences. *Journal of Combinatorial Theory, Series B*, 96(6):933–957, 2006.

R Duncan Luce and Albert D Perry. A method of matrix analysis of group structure. *Psychometrika*, 14(2):95–116, 1949.

Hanbaek Lyu, Deanna Needell, and Laura Balzano. Online matrix factorization for markovian data and applications to network dictionary learning. *Journal of Machine Learning Research*, 21(251):1–49, 2020.

Hanbaek Lyu, Yacoub H Kureh, Joshua Vendrow, and Mason A Porter. Learning low-rank latent mesoscale structures in networks. *arXiv preprint arXiv:2102.06984*, 2021.

Facundo Mémoli and Guilherme Vituri F. Pinto. Motivic clustering schemes for directed graphs. *CoRR*, abs/2001.00278, 2020. URL http://arxiv.org/abs/2001.00278.

Facundo Mémoli and Kritika Singhal. A primer on persistent homology of finite metric spaces. *Bulletin of mathematical biology*, pages 1–43, 2019.

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

Mark Newman. *Networks*. Oxford university press, 2018a.

Mark E. J. Newman. *Networks*. Oxford University Press, Oxford, UK, second edition, 2018b.

Mark EJ Newman. Assortative mixing in networks. *Physical review letters*, 89(20):208701, 2002.

Jukka-Pekka Onnela, Daniel J Fenn, Stephen Reid, Mason A Porter, Peter J Mucha, Mark D Fricker, and Nick S Jones. Taxonomies of networks from community structure. *Physical Review E*, 86(3):036104, 2012.

Nina Otter, Mason A Porter, Ulrike Tillmann, Peter Grindrod, and Heather A Harrington. A roadmap for the computation of persistent homology. *EPJ Data Science*, 6(1):17, 2017.

Daniel Paulin et al. Concentration inequalities for Markov chains by marton couplings and spectral methods. *Electronic Journal of Probability*, 20, 2015.

Jianhao Peng, Chao Pan, Hanbaek Lyu, Minji Kim, Albert Cheng, and Olgica Milenkovic. Inferring single-molecule chromatin interactions via online convex network dictionary learning. *bioRxiv*, 2022.

Luana Ruiz, Luiz FO Chamon, and Alejandro Ribeiro. Graphon signal processing. *IEEE Transactions on Signal Processing*, 69:4961–4976, 2021.

Jesús Salas and Alan D Sokal. Absence of phase transition for antiferromagnetic potts models via the dobrushin uniqueness theorem. *Journal of Statistical Physics*, 86(3-4): 551–579, 1997.

Alice C Schwarze and Mason A Porter. Motifs for processes on networks. *arXiv preprint arXiv:2007.07447*, 2020.

Santiago Segarra, Mark Eisen, and Alejandro Ribeiro. Authorship attribution through function word adjacency networks. *IEEE Transactions on Signal Processing*, 63(20):5464–5478, 2015.

Zane Smith, Samir Chowdhury, and Facundo Mémoli. Hierarchical representations of network data with optimal distortion bounds. In *2016 50th Asilomar Conference on Signals, Systems and Computers*, pages 1834–1838. IEEE, 2016.

Amanda L Traud, Peter J Mucha, and Mason A Porter. Social structure of facebook networks. *Physica A: Statistical Mechanics and its Applications*, 391(16):4165–4180, 2012.

Katharine Turner. Rips filtrations for quasimetric spaces and asymmetric functions with stability results. *Algebraic & Geometric Topology*, 19(3):1135–1170, 2019.

Eric Vigoda. Improved bounds for sampling colorings. *Journal of Mathematical Physics*, 41 (3):1555–1569, 2000.

Stephen Warshall. A theorem on boolean matrices. *Journal of the ACM (JACM)*, 9(1): 11–12, 1962.

Duncan J Watts and Steven H Strogatz. Collective dynamics of "small-world" networks. *nature*, 393(6684):440–442, 1998.

Hao Yin, Austin R Benson, and Jure Leskovec. Higher-order clustering in networks. *Physical Review E*, 97(5):052306, 2018.

Afra Zomorodian and Gunnar Carlsson. Computing persistent homology. *Discrete & Computational Geometry*, 33(2):249–274, 2005.

Appendix A. Motif transforms and spectral decomposition

In this section, we compute the motif transform by paths using a certain spectral decomposition and consider motif transforms in terms of graphons. We denote the path and cycle motifs by $P_k = ([k], \mathbf{1}_{\{(1,2),(2,3),\cdots,(k-1,k)\}})$ and $C_k = ([k], \mathbf{1}_{\{(1,2),\cdots,(k-1,k),(k,1)\}})$, respectively.

A.1 Motif transform by paths

For any function $f : [n] \to [0,1]$, denote by $\mathrm{diag}(f)$ the $(n \times n)$ diagonal matrix whose (i,i) entry is $f(i)$. For a given network $\mathcal{G} = ([n], A, \alpha)$, observe that

$$\mathtt{t}(P_k, \mathcal{G})\alpha(x_1)^{-1/2} A^{P_k}(x_1, x_k)\alpha(x_k)^{-1/2} = \sum_{x_2,\cdots,x_{k-1}\in[n]} \prod_{\ell=1}^{k-1} \sqrt{\alpha(x_\ell)} A(x_\ell, x_{\ell+1}) \sqrt{\alpha(x_{\ell+1})} \tag{80}$$

$$= \left[\left(\mathrm{diag}(\sqrt{\alpha})\, A\, \mathrm{diag}(\sqrt{\alpha}) \right)^{k-1} \right]_{x_1, x_k}. \tag{81}$$

If we denote $B = \mathrm{diag}(\sqrt{\alpha})\, A\, \mathrm{diag}(\sqrt{\alpha})$, this yields

$$\mathtt{t}(P_k, \mathcal{G}) A^{P_k} = \mathrm{diag}(\sqrt{\alpha}) B^{k-1} \mathrm{diag}(\sqrt{\alpha}). \tag{82}$$

Since B is a real symmetric matrix, its eigenvectors form an orthonormal basis of $\mathbb{R}^n$. Namely, let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$ be the eigenvalues of B and let v_i be the corresponding eigenvector of λ_i. Then v_i and v_j are orthogonal if $i \neq j$. Furthermore, we may normalize the eigenvectors so that if we let V be the $(n \times n)$ matrix whose ith column is v_i, then $V^T V = I_n$, the $(n \times n)$ identity matrix. The spectral decomposition for B gives $B = V\, \mathrm{diag}(\lambda_1, \cdots, \lambda_n)\, V^T$. Hence

$$\mathtt{t}(P_k, \mathcal{G}) A^{P_k} = \mathrm{diag}(\sqrt{\alpha}) V\, \mathrm{diag}(\lambda_1^{k-1}, \cdots, \lambda_n^{k-1})\, V^T \mathrm{diag}(\sqrt{\alpha}), \tag{83}$$

or equivalently,

$$\mathtt{t}(P_k, \mathcal{G}) A^{P_k}(i,j) = \sum_{\ell=1}^{n} \lambda_\ell^{k-1} \sqrt{\alpha(i)} v_\ell(i) \sqrt{\alpha(j)} v_\ell(j), \tag{84}$$

where $v_\ell(i)$ denotes the ith coordinate of the eigenvector v_ℓ. Summing the above equation over all i, j gives

$$\mathtt{t}(P_k, \mathcal{G}) = \sum_{\ell=1}^{n} \lambda_\ell^{k-1} (\langle \sqrt{\alpha}, v_\ell \rangle)^2, \tag{85}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product between two vectors in $\mathbb{R}^n$. Combining the last two equations yields

$$A^{P_k}(i,j) = \frac{\sum_{\ell=1}^{n} \lambda_\ell^{k-1} \sqrt{\alpha(i)} v_\ell(i) \sqrt{\alpha(j)} v_\ell(j)}{\sum_{\ell=1}^{n} \lambda_\ell^{k-1} \langle \sqrt{\alpha}, v_\ell \rangle^2}, \tag{86}$$

Now suppose $\mathcal{G}$ is irreducible. Then by Perron-Frobenius theorem for nonnegative irrducible matrices, λ_1 is the eigenvalue of B with maximum modulus whose associated

eigenspace is simple, and the components of the corresponding normalized eigenvector v_1 are all positive. This yields $\langle\sqrt{\alpha}, v_1\rangle > 0$, and consequently

$$\bar{A} := \lim_{k\to\infty} A^{P_k} = \frac{1}{\langle\sqrt{\alpha}, v_1\rangle^2}\mathrm{diag}(\sqrt{\alpha})v_1v_1^T\mathrm{diag}(\sqrt{\alpha}). \tag{87}$$

If $\mathcal{G}$ is not irreducible, then the top eigenspace may not be simple and $\lambda_1 = \cdots = \lambda_r > \lambda_{r+1}$ for some $1 \le r < n$. By decomposing A into irreducible blocks and applying the previous observation, we have $\langle\sqrt{\alpha}, v_i\rangle > 0$ for each $1 \le i \le r$ and

$$\bar{A} := \lim_{k\to\infty} A^{P_k} = \frac{1}{\sum_{i=1}^r\langle\sqrt{\alpha}, v_1\rangle^2}\mathrm{diag}(\sqrt{\alpha})\left(\sum_{i=1}^r v_iv_i^T\right)\mathrm{diag}(\sqrt{\alpha}). \tag{88}$$

We denote $\bar{\mathcal{G}} = ([n], \bar{A}, \alpha)$ and call this network as the *transitive closure* of $\mathcal{G}$.

It is well-known that the Perron vector of an irreducible matrix A, which is the normalized eigenvector corresponding to the Perron-Frobenius eigenvalue λ_1 of A, varies continuously under small perturbation of A, as long as resulting matrix is still irreducible Kato (2013). It follows that the transitive closure $\bar{\mathcal{G}}$ of an irreducible network $\mathcal{G}$ is stable under small perturbation. However, it is easy to see that this is not the case for reducible networks (see Example A.5).

Below we give an example of motif transform by paths and transitive closure of a three-node network.

Example A.1 (Transitive closure of a three-node network) Consider a network $\mathcal{G} = ([3], A, \alpha)$, where $\alpha = ((1-\epsilon)/2, \epsilon, (1-\epsilon)/2)$ and

$$A = \begin{bmatrix} 1 & s & 0 \\ s & 1 & s \\ 0 & s & 1 \end{bmatrix}. \tag{89}$$

Then $\mathcal{G}$ is irreducible if and only if $s > 0$. Suppose $s > 0$. Also, note that

$$\mathrm{diag}(\sqrt{\alpha})\,A\,\mathrm{diag}(\sqrt{\alpha}) = \begin{bmatrix} (1-\epsilon)/2 & s\sqrt{(1-\epsilon)\epsilon/2} & 0 \\ s\sqrt{(1-\epsilon)\epsilon/2} & \epsilon & s\sqrt{(1-\epsilon)\epsilon/2} \\ 0 & s\sqrt{(1-\epsilon)\epsilon/2} & (1-\epsilon)/2 \end{bmatrix}. \tag{90}$$

The eigenvalues of this matrix are

$$\begin{aligned} \lambda_0 &= \frac{1-\epsilon}{2} \\ \lambda_- &= \frac{1}{4}\left((\epsilon+1) - \sqrt{(3\epsilon-1)^2 - 16s^2\epsilon(1-\epsilon)}\right) \\ \lambda_+ &= \frac{1}{4}\left((\epsilon+1) + \sqrt{(3\epsilon-1)^2 - 16s^2\epsilon(1-\epsilon)}\right) \end{aligned}$$

and the corresponding eigenvectors are

$$v_0 = (-1, 0, 1)^T \tag{91}$$

$$v_- = \left(1, \frac{3\epsilon - 1 - \sqrt{(3\epsilon-1)^2 + 16s^2\epsilon(1-\epsilon)}}{2s\sqrt{2\epsilon(1-\epsilon)}}, 1\right)^T \tag{92}$$

$$v_+ = \left(1, \frac{3\epsilon - 1 + \sqrt{(3\epsilon-1)^2 + 16s^2\epsilon(1-\epsilon)}}{2s\sqrt{2\epsilon(1-\epsilon)}}, 1\right)^T \tag{93}$$

The Perron-Frobenius eigenvector of the matrix in (90) is v_+. Then using (87), we can compute

$$\bar{A} = \begin{bmatrix} 1/4 & 0 & 1/4 \\ 0 & 0 & 0 \\ 1/4 & 0 & 1/4 \end{bmatrix} + \epsilon \begin{bmatrix} -s & s & -s \\ s & 0 & s \\ -s & s & -s \end{bmatrix} + O(\epsilon^2). \tag{94}$$

Hence in the limit as $\epsilon \searrow 0$, the transitive closure of $\mathcal{G}$ consists of two clusters with uniform communication strength of $1/4$. However, if we change the order of limits, that is, if we first let $\epsilon \searrow 0$ and then $k \to \infty$, then the two clusters do not communicate in the limit. Namely, one can compute

$$A^{P_k} = \begin{bmatrix} 1/2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1/2 \end{bmatrix} + \epsilon \begin{bmatrix} -(k-1)s^2 - 2s & s & ks^2 \\ s & 0 & s \\ ks^2 & s & -(k-1)s^2 - 2s \end{bmatrix} + O(\epsilon^2), \tag{95}$$

which is valid for all $k \geq 2$. Hence for any fixed $k \geq 1$, the motif transform of $\mathcal{G}$ by P_k gives two non-communicating clusters as $\epsilon \searrow 0$. ▲

A.2 Motif transform of graphons

Recall the n-block graphon $U_{\mathcal{G}} : [0,1]^2 \to [0,1]$ associated with network $\mathcal{G} = ([n], A, \alpha)$, which is introduced in (52). For each graphon U and a simple motif $F = ([k], \mathbf{1}(E))$ with $k \geq 2$, define a graphon U^F by

$$U^F(x_1, x_k) = \frac{1}{\mathtt{t}(F, U)} \int_{[0,1]^{k-2}} \prod_{(i,j)\in E} U(x_i, x_j)\, dx_2 \cdots dx_{k-1}. \tag{96}$$

It is easy to verify that the graphon corresponding to the motif transforms $\mathcal{G}^F$ agrees with $(U_{\mathcal{G}})^F$. Below we give some examples.

Example A.2 (path) Let P_k be the path motif on node set $[k]$. Let $U : [0,1]^2 \to [0,1]$ be a graphon. Then

$$\mathtt{t}(P_k, U) U^{P_k}(x_1, x_k) = \int_{[0,1]^{k-2}} U(x_1, x_2) U(x_2, x_3) \cdots U(x_{k-1}, x_k)\, dx_2 \cdots dx_{k-1}. \tag{97}$$

We denote the graphon on the right-hand side as $U^{\circ(k-1)}$, which is called the $(k-1)$st power of U. ▲

Example A.3 (cycle) Let C_k be a cycle motif on node set $[k]$. Let $U : [0,1]^2 \to [0,1]$ be a graphon. Then

$$\mathtt{t}(C_k, U)U^{C_k}(x_1, x_k) = U(x_1, x_k) \int_{[0,1]^{k-2}} U(x_1, x_2) \cdots U(x_{k-1}, x_k)\, dx_2 \cdots dx_{k-1} \tag{98}$$

$$= U(x_1, x_k)U^{\circ(k-1)}(x_1, x_k). \tag{99}$$

▲

Below, we give an explicit example of motif transforms applied to graphons.

Example A.4 Let $\mathcal{G} = ([3], A, \alpha)$ be the network in Example A.1. Let $U_{\mathcal{G}}$ be the corresponding graphon. Namely, let $[0,1] = I_1 \sqcup I_2 \sqcup I_3$ be a partition where $I_1 = [0, (1-\varepsilon)/2)$, $I_2 = [(1-\varepsilon)/2, (1+\varepsilon)/2)$, and $I_3 = [(1+\varepsilon)/2, 1]$. Then $U_{\mathcal{G}}$ is the 3-block graphon taking value $A(i,j)$ on rectangle $I_i \times I_j$ for $1 \le i, j \le 3$. Denoting $U = U_{\mathcal{G}}$, the three graphons U, $U^{\circ 2}$, and $U \cdot U^{\circ 2}$ are shown in Figure 24.

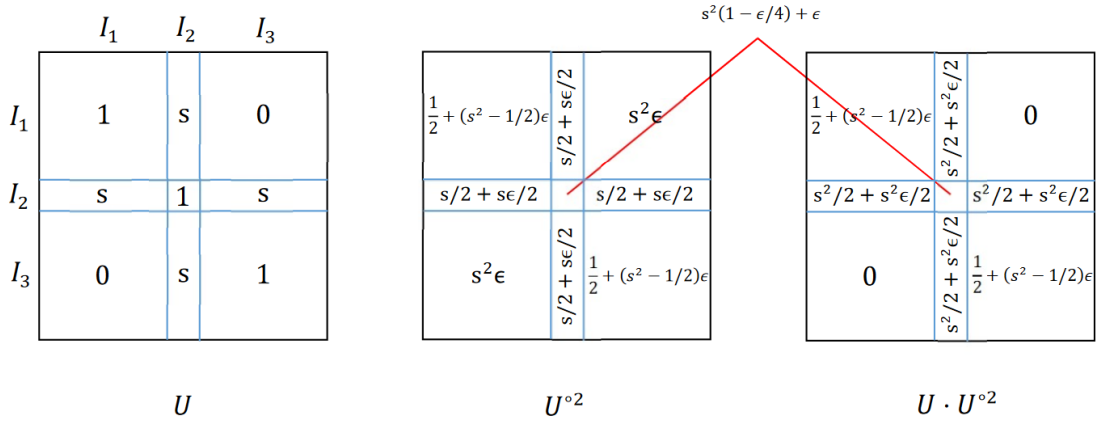

Figure 24: Graphons $U = U_{\mathcal{G}}$ (left), $U^{\circ 2}$ (middle), and $U \cdot U^{\circ 2}$ (right). These are three-block graphons with the same block structure $I_i \times I_j$ for $1 \le i, j \le 3$ whose values on each block are depicted in the figure.

According to the previous examples, we have

$$U^{P_3} = \frac{U^{\circ 2}}{\mathtt{t}(P_3, U)}, \quad U^{C_3} = \frac{U \cdot U^{\circ 2}}{\mathtt{t}(C_3, U)}, \tag{100}$$

where $\mathtt{t}(P_3, U) = \|U^{\circ 2}\|_1$ and $\mathtt{t}(P_3, U) = \|U \cdot U^{\circ 2}\|_1$. See Figure 11 for hierarchical clustering dendrograms of these graphons. ▲

A.3 Spectral decomposition and motif transform by paths

In this subsection, we assume all kernels and graphons are symmetric.

A graphon $W : [0,1]^2 \to [0,1]$ induces a compact Hilbert-Schmidt operator T_W on $\mathcal{L}^2[0,1]$ where

$$T_W(f)(x) = \int_0^1 W(x,y) f(y)\, dy. \tag{101}$$

T_W has a discrete spectrum, i.e., its spectrum is a countable multiset $\mathrm{Spec}(W) = \{\lambda_1, \lambda_2, \cdots\}$, where each eigenvalue has finite multiplicity and $|\lambda_n| \to 0$ as $n \to \infty$. Since W is assumed to be symmetric, all λ_is are real so we may arrange them so that $\lambda_1 \geq \lambda_2 \geq \cdots$. Via a spectral decomposition, we may write

$$W(x,y) = \sum_{j=1}^{\infty} \lambda_j f_j(x) f_j(y), \tag{102}$$

where $\int_0^1 f_i(x) f_j(x)\, dx = \mathbf{1}(i=j)$, that is, f_j is an eigenfunction associated to λ_j and they form an orthonormal basis for $\mathcal{L}^2[0,1]$.

Let P_k be the path on the node set $[k]$. Let U be a graphon with eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots$. Orthogonality of the eigenfunctions easily yields

$$U^{P_k}(x,y) = \frac{\sum_j \lambda_j^k f_j(x) f_j(y)}{\sum_j \lambda_j^k \left(\int f_j(x_1)\, dx_1\right)^2}, \tag{103}$$

Further, suppose the top eigenvalue of U has multiplicity $r \geq 1$. Then

$$\bar{U}(x,y) = \lim_{k\to\infty} U^{P_k}(x,y) = \frac{\sum_{j=1}^r f_j(x) f_j(y)}{\sum_{j=1}^r \left(\int f_j(x_1)\, dx_1\right)^2}. \tag{104}$$

Note that (103) and (104) are the graphon analogues of formulas (84) and (88).

The network and graphon versions of these formulas are compatible through the following simple observation.

Proposition A.1 *Let $\mathcal{G} = ([n], A, \alpha)$ be a network such that A is symmetric, and let $U = U_{\mathcal{G}}$ be the corresponding graphon (see (52)). Let $\lambda \in \mathbb{R}$ and $v = (v_1, \cdots, v_n)^T \in \mathbb{R}^n$ be a pair of an eigenvalue and its associate eigenvector of the matrix $B = \mathrm{diag}(\sqrt{\alpha})\, A\, \mathrm{diag}(\sqrt{\alpha})$. Then the following function $f_v : [0,1] \to \mathbb{R}$*

$$f_v(x) = \sum_{i=1}^{n} \frac{v_i}{\sqrt{\alpha(i)}} \mathbf{1}(x \in I_i) \tag{105}$$

is an eigenfunction of the integral operator T_U associated to the eigenvalue λ. Conversely, every eigenfunction of T_U is given this way.

Proof First observe that any eigenfunction f of T_U must be constant over each interval I_i. Hence we may write $f = \sum a_i \mathbf{1}(I_i)$ for some $a_i \in \mathbb{R}$. Then for each $x \in [0,1]$,

$$\begin{aligned} T_U(f_v)(x) &= \int_0^1 U(x,y) f(y)\, dy && (106)\\ &= \int_0^1 \sum_{i,j,k} A(i,j) \mathbf{1}(x \in I_i) \mathbf{1}(y \in I_j)\, a_k\, \mathbf{1}(y \in I_k)\, dy && (107)\\ &= \sum_i \mathbf{1}(x \in I_i) \sum_j A(i,j)\, \alpha(j)\, a_j. && (108) \end{aligned}$$

Hence f is an eigenfunction of T_U with eigenvalue λ if and only if

$$\sum_{j=1}^{n} A(i,j)\alpha(j)a_j = \lambda a_i \quad \forall 1 \le i \le n, \tag{109}$$

which is equivalent to saying that $u := (a_1, \cdots, a_n)^T$ is an eigenvector of the matrix $A\,\mathrm{diag}(\alpha)$ with eigenvalue λ. Further note that $A\,\mathrm{diag}(\alpha)u = \lambda u$ is equivalent to

$$B\,\mathrm{diag}(\sqrt{\alpha})u = \lambda \mathrm{diag}(\sqrt{\alpha})u. \tag{110}$$

Writing $v_i := a_i\sqrt{\alpha(i)}$, then shows the assertion. ■

Remark A.1 (Ruiz et al., 2021, Lem. 2) states a similar observation for associating eigenvalue/eigenvector pairs of a network with that of the associated graphon. While our statement holds for general probability distribution α on the node set $[n]$, in the reference, the uniform probability distribution $\alpha \equiv 1/n$ is assumed. In this special case, (109) reduces to

$$Au = \frac{\lambda}{n}u, \tag{111}$$

as stated in (Ruiz et al., 2021, Lem. 2).

When a graphon U is not irreducible, its top eigenspace is not simple and its dimension can change under an arbitrarily small perturbation. Hence formula (104) suggests that the operation of transitive closure $U \to \bar{U}$ is not stable under any norm. The following example illustrates this.

Example A.5 (Instability of transitive closure) Let $f_1 = \mathbf{1}([0,1])$ and choose a function $f_2 : [0,1] \to \{-1,1\}$ so that $\int_0^1 f_2(x)\,dx = 0$. Then $\|f_2\|_2 = 1$ and $\langle f_1, f_2\rangle = 0$. Now fix $\epsilon > 0$, and define two graphons U and U_ϵ through their spectral decompositions

$$U = f_1 \otimes f_1 + f_2 \otimes f_2 \quad \text{and} \quad U_\epsilon = f_1 \otimes f_1 + (1-\epsilon) f_2 \otimes f_2, \tag{112}$$

where $(f_i \otimes f_j)(x,y) = f_i(x)f_j(y)$. Then by (104), we get $\bar{U} = U$ and $\bar{U}_\epsilon = f_1 \otimes f_1$. This yields

$$\varepsilon(\bar{U} - \bar{U}_\epsilon) = \varepsilon f_2 \otimes f_2 = U - U_\epsilon. \tag{113}$$

▲

Appendix B. Proof of convergence and mixing time bounds of the Glauber and pivot chains

In this section, we establish convergence and mixing properties of the Glauber and pivot chains of homomorphisms $F \to \mathcal{G}$ by proving Theorems 2.1, 2.4, 2.2, 2.5, and Corollary 3.1.

B.1 Convergence and mixing of the pivot chain

Let $(\mathbf{x}_t)_{t\geq 0}$ be a pivot chain of homomorphisms $F \to \mathcal{G}$. We first show that the pivot chain converges to the desired distribution $\pi_{F\to\mathcal{G}}$ over $[n]^{[k]}$, defined in (9). Recall the α is the unique stationary distribution of the simple random walk on $\mathcal{G}$ with the modified kernel (21). In this subsection, we write a rooted tree motif $F = ([k], A_F)$ as $([k], E_F)$, where $E_F = \{(i,j) \in [k]^2 \,|\, A_F(i,j) = 1\}$.

Proof of Theorem 2.2 . Since the network $\mathcal{G}$ is irreducible and finite, the random walk $(\mathbf{x}_t(1))_{t\geq 0}$ of pivot on $\mathcal{G}$ with kernel P defined at (21) is also irreducible. It follows that the pivot chain is irreducible with a unique stationary distribution, say, π. We show π is in fact the desired measure $\pi_{F\to\mathcal{G}}$. First, recall that $\mathbf{x}_t(1)$ is a simple random walk on the network $\mathcal{G}$ modified by the Metropolis-Hastings algorithm so that it has the following marginal distribution as its unique stationary distribution: (see, e.g., (Levin and Peres, 2017, Sec. 3.2))

$$\pi^{(1)}(x_1) = \frac{\sum_{x_2,\cdots,x_k\in[n]} \prod_{(i,j)\in E_F} A(x_i,x_j)\alpha(x_1)\alpha(x_2)\cdots\alpha(x_k)}{\mathtt{t}(F,\mathcal{G})} \tag{114}$$

Second, we decompose $\mathbf{x}_t$ into return times of the pivot $\mathbf{x}_t(1)$ to a fixed node $x_1 \in [n]$ in $\mathcal{G}$. Namely, let $\tau(\ell)$ be the ℓth return time of $\mathbf{x}_t(1)$ to x_1. Then by independence of sampling $\mathbf{x}_t$ over $\{2,\cdots,k\}$ for each t, the strong law of large numbers yields

$$\lim_{M\to\infty} \frac{1}{M} \sum_{\ell=1}^{M} \mathbf{1}(\mathbf{x}_{\tau(\ell)}(2) = x_2, \cdots, \mathbf{x}_{\tau(\ell)}(k) = x_k) \tag{115}$$

$$= \frac{\prod_{\{i,j\}\in E_F} A(x_i,x_j)\alpha(x_2)\cdots\alpha(x_k)}{\sum_{x_2,\cdots,x_k\in[n]} \prod_{\{i,j\}\in E_F} A(x_i,x_j)\alpha(x_2)\cdots\alpha(x_k)}. \tag{116}$$

Now, for each fixed homomorphism $\mathbf{x} : F \to \mathcal{G}$, $i \mapsto x_i$, we use the Markov chain ergodic theorem and previous estimates to write

$$\pi(\mathbf{x}) = \lim_{N\to\infty} \frac{1}{N} \sum_{t=0}^{N} \mathbf{1}(\mathbf{x}_t = \mathbf{x}) \tag{117}$$

$$= \lim_{N\to\infty} \frac{\sum_{t=0}^{N} \mathbf{1}(\mathbf{x}_t = \mathbf{x})}{\sum_{t=0}^{N} \mathbf{1}(\mathbf{x}_t(1) = x_1)} \frac{\sum_{t=0}^{N} \mathbf{1}(\mathbf{x}_t(1) = x_1)}{N} \tag{118}$$

$$= \frac{\prod_{\{i,j\}\in E_F} A(x_i,x_j)\alpha(x_2)\cdots\alpha(x_k)}{\sum_{1\leq x_2,\cdots,x_k\leq n} \prod_{\{i,j\}\in E_F} A(x_i,x_j)\alpha(x_2)\cdots\alpha(x_k)} \pi^{(1)}(x_1) \tag{119}$$

$$= \frac{\prod_{\{i,j\}\in E_F} A(x_i,x_j)\alpha(x_1)\alpha(x_2)\cdots\alpha(x_k)}{\mathtt{t}(F,\mathcal{G})} = \pi_{F\to\mathcal{G}}(\mathbf{x}). \tag{120}$$

This shows the assertion. ∎

Next, we bound the mixing time of the pivot chain. Our argument is based on the well-known bounds on the mixing time and meeting time of random walks on graphs.

Proof of Theorem 2.5. Fix a rooted tree motif $F = ([k], E_F)$ and a network $\mathcal{G} = ([n], A, \alpha)$. Let P denote the transition kernel of the random walk of pivot on $\mathcal{G}$ given at (21). Note that (ii) follows immediately from the equality in (i) and known bounds on mixing times of random walks (see, e.g., (Levin and Peres, 2017, Thm 12.3 and 12.4)).

Now we show (i). The entire pivot chain and the random walk of the pivot have the same mixing time after each move of the pivot, since the pivot converges to the correct marginal distribution $\pi^{(1)}$ induced from the joint distribution $\pi_{F\to\mathcal{G}}$, and we always sample the non-pivot nodes from the correct distribution conditioned on the location of the pivot. To make this idea more precise, let $\mathbf{y} : [k] \to [n]$ be an arbitrary homomorphism $F \to \mathcal{G}$ and let $(\mathbf{x}_t)_{t\geq 0}$ denote the pivot chain $F \to \mathcal{G}$ with $\mathbf{x}_0 = \mathbf{y}$. Write $\pi = \pi_{F\to\mathcal{G}}$ and π_t for the distribution of $\mathbf{x}_t$. Let $\pi^{(1)}$ denote the unique stationary distribution of the pivot $(\mathbf{x}_t(1))_{t\geq 0}$. Let $\mathbf{x} : F \to \mathcal{G}$ be a homomorphism and write $\mathbf{x}(1) = x_1$. Then for any $t \geq 0$, note that

$$\mathbb{P}(\mathbf{x}_t = \mathbf{x} \,|\, \mathbf{x}_t(1) = x_1) = \frac{\left(\prod_{(i,j)\in E_F} A(x_i, x_j)\right) \alpha(x_2)\cdots\alpha(x_k)}{\sum_{x_2,\cdots,x_k\in[n]} \left(\prod_{(i,j)\in E_F} A(x_i, x_j)\right) \alpha(x_2)\cdots\alpha(x_k)} \tag{121}$$

$$= \mathbb{P}_\pi(\mathbf{x}_t = \mathbf{x} \,|\, \mathbf{x}_t(1) = x_1). \tag{122}$$

Hence we have

$$|\pi_t(\mathbf{x}) - \pi(\mathbf{x})| = |\mathbb{P}(\mathbf{x}_t(1) = x_1) - \pi^{(1)}(x_1)| \cdot \mathbb{P}(\mathbf{x}_t = \mathbf{x} \,|\, \mathbf{x}_t(1) = x_1). \tag{123}$$

Thus summing the above equation over all homomorphisms $\mathbf{x} : F \to \mathcal{G}$, we get

$$\|\pi_t - \pi\|_{\mathrm{TV}} = \frac{1}{2} \sum_{\mathbf{x}:[k]\to[n]} |\pi_t(\mathbf{x}) - \pi(\mathbf{x})| \tag{124}$$

$$= \frac{1}{2} \sum_{x_1\in[n]} |\mathbb{P}(\mathbf{x}_t(1) = x_1) - \pi^{(1)}(x_1)| \tag{125}$$

$$= \|P^t(\mathbf{y}(1), \cdot) - \pi^{(1)}(x_1)\|_{\mathrm{TV}}. \tag{126}$$

This shows (i).

To show (iii), let $(X_t)_{t\geq 0}$ and $(Y_t)_{t\geq 0}$ be two independent random walks on $\mathcal{G}$ with kernel P, where at each time t we choose one of them independently with equal probability to move. Let t_M be the first time that these two chains meet, and let τ_M be their worst-case expected meeting time, that is,

$$\tau_M = \max_{x_0,y_0\in[n]} \mathbb{E}[t_M \,|\, X_0 = x_0, Y_0 = y_0]. \tag{127}$$

Then by a standard coupling argument and Markov's inequality, we have

$$\|P^t(x, \cdot) - \alpha\|_{\mathrm{TV}} \leq \mathbb{P}(X_t \neq Y_t) = \mathbb{P}(t_M > t) \leq \frac{\tau_M}{t}. \tag{128}$$

By imposing the last expression to be bounded by $1/4$, this yields $t_{mix}(1/4) \leq 4\tau_M$. Hence we get

$$t_{mix}(\varepsilon) \leq 4\tau_M \log_2(\varepsilon^{-1}). \tag{129}$$

Now under the hypothesis in (iii), there is a universal cubic upper bound on the meeting time τ_M due to Coppersmith, Tetali, and Winkler (Coppersmith et al., 1993, Thm. 3). This shows (iii). ∎

Lastly in this subsection, we prove Corollary 3.1 for the pivot chain. The assertion for the Glauber chain follows similarly from Theorem 2.1, which will be proved in Subsection B.3.

Proof of Corollary 3.1. Let $F = ([k], E_F)$ be a directed tree motif and $\mathcal{G} = ([n], A, \alpha)$ be an irreducible network. Let $(\mathbf{x}_t)_{t\geq 0}$ be a pivot chain of homomorphisms $F \to \mathcal{G}$ and let $\pi := \pi_{F\to\mathcal{G}}$ be its unique stationary distribution. To show (50), note that

$$\lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} \prod_{1\leq i,j\leq k} \mathbf{1}(A(\mathbf{x}_t(i), \mathbf{x}_t(j))^{A_H(i,j)} \geq t) \tag{130}$$

$$= \mathbb{E}_\pi \left[\prod_{1\leq i,j\leq k} \mathbf{1}(A(\mathbf{x}_t(i), \mathbf{x}_t(j))^{A_H(i,j)} \geq t) \right] \tag{131}$$

$$= \mathbb{P}_{F\to\mathcal{G}} \left(\min_{1\leq i,j\leq k} A(\mathbf{x}(i), \mathbf{x}(j))^{A_H(i,j)} \geq t \right), \tag{132}$$

where the first equality is due to Theorem 2.3. In order to show (49), note that

$$\lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} \prod_{1\leq i,j\leq k} A(\mathbf{x}_t(i), \mathbf{x}_t(j))^{A_H(i,j)} \tag{133}$$

$$= \mathbb{E}_\pi \left[\prod_{1\leq i,j\leq k} A(\mathbf{x}(i), \mathbf{x}(j))^{A_H(i,j)} \right] \tag{134}$$

$$= \sum_{\mathbf{x}:[k]\to[n]} \left(\prod_{1\leq i,j\leq k} A(\mathbf{x}(i), \mathbf{x}(j))^{A_H(i,j)} \right) \frac{\left[\prod_{(i,j)\in E_F} A(\mathbf{x}(i), \mathbf{x}(j))\right] \alpha(\mathbf{x}(1)) \cdots \alpha(\mathbf{x}(k))}{\mathtt{t}(F, \mathcal{G})} \tag{135}$$

$$= \sum_{\mathbf{x}:[k]\to[n]} \left(\prod_{1\leq i,j\leq k} A(\mathbf{x}(i), \mathbf{x}(j))^{A_H(i,j)+A_F(i,j)} \right) \frac{\alpha(\mathbf{x}(1)) \cdots \alpha(\mathbf{x}(k))}{\mathtt{t}(F, \mathcal{G})} \tag{136}$$

$$= \frac{\mathtt{t}(H, \mathcal{G})}{\mathtt{t}(F, \mathcal{G})} = \mathtt{t}(F + H, \mathcal{G} \,|\, F). \tag{137}$$

For the last equation (51), we fix $x_1, x_k \in [n]$. By definition, we have

$$A^H(x_1, x_k) = \mathbb{E}_{\pi_{H\to\mathcal{G}}} \left[\mathbf{1}(\mathbf{x}(1) = x_1,\, \mathbf{x}(k) = x_k) \right]. \tag{138}$$

By similar computation as above, we can write

$$\lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} \left(\prod_{1\le i,j\le k} A(\mathbf{x}_t(i), \mathbf{x}_t(j))^{A_H(i,j)} \right) \mathbf{1}(\mathbf{x}_t(1) = x_1,\, \mathbf{x}_t(k) = x_k) \tag{139}$$

$$= \mathbb{E}_\pi \left[\left(\prod_{1\le i,j\le k} A(\mathbf{x}_t(i), \mathbf{x}_t(j))^{A_H(i,j)} \right) \mathbf{1}(\mathbf{x}(1) = x_1,\, \mathbf{x}(k) = x_k) \right] \tag{140}$$

$$= \mathtt{t}(F + H, \mathcal{G} \,|\, F)\, \mathbb{E}_{\pi_{H\to\mathcal{G}}} \left[\mathbf{1}(\mathbf{x}(1) = x_1,\, \mathbf{x}(k) = x_k) \right]. \tag{141}$$

Hence the assertion follows from (138). ■

B.2 Concentration of the pivot chain and rate of convergence

Proof of Theorem 2.6. This is a direct consequence of McDirmid's inequality for Markov chains (Paulin et al., 2015, Cor. 2.11) and the first equality in Theorem 2.5 (i). ■

Next, we prove Theorem 2.7. An essential step is given by the following lemma, which is due to Hayes (2005) and Kallenberg and Sztencel (1991). Let $\mathcal{H}$ be Hilbert space, and let $(X_t)_{t\ge 0}$ be a sequence of $\mathcal{H}$-valued random 'vectors'. We say it is a *very-weak martingale* if $X_0 = \mathbf{0}$ and

$$\mathbb{E}[X_{t+1} \,|\, X_t] = X_t \qquad \forall t \ge 0. \tag{142}$$

Lemma B.1 (Thm. 1.8 in (Hayes, 2005)) *Let $(X_t)_{t\ge 0}$ be a very-weak martingale taking values in a Hilbert space $\mathcal{H}$ and $\|X_{t+1} - X_t\| \le 1$ for all $t \ge 0$. Then for any $a > 0$ and $t \ge 0$,*

$$\mathbb{P}(\|X_t\| \ge a) \le 2e^2 \exp\left(\frac{-a^2}{2t} \right). \tag{143}$$

Proof The original statement (Hayes, 2005, Thm. 1.8) is asserted for a Euclidean space $\mathbb{E}$ in place of the Hilbert space $\mathcal{H}$. The key argument is given by a discrete-time version of a Theorem of Kallenberg and Sztencel (1991, Thm. 3.1), which is proved by Hayes in (Hayes, 2005, Prop. 1.5) for Euclidean space. The gist of the argument is that given a very-weak martingale $(X_t)_{t\ge 0}$ in a Euclidean space with norm $\|\cdot\|$, we can construct a very-weak martingale $(Y_t)_{t\ge 0}$ in R^2 in such a way that

$$(\|X_t\|, \|X_{t+1}\|, \|X_{t+1} - X_t\|) = (\|Y_t\|_2, \|Y_{t+1}\|_2, \|Y_{t+1} - Y_t\|_2). \tag{144}$$

By examining the proof of Hayes (2005, Prop. 1.5), one finds that the existence of such a 2-dimensional 'local martingale' is guaranteed by an inner product structure and completeness with respect to the induced norm of the underlying space. Hence the same conclusion holds for Hilbert spaces. ■

Proof of Theorem 2.7. We use a similar coupling idea that is used in the proof of Levin and Peres (2017, Thm. 12.19). Recall that $t_{mix} \equiv t_{mix}^{(1)}$ by Theorem 2.5 (i). Fix an integer $r \geq t_{mix}^{(1)}(\varepsilon) = t_{mix}(\varepsilon)$. Let $\Omega = [n]^{[k]}$ and fix a homomorphism $x : F \to \mathcal{G}$ for the initial state of the pivot chain $(\mathbf{x}_t)_{t\geq 0}$. Let π_t denote the law of $\mathbf{x}_t$ and let $\pi := \pi_{F\to\mathcal{G}}$. Let μ_r be the optimal coupling between π_t and π, so that

$$\sum_{\mathbf{x}\neq\mathbf{y}} \mu_r(\mathbf{x},\mathbf{y}) = \|\pi_t - \pi\|_{TV}. \tag{145}$$

We define a pair $(\mathbf{y}_t, \mathbf{z}_t)$ of pivot chains such that 1) The law of $(\mathbf{y}_0, \mathbf{z}_0)$ is μ_r and 2) individually $(\mathbf{y}_t)_{t\geq 0}$ and $(\mathbf{z}_t)_{t\geq 0}$ are pivot chains $F \to \mathcal{G}$, and 3) once these two chains meet, they evolve in unison. Note that $(\mathbf{y}_t)_{t\geq 0}$ has the same law as $(\mathbf{x}_{r+t})_{t\geq 0}$. Also note that by the choice of r and μ_r,

$$\mathbb{P}(\mathbf{y}_0 \neq \mathbf{z}_0) = \|\pi_t - \pi\|_{TV} \leq \varepsilon. \tag{146}$$

Now let $\mathcal{H}$ be a Hilbert space and let $g : \Omega \to \mathcal{H}$ be any function. By subtracting $\mathbb{E}_\pi(g(\mathbf{x}))$ from g, we may assume $\mathbb{E}_\pi(g(\mathbf{x})) = 0$. Then by conditioning on whether $\mathbf{y}_0 = \mathbf{z}_0$ or not, we have

$$\mathbb{P}\left(\left\|\sum_{t=1}^{N} g(\mathbf{x}_{r+t})\right\| \geq N\delta\right) = \mathbb{P}\left(\left\|\sum_{t=1}^{N} g(\mathbf{y}_t)\right\| \geq N\delta\right) \tag{147}$$

$$\leq \mathbb{P}\left(\left\|\sum_{t=1}^{N} g(\mathbf{z}_t)\right\| \geq N\delta\right) + \mathbb{P}(\mathbf{y}_0 \neq \mathbf{z}_0). \tag{148}$$

The last term is at most ε by (146), and we can apply Lemma B.1 for the first term. This gives the assertion. ∎

B.3 Convergence and mixing of the Glauber chain

In this subsection, we consider convergence and mixing of the Glauber chain $(\mathbf{x}_t)_{t\geq 0}$ of homomorphisms $F \to \mathcal{G}$. We first investigate under what conditions the Glauber chain is irreducible.

For two homomorphisms $\mathbf{x}, \mathbf{x}' : F \to \mathcal{G}$, denote $\mathbf{x} \sim \mathbf{x}'$ if they differ by at most one coordinate. Define a graph $\mathcal{S}(F,\mathcal{G}) = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is the set of all graph homomorphisms $F \to \mathcal{G}$ and $\{\mathbf{x}, \mathbf{x}'\} \in \mathcal{E}$ if and only if $\mathbf{x} \sim \mathbf{x}'$. We say $\mathbf{x}'$ is *reachable from* $\mathbf{x}$ *in* r *steps* if there exists a walk between $\mathbf{x}'$ and $\mathbf{x}$ of length r in $\mathcal{S}(F,\mathcal{G})$. Lastly, denote the shortest path distance on $\mathcal{S}(F,\mathcal{G})$ by $d_{F,\mathcal{G}}$. Then $d_{F,\mathcal{G}}(\mathbf{x},\mathbf{x}') = r$ if $\mathbf{x}'$ is reachable from $\mathbf{x}$ in r steps and r is as small as possible. It is not hard to see that the Glauber chain $(\mathbf{x}_t)_{t\geq 0}$ is irreducible if and only if $\mathcal{S}(F,\mathcal{G})$ is connected. In the following proposition, we show that this is the case when F is a tree motif and $\mathcal{G}$ contains an odd cycle.

Proposition B.1 *Suppose $F = ([k], A_F)$ is a tree motif and $\mathcal{G} = ([n], A, \alpha)$ is irreducible and bidirectional network. Further, assume that the skeleton of $\mathcal{G}$ contains an odd cycle. Then $\mathcal{S}(F,\mathcal{G})$ is connected and*

$$\operatorname{diam}(\mathcal{S}(F,\mathcal{G})) \leq 2k\operatorname{diam}(\mathcal{G}) + 4(k-1). \tag{149}$$

Proof We may assume $\mathtt{t}(F,\mathcal{G}) > 0$ since otherwise $\mathcal{S}(F,\mathcal{G})$ is empty and hence is connected. If $k = 1$, then each Glauber update is to sample the location of 1 uniformly at random from $[n]$, so the assertion holds. We may assume $k \geq 2$.

We first give a sketch of the proof of connectedness of $\mathcal{S}(F,\mathcal{G})$. Since $\mathcal{G}$ is bidirectional, we can fold the embedding $\mathbf{x} : F \to \mathcal{G}$ until we obtain a copy of K_2 (complete graph with two nodes) that is still a valid embedding $F \to \mathcal{G}$. One can also 'contract' the embedding $\mathbf{x}'$ in a similar way. By using irreducibility, then one can walk these copies of K_2 in $\mathcal{G}$ until they completely overlap. Each of these moves occurs with positive probability since $\mathcal{G}$ is bidirectional, and the issue of parity in matching the two copies of K_2 can be handled by 'going around' the odd cycle in $\mathcal{G}$.

Below we give a more careful argument for the above sketch. Fix two homomorphisms $\mathbf{x}, \mathbf{x}' : F \to \mathcal{G}$. It suffices to show that $\mathbf{x}'$ is reachable from $\mathbf{x}$ in $2k\mathrm{diam}(\mathcal{G}) + 4(k-1)$ steps. Choose a any two nodes $\ell, \ell' \in [k]$ such that ℓ is a leaf in F (i.e., $A_F(\ell, i) = 0$ for all $i \in [k]$) and they have a common neighbor in F (i.e., $A_F(i,\ell) > 0$ and $A_F(i,\ell') + A_F(\ell', i) > 0$ for some $i \in [k]$). Consider the vertex map $\mathbf{x}^{(1)} : [k] \to [n]$ defined by $\mathbf{x}^{(1)}(j) = \mathbf{x}(j)$ for $i \neq j$ and $\mathbf{x}^{(1)}(\ell) = \mathbf{x}^{(1)}(\ell')$. Since $\mathcal{G}$ is bidirectional, we see that $\xi^{(1)}$ is a homomorphism $F \to \mathcal{G}$. Also note that $\mathbf{x} \sim \mathbf{x}^{(1)}$ and $\mathbf{x}^{(1)}$ uses at most $k-1$ distinct values in $[n]$. By repeating a similar operation, we can construct a sequence of homomorphisms $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \cdots, \mathbf{x}^{(k-2)} =: \mathbf{y}^{(1)}$ such that $\mathbf{y}$ uses only two distinct values in $[n]$.

Next, let G denote the skeleton of $\mathcal{G}$, which is connected since $\mathcal{G}$ is irreducible and bidirectional. Suppose there exists a walk $W = (a_1, a_2, \cdots, a_{2m})$ in G for some integer $m \geq 0$ such that $\mathbf{y}(1) = a_1$ and $\mathbf{x}'(1) = a_{2m-1}$. We claim that this implies $\mathbf{x}'$ is reachable from $\mathbf{y}^{(1)}$ in $k(m+1) + k - 2$ steps.

To see this, recall that the walk W is chosen in the skeleton G so that at least one of $A(a_2, a_3)$ and $A(a_3, a_2))$ is positive. Hence with positive probability, we can move all nodes in $\mathbf{y}^{(1)}[F]$ at location a_1 in $\mathcal{G}$ to location a_3, and the resulting vertex map $\mathbf{y}^{(2)} : [k] \to \{a_2, a_3\}$ is still a homomorphism $F \to \mathcal{G}$. By a similar argument, we can construct a homomorphism $y^{(3)} : F \to \mathcal{G}$ such that $\mathbf{y}^{(3)}(1) = a_3$ and $y^{(3)}$ maps all nodes of F onto $\{a_3, a_4\}$. Also note that $\mathbf{y}^{(3)}$ is reachable from $y^{(1)}$ in k steps. Hence we can 'slide over' $y^{(1)}$ onto the nodes $\{a_3, a_4\}$ in k steps. Repeating this argument, this shows that there is a homomorphism $\mathbf{y}^{(m)} : F \to \mathcal{G}$ such that $\mathbf{y}^{(m)}$ maps $[k]$ onto $\{a_{2m-1}, a_{2m}\}$ and it is reachable from $\mathbf{y}^{(1)}$ in km steps.

To finish the proof, we first choose a walk $W_1 = (a_1, a_2, \cdots, a_{2m-1})$ in the skeleton G such that $a_1 = \mathbf{y}^{(1)}(1)$ and $a_{2m-1} = \mathbf{x}'(1)$ for some integer $m \geq 1$. We can always choose such a walk using the odd cycle in G, say C, and the connectivity of G: first walk from $\mathbf{y}^{(1)}(1)$ to the odd cycle C, traverse it in one of the two ways, and then walk to $\mathbf{x}'(1)$. Moreover, it is easy to see that this gives $2m - 2 \leq 4\mathrm{diam}(\mathcal{G})$. Lastly, since $\mathbf{x}'$ is a homomorphism $F \to \mathcal{G}$ with $\mathbf{x}'(1) = a_{2m-1}$ and since $k \geq 2$, there must exist some node $a_{2m} \in [m]$ such that $A(a_{2m-1}, a_{2m}) > 0$. Since $\mathcal{G}$ is bidirectional, we also have $A(a_{2m}, a_{2m-1}) > 0$, so a_{2m-1} and a_{2m} are adjacent in the skeleton G. Hence we can let W be the walk $(a_1, a_2, \cdots, a_{2m-1}, a_{2m})$. Then $\mathbf{y}'$ is reachable from $\mathbf{x}$ in $k-2$ steps by construction, and $\mathbf{x}'$ is reachable from $\mathbf{y}^{(1)}$ in $k(m+1) + k - 2 \leq 2k(\mathrm{diam}(\mathcal{G}) + 1) + k - 2$ steps by the claim. Hence $\mathbf{x}'$ is reachable from $\mathbf{x}$ in $2k\mathrm{diam}(\mathcal{G}) + 4(k-1)$ steps, as desired. ■

When F is not necessarily a tree, a straightforward generalization of the argument in the Proof of B.1 shows the following.

Proposition B.2 *Let F be any simple motif and $\mathcal{G}$ be an irreducible and bidirectional network. Suppose there exists an integer $r \geq 1$ with following three conditions:*

(i) *For each $\mathbf{x} \in \mathcal{G}(F, \mathcal{G})$, there exists $\mathbf{y} \in \mathcal{G}(F, \mathcal{G})$ such that $\mathbf{y}$ is reachable from $\mathbf{x}$ in k steps and the skeleton of $\mathbf{y}[F]$ is isomorphic to K_r.*

(ii) *$d_G(u, v) < r$ implies $\{u, v\} \in E_G$.*

(iii) *G contains K_{r+1} as a subgraph.*

Then $\mathcal{S}(F, \mathcal{G})$ is connected and

$$\operatorname{diam}(\mathcal{S}(F, \mathcal{G})) \leq 2k \cdot \operatorname{diam}(\mathcal{G}) + 2(k - r). \tag{150}$$

Proof Omitted. ∎

Next, we prove Theorem 2.1.

Proof of Theorem 2.1.

Proposition B.1 and an elementary Markov chain theory implies that the Glauber chain is irreducible under the assumption of (ii) and has a unique stationary distribution. Hence it remains to show (i), that $\pi := \mathbb{P}_{F \to \mathcal{G}}$ is a stationary distribution of the Glauber chain. To this end, write $F = ([k], A_F)$ and let P be the transition kernel of the Glauber chain. It suffices to check the detailed balance equation is satisfied by π. Namely, let $\mathbf{x}, \mathbf{y}$ be any homomorphisms $F \to \mathcal{G}$ such that they agree at all nodes of F but for some $\ell \in [k]$. We will show that

$$\pi(\mathbf{x})P(\mathbf{x}, \mathbf{y}) = \pi(\mathbf{y})P(\mathbf{y}, \mathbf{x}). \tag{151}$$

Decompose F into two motifs $F_\ell = ([k], A_\ell)$ and $F_\ell^c = ([k], A_\ell^c)$, where $A_\ell(i, j) = A_F(i, j)\mathbf{1}(u \in \{i, j\})$ and $A_\ell^c(i, j) = A_F(i, j)\mathbf{1}(u \notin \{i, j\})$. Note that $A_F = A_\ell + A_\ell^c$. Then we can write

$$\pi(\mathbf{x})P(\mathbf{x}, \mathbf{y}) = \frac{k^{-1}}{\mathtt{t}(F, \mathcal{G})} \left(\prod_{1 \leq i, j \leq k} A(\mathbf{x}(i), \mathbf{x}(j))^{A_\ell^c(i,j)} \right) \left(\prod_{\substack{i \in [k] \\ i \neq \ell}} \alpha(\mathbf{x}(i)) \right) \tag{152}$$

$$\times \frac{\prod_{j \neq \ell} [A(\mathbf{x}(j), \mathbf{x}(\ell))A(\mathbf{x}(j), \mathbf{y}(\ell))]^{A_\ell(j,\ell)} [A(\mathbf{x}(j), \mathbf{x}(\ell))A(\mathbf{y}(\ell), \mathbf{x}(j))]^{A_\ell(\ell,j)}}{\sum_{1 \leq c \leq n} \left(\prod_{j \neq c} A(\mathbf{x}(j), c)^{A_\ell(j,\ell)} A(c, \mathbf{x}(j))^{A_\ell(\ell,j)} \right) A(c, c)^{A_\ell(\ell,\ell)} \alpha(c)} \tag{153}$$

$$\times A(\mathbf{x}(\ell), \mathbf{x}(\ell))^{A_\ell(\ell,\ell)} A(\mathbf{y}(\ell), \mathbf{y}(\ell))^{A_\ell(\ell,\ell)} \alpha(\mathbf{x}(\ell))\alpha(\mathbf{y}(\ell)). \tag{154}$$

From this and the fact that $\mathbf{x}$ and $\mathbf{y}$ agree on all nodes $j \neq \ell$ in $[k]$, we see that the value of $\pi(\mathbf{x})P(\mathbf{x}, \mathbf{y})$ is left unchanged if we exchange the roles of $\mathbf{x}$ and $\mathbf{y}$. This shows (151), as

desired. ■

To prove Theorem 2.4, we first recall a canonical construction of coupling (X, Y) between two distributions μ and ν on a finite set Ω such that $\mu(x) \wedge \nu(x) > 0$ for some $x \in \Omega$. Let $p = \sum_{x\in\Omega} \mu(x) \wedge \nu(x) \in (0, 1)$. Flip a coin with the probability of heads equal to p. If it lands heads, draw Z from the distribution $p^{-1}\mu \wedge \nu$ and let $X = Y = Z$. Otherwise, draw independently X and Y from the distributions $(1-p)^{-1}(\mu-\nu)\mathbf{1}(\mu > \nu)$ and $(1-p)^{-1}(\nu-\mu)\mathbf{1}(\nu > \mu)$, respectively. It is easy to verify that X and Y have distributions μ and ν, respectively, and that $X = Y$ if and only if the coin lands heads. This coupling is called the *optimal coupling* between μ and ν, since

$$\mathbb{P}(X \neq Y) = 1 - p = \|\mu - \nu\|_{\mathrm{TV}}. \tag{155}$$

The following lemma is a crucial ingredient for the proof of Theorem 2.4.

Lemma B.2 *Fix a network $\mathcal{G} = ([n], A, \alpha)$ and a simple motif $F = ([k], A_F)$. Let $(\mathbf{x}_t)_{t\geq 0}$ and $(\mathbf{x}'_t)_{t\geq 0}$ be the Glauber chains of homomorphisms $F \to G$ such that $\mathbf{x}_0$ is reachable from $\mathbf{x}'_0$. Then there exists a coupling between the two chains such that*

$$\mathbb{E}[d_{F,\mathcal{G}}(\mathbf{x}_t, \mathbf{x}'_t)] \leq \exp\left(-\frac{c(\Delta, \mathcal{G})t}{k}\right) d_{F,\mathcal{G}}(\mathbf{x}_0, \mathbf{x}'_0), \tag{156}$$

where $\Delta = \Delta(F)$ denotes the maximum degree of F defined at (7).

Proof Denote $\rho(t) = d_{F,\mathcal{G}}(\mathbf{x}_t, \mathbf{x}'_t)$ for all $t \geq 0$. Let P denote the transition kernel of the Glauber chain. We first claim that if $\mathbf{x}_t \sim \mathbf{x}_{t'}$, then there exists a coupling between $\mathbf{x}_{t+1}$ and $\mathbf{x}'_{t+1}$ such that

$$\mathbb{E}[\rho(t+1) \,|\, \rho(t) = 1] = 1 - \frac{c(\Delta, \mathcal{G})}{k}. \tag{157}$$

Suppose $\mathbf{x}_t$ and $\mathbf{x}'_t$ differ at a single coordinate, say $u \in [k]$. Denote $N_F(u) = \{i \in [k] \,|\, A_F(i, u) + A_F(u, i) > 0\}$. To couple $\mathbf{x}_t$ and $\mathbf{x}'_{t+1}$, first sample $v \in [k]$ uniformly at random. Let $\mu = \mu_{\mathbf{x}_t, v}$ and $\mu' = \mu_{\mathbf{x}'_t, v}$. Note that $\mu = \mu'$ if $v \notin N_F(u)$. If $v \in N_F(u)$, then since $b := \mathbf{x}_t(v) = \mathbf{x}'_t(v)$ and $\mathbf{x}_t, \mathbf{x}'_t$ are homomorphisms $F \to \mathcal{G}$, we have $\mu(b) \wedge \mu'(b) > 0$. Hence the optimal coupling (X, Y) between μ and μ' are well-defined. We then let $\mathbf{x}_{t+1}(v) = X$ and $\mathbf{x}'_{t+1}(v) = Y$.

Note that if $v \notin N_F(u) \cup \{u\}$, then $X = Y$ with probability 1 and $\rho(t+1) = 1$. If $v = u$, then also $X = Y$ with probability 1 and $\rho(t+1) = 0$. Otherwise, $v \in N_F(u)$ and noting that (155), either $X = Y$ with probability $1 - \|\mu - \mu'\|_{\mathrm{TV}}$ and $\rho(t+1) = 0$, or $X \neq Y$ with probability $\|\mu - \mu'\|_{\mathrm{TV}}$. In the last case, we have $\rho(t+1) = 2$ or 3 depending on the structure of $\mathcal{G}$. Combining these observations, we have

$$\begin{aligned}
&\mathbb{E}[\rho(t+1) - 1 \,|\, \rho(t) = 1] && (158)\\
&\quad \leq 2\mathbb{P}(\rho(t+1) \in \{2, 3\} \,|\, \rho(t) = 1) - \mathbb{P}(\rho(t+1) = 0 \,|\, \rho(t) = 1) && (159)\\
&\quad \leq -k^{-1}\left(1 - 2\Delta\|\mu - \mu'\|_{\mathrm{TV}}\right). && (160)
\end{aligned}$$

Further, since μ and μ' are determined locally, the expression in the bracket is at most $c(\Delta, \mathcal{G})$. This shows the claim.

To finish the proof, first note that since $\mathbf{x}_0$ and $\mathbf{x}'_0$ belongs to the same component of $\mathcal{S}(F,\mathcal{G})$, so do $\mathbf{x}_t$ and $\mathbf{x}'_t$ for all $t \geq 0$. We may choose a sequence $\mathbf{x}_t = \mathbf{x}_t^{(0)}, \mathbf{x}_t^{(1)}, \cdots, \mathbf{x}_t^{(\rho(t))} = \mathbf{x}'_t$ of homomorphisms $x_t^{(i)} : F \to G$ such that $\mathbf{x}_t^{(i)} \sim \mathbf{x}_t^{(i+1)}$ for all $0 \leq i < m$. Use the similar coupling between each pair $\mathbf{x}_t^{(i)}$ and $\mathbf{x}_t^{(i+1)}$. Then triangle inequality and the claim yields

$$\mathbb{E}[\rho(t+1)] \leq \sum_{i=0}^{\rho(t)} \mathbb{E}[d_H(\mathbf{x}_{t+1}^{(i)}, \mathbf{x}_{t+1}^{(i+1)})] \leq \left(-\frac{c(\Delta,\mathcal{G})}{k}\right)\rho(t), \tag{161}$$

where we denoted by $\mathbf{x}_{t+1}^{(i)}$ the homomorphism obtained after a one-step update of the Glauber chain from $\mathbf{x}_t^{(i)}$. Iterating this observation shows the assertion. ■

Remark B.1 *In the second paragraph in the proof of Lemma B.2, we always have* $\rho(t+1) \in \{0,1,2\}$ *if* $A(x,y) > 0$ *for all* $x \neq y \in [n]$. *In this case, Lemma B.2 holds with* $c(\Delta,\mathcal{G})$ *replaced by* $c'(\Delta,\mathcal{G})$, *which is defined similarly as in (31) without the factor of* 2. ▲

Now Theorem 2.4 follows immediately.

Proof of Theorem 2.4. Let $(\mathbf{x}_t)_{t\geq 0}$ and $(\mathbf{x}'_t)_{t\geq 0}$ be Glauber chains of homomorphisms $F \to \mathcal{G}$. Let P be the transition kernel of the Glauber chain. By Proposition B.1, $\mathbf{x}_0$ is reachable from $\mathbf{x}'_0$ and $d_{F,\mathcal{G}}(\mathbf{x}_0,\mathbf{x}'_0) \leq 2k(\mathrm{diam}(\mathcal{G})+1)$. Using the coupling between $\mathbf{x}_t$ and $\mathbf{x}'_t$ as in Lemma B.2 and Markov's inequality give

$$\mathbb{P}(\mathbf{x}_t \neq \mathbf{x}'_t) = \mathbb{P}(d_{F,\mathcal{G}}(\mathbf{x}_t,\mathbf{x}'_t) \geq 1) \leq \mathbb{E}(d_{F,\mathcal{G}}(\mathbf{x}_t,\mathbf{x}'_t)) \leq 2k\exp\left(-\frac{c(\Delta,\mathcal{G})t}{k}\right)(\mathrm{diam}(\mathcal{G})+1). \tag{162}$$

Minimizing the left hand side overall coupling between $P^t(\mathbf{x}_0,\cdot)$ and $P^t(\mathbf{x}'_0,\cdot)$ gives

$$\|P^t(\mathbf{x}_0,\cdot) - P^t(\mathbf{x}'_0,\cdot)\|_{TV} \leq 2k\exp\left(-\frac{c(\Delta,\mathcal{G})t}{k}\right)(\mathrm{diam}(\mathcal{G})+1). \tag{163}$$

Then the assertion follows. ■

Remark B.2 *Suppose that* $\mathcal{G}$ *is the complete graph* K_q *with* q *nodes and uniform distribution on its nodes. Then a homomorphism* $F \to K_q$ *is a* q*-coloring of* F *and it is well-known that the Glauber chain of* q*-colorings of* F *mixes rapidly with mixing time*

$$t_{mix}(\varepsilon) \leq \left\lceil \left(\frac{q-2\Delta}{q-\Delta}\right) k\log(\varepsilon/k) \right\rceil, \tag{164}$$

provided $q > 2\Delta$ *(e.g., (Levin and Peres, 2017, Thm. 14.8)). This can be obtained as a special case of Lemma B.2. Indeed, note that* $\mathcal{S}(F,K_q)$ *is connected and has a diameter at most* k. *Hence according to Lemma B.2 and Remark B.1, it is enough to show that*

$$c'(\Delta,K_q) \geq \frac{q-2\Delta}{q-\Delta}, \tag{165}$$

where the quantity on the left-hand side is defined in Remark B.1. To see this, note that when $\mathcal{G}$ is a simple graph with uniform distribution on its nodes,

$$1 - \|\mu_{\mathbf{x},v} - \mu_{\mathbf{x}',v}\|_{TV} = \sum_{z\in[n]} \mu_{\mathbf{x},v}(z) \wedge \mu_{\mathbf{x}',v}(z) = \frac{|supp(\mu_{\mathbf{x},v}) \cap supp(\mu_{\mathbf{x}',v})|}{|supp(\mu_{\mathbf{x},v})| \vee |supp(\mu_{\mathbf{x}',v})|}. \tag{166}$$

When we take $\mathcal{G} = K_q$, it is not hard to see that the last expression in (166) is at most $1 - 1/(q-\Delta)$. Hence we have (165), as desired. ▲

Appendix C. Proof of Stability inequalities

In this section, we provide proofs of the stability inequalities stated in Subsection **??**, namely, Propositions 4.1, 4.2, and 4.1.

Proof of Proposition 4.1. First, write

$$|\mathtt{t}(H,U\,|\,F) - \mathtt{t}(H,W\,|\,F)| \le \frac{\mathtt{t}(H,U)|\mathtt{t}(F,W) - \mathtt{t}(F,U)| + \mathtt{t}(F,U)|\mathtt{t}(H,U) - \mathtt{t}(H,W)|}{\mathtt{t}(F,U)\mathtt{t}(F,W)}. \tag{167}$$

Since F is a subgraph of H, we have $\mathtt{t}(H,U) \le \mathtt{t}(F,U)$ and $|E_H| \le |E_H|$. Hence the assertion follows by (5). ■

In order to prove Proposition 4.2, note that the norm of a kernel $W : [0,1]^2 \to [0,\infty)$ can also defined by the formula

$$\|W\|_\square = \sup_{0\le f,g\le 1} \left| \int_0^1 \int_0^1 W(x,y) f(x) g(y)\, dx\, dy \right|, \tag{168}$$

where $f, g : [0,1] \to [0,1]$ are measurable functions.

Proof of Proposition 4.2. Let $F = ([k], A_F)$ be a simple motif and U, W denote graphons. Write $\bar{U} = \mathtt{t}(F,U)U^F$ and $\bar{W} = \mathtt{t}(F,W)W^F$. We first claim that

$$\|\bar{U} - \bar{W}\|_\square \le \|A_F\|_1 \cdot \|U - W\|_\square, \tag{169}$$

from which the assertion follows easily. Indeed,

$$\begin{aligned} \|U^F - W^F\|_\square &= \frac{1}{\mathtt{t}(F,U)\mathtt{t}(F,W)} \|\mathtt{t}(F,W)\bar{U} - \mathtt{t}(F,W)\bar{W}\|_\square \\ &\le \frac{1}{\mathtt{t}(F,U)\mathtt{t}(F,W)} \left(\mathtt{t}(F,W) \cdot \|\bar{U} - \bar{W}\|_\square + |\mathtt{t}(F,U) - \mathtt{t}(F,W)| \cdot \|\bar{W}\|_\square\right), \end{aligned}$$

and we have $\|\bar{W}\|_\square / \mathtt{t}(F,U) = \|W^F\|_\square = \|W^F\|_1 = 1$. Then the assertion follows from (62) and a similar inequality after changing the role of U and W.

To show the claim, let $f, g : [0,1] \to [0,1]$ be two measurable functions. It suffices to show that

$$\left| \int_0^1 \int_0^1 f(x_1) g(x_n) (\bar{U}(x_1, x_n) - \bar{W}(x_1, x_n))\, dx_1 dx_n \right| \leq \|A_F\|_1 \cdot \|U - W\|_{\square}. \tag{170}$$

Indeed, the double integral on the left-hand side can be written as

$$\int_{[0,1]^n} f(x_1) g(x_n) \left(\prod_{1 \leq i,j \leq k} U(z_i, w_j)^{A_F(i,j)} - \prod_{1 \leq i,j \leq k} W(z_i, w_j)^{A_F(i,j)} \right) dx_1 \cdots dx_n. \tag{171}$$

We say a pair $(i,j) \in [k]^2$ a 'directed edge' of F if $A_F(i,j) = 1$. Order all directed edges of F as $E = \{e_1, e_2, \cdots, e_m\}$, and denote $e_r = (i_r, j_r)$. Since F is a simple motif, there is at most one directed edge between each pair of nodes. Hence we can write the term in the parenthesis as the following telescoping sum

$$\begin{aligned} &\sum_{r=1}^{m} U(e_1) \cdots U(e_{r-1})(U(e_r) - W(e_r)) W(e_{r+1}) \cdots W(e_m) \\ &\qquad = \sum_{r=1}^{m} \alpha(z_{i_r}) \beta(w_{j_r}) (U(z_{i_r}, w_{j_r}) - W(z_{i_r}, w_{j_r})), \end{aligned}$$

where $\alpha(z_{i_r})$ is the product of all $U(e_k)$'s and $W(e_k)$'s such that e_k uses the node i_r and $\beta(w_{j_r})$ is defined similarly. Now for each $1 \leq r \leq m$, we have

$$\left| \int_{[0,1]^n} f(x_1) g(x_n) \alpha(z_{i_r}) \beta(w_{j_r}) (U(z_{i_r}, w_{j_r}) - W(z_{i_r}, w_{j_r}))\, dx_1 \cdots dx_n \right| \leq \|U - W\|_{\square}. \tag{172}$$

The claim then follows. ∎

Lastly, we prove Theorem 4.1. It will be convenient to introduce the following notion of distance between filtrations of kernels.

$$d_{\blacksquare}(U, W) = \int_0^{\infty} \|\mathbf{1}(U \geq t) - \mathbf{1}(W \geq t)\|_{\square}\, dt \tag{173}$$

For its 'unlabeled' version, we define

$$\delta_{\blacksquare}(U, W) = \inf_{\varphi} d_{\blacksquare}(U, W^{\varphi}) \tag{174}$$

where the infimum ranges over all measure-preserving maps $\varphi : [0,1] \to [0,1]$.

An interesting observation is that this new notion of distance between kernels interpolates the distances induced by the cut norm and the 1-norm. For a given graphon $U : [0,1]^2 \to [0,1]$ and $t \geq 0$, we denote by $U_{\geq t}$ the 0-1 graphon defined by

$$U_{\geq t}(x, y) = \mathbf{1}(U(x, y) \geq t). \tag{175}$$

Proposition C.1 *For any two graphons $U, W : [0,1]^2 \to [0,1]$, we have*

$$\delta_\square(U,W) \le \delta_\blacksquare(U,W) \le \delta_1(U,W). \tag{176}$$

Proof It suffices to show the following 'labeled' version of the assertion:

$$d_\square(U,W) \le d_\blacksquare(U,W) \le d_1(U,W). \tag{177}$$

To show the first inequality, note that for any fixed $(x,y) \in [0,1]^2$,

$$\int_0^1 \mathbf{1}(U(x,y) \ge t) - \mathbf{1}(W(x,y) \ge t)\, dt = W(x,y) - U(x,y). \tag{178}$$

Hence the first inequality follows easily from the definition and Fubini's theorem:

$$\begin{aligned}
\|U-W\|_\square &= \sup_{S\times T\subseteq[0,1]^2} \left| \int_S \int_T U(x,y) - W(x,y)\, dx\, dy \right| && (179)\\
&= \sup_{S\times T\subseteq[0,1]^2} \left| \int_S \int_T \int_0^1 \mathbf{1}(U>t) - \mathbf{1}(W>t)\, dt\, dx\, dy \right| && (180)\\
&= \sup_{S\times T\subseteq[0,1]^2} \left| \int_0^1 \int_S \int_T \mathbf{1}(U>t) - \mathbf{1}(W>t)\, dx\, dy\, dt \right| && (181)\\
&\le \int_0^1 \sup_{S\times T\subseteq[0,1]^2} \left| \int_S \int_T \mathbf{1}(U>t) - \mathbf{1}(W>t)\, dx\, dy \right| dt. && (182)
\end{aligned}$$

For the second inequality, by a standard approximation argument, it is enough to show the assertion for the special case when both U and W are simple functions. Hence we may assume that there exists a partition $[0,1]^2 = R_1 \sqcup \cdots \sqcup R_n$ into measurable subsets such that both kernels are constant on each R_j. Define kernels $U^0, \cdots, U^n$ by

$$U^j(x,y) = U(x,y)\mathbf{1}\{(x,y) \in R_1 \cup \cdots \cup R_j\} + W(x,y)\mathbf{1}\{(x,y) \in R_{j+1} \cup \cdots \cup R_n\}.$$

In words, U^j uses values from U on the first j R_i's, but agrees with W on the rest. Denote by u_j and w_j the values of U and W on the R_j, respectively. Observe that

$$\left|\mathbf{1}\{U^j(x,y) > t\} - \mathbf{1}\{U^{j-1}(x,y) > t\}\right| = \begin{cases} 1 & \text{if } t \in [u_j \wedge w_j, u_j \vee w_j] \text{ and } (x,y) \in R_j \\ 0 & \text{otherwise.} \end{cases}$$

This yields that, for any $p \in [0,\infty)$

$$\|U^j_{\ge t} - U^{j-1}_{\ge t}\|_\square = \mu(R_j)\mathbf{1}\{t \in [u_j \wedge w_j, u_j \vee w_j]\}.$$

Now triangle inequality for the cut norm gives

$$\begin{aligned}
\int_0^1 \|U_{\ge t} - W_{\ge t}\|_\square\, dt &\le \sum_{j=1}^n |u_j - w_j| \mu(R_j) \\
&= \int_0^1 \int_0^1 \sum_{j=1}^{k^2} \left|U^j(x,y) - U^{j-1}(x,y)\right| \, dx\, dy \\
&= \int_0^1 \int_0^1 |U(x,y) - W(x,y)|\, dx\, dy \\
&= \|U-W\|_1.
\end{aligned}$$

This shows the assertion. ∎

We need one more preparation to prove Theorem 4.1. Let $F = ([k], A_F)$ and $H = ([k], A_H)$ be motifs and $U : [0,1]^2 \to [0,1]$ be a graphon. For each $t \geq 0$, denote

$$\mathtt{t}(H, U_{\geq t}\,;\, F) = \int_{[0,1]^k} \prod_{1 \leq i,j \leq k} \mathbf{1}(U(x_i, x_j)^{A_H(i,j)} \geq t) \prod_{1 \leq i,j \leq k} U(x_i, x_j)^{A_F(i,j)}\, dx_1 \cdots dx_k. \tag{183}$$

Then it is easy to see that

$$\mathtt{f}(H, U \,|\, F)(t) = \frac{1}{\mathtt{t}(F, U)} \mathtt{t}(H, U_{\geq t}\,;\, F). \tag{184}$$

Proposition C.2 *Let $H = ([k], A_H)$ and $F = ([k], A_F)$ be simple motifs such that $H + F = ([k], A_F + A_H)$ is simple. Fix graphons $U, W : [0,1]^2 \to [0,1]$. Then*

$$|\mathtt{t}(H, U_{\geq t}; F) - \mathtt{t}(H, W_{\geq t}; F)| \leq \|A_F\|_1 \cdot \delta_\square(U, W) + \|A_H\|_1 \cdot \delta_\square(U_{\geq t}, W_{\geq t}). \tag{185}$$

Proof Denote $E_F = \{(i,j) \in [k]^2 \,|\, A_F(i,j) > 0\}$ and $E_H = \{(i,j) \in [k]^2 \,|\, A_H(i,j) > 0\}$. Then by the hypothesis, E_F and E_H are disjoint and $E := E_F \cup E_F = \{(i,j) \in [k]^2 \,|\, A_F(i,j) + A_H(i,j) > 0\}$. Write $E = \{e_1, e_2, \cdots, e_m\}$, where $m = |E|$.

Fix a vertex map $[k] \mapsto [0,1]$, $i \mapsto x_i$. For each $1 \leq \ell \leq m$, define a_ℓ and b_ℓ by

$$a_\ell = \begin{cases} U(e_\ell) & \text{if } e_\ell \in E_F \\ \mathbf{1}(U(e_\ell \geq t)) & \text{if } e_\ell \in E_H, \end{cases} \qquad b_\ell = \begin{cases} W(e_\ell) & \text{if } e_\ell \in E_F \\ \mathbf{1}(W(e_\ell \geq t)) & \text{if } e_\ell \in E_H. \end{cases} \tag{186}$$

Then we have

$$\prod_{(i,j) \in E_H} \mathbf{1}(U(x_i, x_j) \geq t) \prod_{(i,j) \in E_F} U(x_i, x_j) = \prod_{\ell=1}^m a_\ell, \tag{187}$$

$$\prod_{(i,j) \in E_H \setminus E_F} \mathbf{1}(W(x_i, x_j) \geq t) \prod_{(i,j) \in E_F} W(x_i, x_j) = \prod_{\ell=1}^m b_\ell. \tag{188}$$

Also, we can write the difference between the integrands as the following telescoping sum

$$\prod_{\ell=1}^m a_\ell - \prod_{\ell=1}^m b_\ell = \sum_{\ell=1}^m (a_1 \cdots a_\ell b_{\ell+1} \cdots b_m - a_1 \cdots a_{\ell-1} b_\ell \cdots b_m) \tag{189}$$

$$= \sum_{\ell=1}^m c_\ell (a_\ell - b_\ell), \tag{190}$$

where each c_ℓ is a suitable product of a_i's and b_i's. Note that since U and W are graphons, each $c_\ell \in [0,1]$. Moreover, say the mth summand corresponds to an edge $(i,j) \in E_H$. By the assumption on simple motifs, none of a_ℓ and b_ℓ depend on both coordinates x_i and x_j

except a_ℓ and b_ℓ. Hence c_ℓ can be written as the product $f_\ell(x_i)g_\ell(x_j)$ of two functions. Furthermore,

$$a_\ell - b_\ell = \begin{cases} U(x_i,x_j) - W(x_i,x_j) & \text{if } (i,j) \in E_F \\ \mathbf{1}(U(x_i,x_j) \geq t) - \mathbf{1}(W(x_i,x_j) \geq t) & \text{if } (i,j) \in E_H. \end{cases} \tag{191}$$

Hence if $(i,j) \in E_F$, we get

$$\left| \int_{[0,1]^k} c_\ell(a_\ell - b_\ell)\, dx_1 \cdots dx_k \right| \tag{192}$$

$$= \left| \int_{[0,1]^{k-2}} \left(\int_{[0,1]^2} f_\ell(x_i)g_\ell(x_j)U(x_i,x_j) - W(x_i,x_j)\, dx_i dx_j \right) \prod_{\ell \neq i,j} dx_\ell \right| \tag{193}$$

$$\leq \|U - W\|_\square. \tag{194}$$

Similarly, for $(i,j) \in E_H$, we have

$$\left| \int_{[0,1]^k} c_\ell(a_\ell - b_\ell)\, dx_1 \cdots dx_k \right| \leq \|U_{\geq t} - W_{\geq t}\|_\square. \tag{195}$$

Therefore the assertion follows from a triangle inequality and optimizing the bound over all measure-preserving maps, as well as noting that $|E_F| = \|A_F\|_1$ and $|E_H| = \|A_H\|_1$. ∎

Now we prove Theorem 4.1.

Proof of Theorem 4.1. Let $F = ([k], A_F)$ and $H = ([k], A_H)$ be simple motifs such that $H + F := ([k], A_F + A_H)$ is simple. First, use a triangle inequality to write

$$|\mathtt{f}(H, U \,|\, F)(t) - \mathtt{f}(H, W \,|\, F)(t)| \tag{196}$$

$$\leq \frac{\mathtt{t}(F,U)|\mathtt{t}(H, W_{t\geq t}\,;\, F) - \mathtt{t}(H, U_{t\geq t}\,;\, F)| + \mathtt{t}(H, U_{\geq t}\,;\, F)|\mathtt{t}(F,U) - \mathtt{t}(F,W)|}{\mathtt{t}(F,U)\mathtt{t}(F,W)}. \tag{197}$$

Note that $\mathtt{t}(F+H,U) \leq \mathtt{t}(F,U)$ and for each $t \in [0,1]$ we have $\mathtt{t}(H, U_{\geq t}\,;\, F) \in [0,1]$ by definition. Hence by using Proposition C.2, we get

$$|\mathtt{f}(H, U \,|\, F)(t) - \mathtt{f}(H, W \,|\, F)(t)| \tag{198}$$

$$\leq \frac{\|A_F\|_1 \cdot \delta_\square(U,W) + \|A_H\|_1 \cdot \delta_\square(U_{\geq t}, W_{\geq t})}{\mathtt{t}(F,W)} + \frac{\|A_F\|_1 \cdot \delta_\square(U,W)}{\mathtt{t}(F,U)}. \tag{199}$$

Integrating this inequality over $t \in [0,1]$ and using Proposition C.1 then give

$$\|\mathtt{f}(H, U \,|\, F)(t) - \mathtt{f}(H, W \,|\, F)(t)\|_1 \tag{200}$$

$$\leq |E_F| \cdot \delta_\square(U,W) \left(\frac{1}{\mathtt{t}(F,W)} + \frac{1}{\mathtt{t}(F,U)} \right) + \frac{|E_H \setminus E_F| \cdot \delta_1(U,W)}{\mathtt{t}(F,W)}. \tag{201}$$

We can obtain a similar inequality after we change the roles of U and W. Then the assertion follows optimizing between the two upper bounds. ∎

Appendix D. Network data sets

In Sections 2.2 and 6, we examined the following real-world and synthetic networks:

1. `Caltech`: This connected network, which is part of the Facebook100 data set Traud et al. (2012) (and which was studied previously as part of the Facebook5 data set **?**), has 762 nodes and 16,651 edges. The nodes represent users in the Facebook network of Caltech on one day in fall 2005, and the edges encode Facebook 'friendships' between these accounts.

2. `Simmons`: This connected network, which is part of the Facebook100 data set Traud et al. (2012) (and which was studied previously as part of the Facebook5 data set **?**), has 1,518 nodes and 65,976 edges. The nodes represent users in the Facebook network of Simmons on one day in fall 2005, and the edges encode Facebook 'friendships' between these accounts.

3. `Reed`: This connected network, which is part of the Facebook100 data set Traud et al. (2012) (and which was studied previously as part of the Facebook5 data set **?**), has 962 nodes and 37,624 edges. The nodes represent users in the Facebook network of Reed on one day in fall 2005, and the edges encode Facebook 'friendships' between these accounts.

4. `NYU`: This connected network, which is part of the Facebook100 data set Traud et al. (2012) (and which was studied previously as part of the Facebook5 data set **?**), has 21,679 nodes and 1,431,430 edges. The nodes represent users in the Facebook network of NYU on one day in fall 2005, and the edges encode Facebook 'friendships' between these accounts.

5. `Virginia`: This connected network, which is part of the Facebook100 data set Traud et al. (2012), has 21,325 nodes and 1,396,356 edges. The nodes represent user accounts in the Facebook network of Virginia on one day in fall 2005, and the edges encode Facebook 'friendships' between these accounts.

6. `UCLA`: This connected network, which is part of the Facebook100 data set Traud et al. (2012), has 20,453 nodes and 747,604 edges. The nodes represent user accounts in the Facebook network of UCLA on one day in fall 2005, and the edges encode Facebook 'friendships' between these accounts.

7. `Wisconsin`: This connected network, which is part of the Facebook100 data set Traud et al. (2012), has 23,842 nodes and 835,952 edges. The nodes represent user accounts in the Facebook network of Wisconsin on one day in fall 2005, and the edges encode Facebook 'friendships' between these accounts.

8. `ER`: An Erdős–Rényi (ER) network Erdős and Rényi (1959); Newman (2018b), which we denote by $\mathrm{ER}(n, p)$, is a random-graph model. The parameter n is the number of nodes and the parameter p is the independent, homogeneous probability that each pair of distinct nodes has an edge between them. The network `ER` is an individual graph that we draw from $\mathrm{ER}(5000, 0.01)$.

9. `WS`: A Watts–Strogatz (WS) network, which we denote by $\mathrm{WS}(n, k, p)$, is a random-graph model to study the small-world phenomenon Watts and Strogatz (1998); Newman (2018b). In the version of WS networks that we use, we start with an n-node ring network in which each node is adjacent to its k nearest neighbors. With independent probability p, we then remove and rewire each edge so that it connects a pair of distinct nodes that we choose uniformly at random. The network `WS` is an individual graph that we draw from $\mathrm{WS}(5000, 50, 0.10)$.

10. `BA`: A Barabási–Albert (BA) network, which we denote by $\mathrm{BA}(n, n_0)$, is a random-graph model with a linear preferential-attachment mechanism Barabási and Albert (1999); Newman (2018b). In the version of BA networks that we use, we start with n_0 isolated nodes and we introduce new nodes with n_0 new edges each that attach preferentially (with a probability that is proportional to node degree) to existing nodes until we obtain a network with n nodes. The network `BA` is an individual graph that we draw from $\mathrm{BA}(5000, 50)$.

11. `SBM`: We use stochastic-block-model (SBM) networks in which each block is an ER network Holland et al. (1983). Fix disjoint finite sets $C_1 \cup \cdots \cup C_{k_0}$ and a $k_0 \times k_0$ matrix B whose entries are real numbers between 0 and 1. An SBM network, which we denote by $\mathrm{SBM}(C_1, \ldots, C_{k_0}, B)$, has the node set $V = C_1 \cup \cdots \cup C_{k_0}$. For each node pair (x, y), there is an edge between x and y with independent probabilities $B[i_0, j_0]$, with indices $i_0, j_0 \in \{1, \ldots, k_0\}$ such that $x \in C_{i_0}$ and $y \in C_{j_0}$. If $k_0 = 1$ and B has a constant p in all entries, this SBM specializes to the Erdős–Rényi (ER) random-graph model $\mathrm{ER}(n, p)$ with $n = |C_1|$. The networks `SBM` is an individual graphs that we draw from $\mathrm{SBM}(C_1, \ldots, C_{k_0}, B)$ with $|C_1| = |C_2| = |C_3| = 1{,}000$, where B is the 3×3 matrix whose diagonal entries are 0.5 and whose off-diagonal entries are 0.001. It has 3,000 nodes and 752,450 edges.

Appendix E. Additional figures and tables

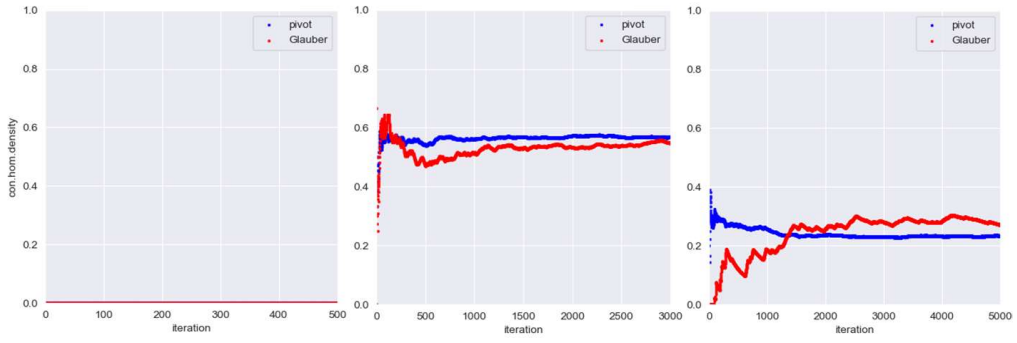

Figure 25: Computing $\mathtt{t}(H_{k,0}, \mathcal{G}_n \mid F_{k,0})$ by time averages of Glauber (red) and Pivot (blue) chains $F_{k,0} \to \mathcal{G}_{50}$ for $k = 0$ (left), $k = 3$ (middle), and $k = 9$ (right).

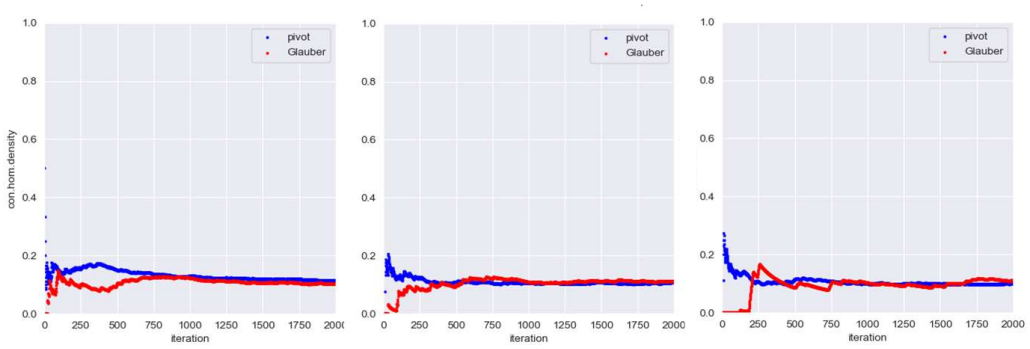

Figure 26: Computing $\mathtt{t}(H_{k,0}, \mathcal{G}_n \,|\, F_{k,0})$ by time averages of Glauber (red) and Pivot (blue) chains $F_{k,0} \to \mathcal{G}_{50}^{0.1,0}$ for $k = 2$ (left), $k = 3$ (middle), and (right) $k = 9$ (right).

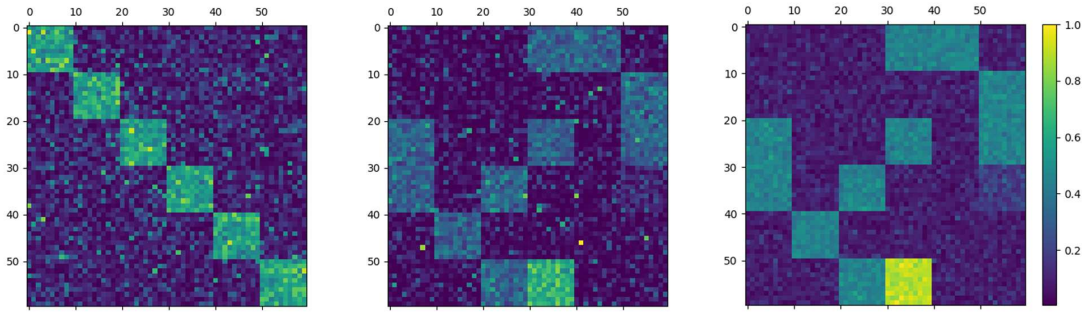

Figure 27: Plots of random block matrices B_1 (left), B_2 (middle), and B_3 (right). Colors from dark blue to yellow denote values of entries from 0 to 1, as shown in the color bar on the right.

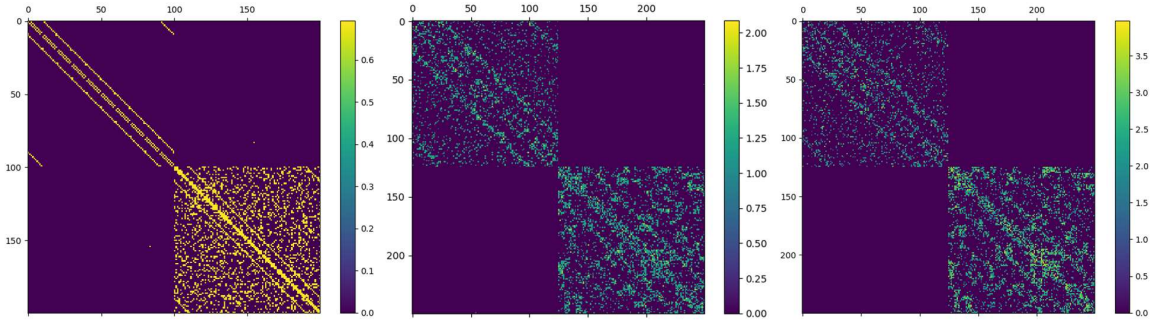

Figure 28: Plot of log transforms of the edge weight matrices A_1 (left), A_2 (middle), and $A_3 = A_2^{C_3}$ (right). Corresponding color bars are shown to the right of each plot.

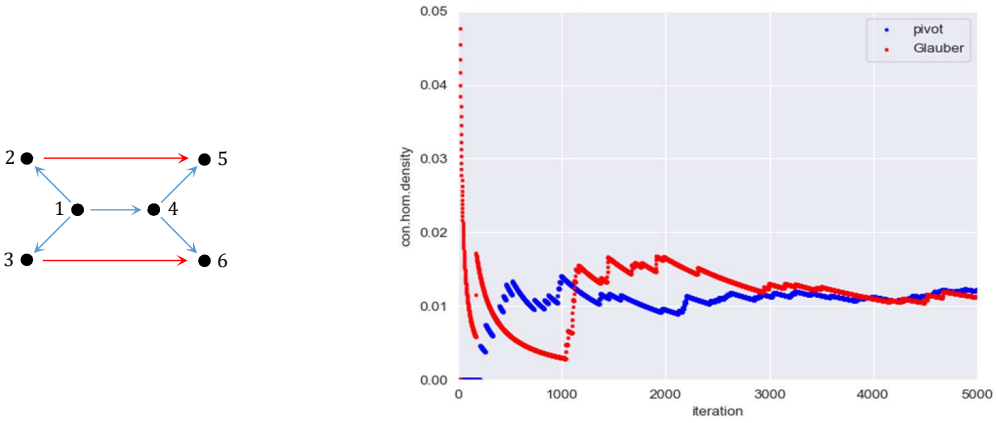

Figure 29: Computing $\mathtt{t}(H, \mathcal{G}_n \,|\, F)$ via time averages of Glauber/Pivot chains $F \to \mathcal{G}_{50}^{0.1,0}$. The underlying rooted tree motif $F = ([6], \mathbf{1}_{\{(1,2),(1,3),(1,4),(4,5),(4,6)\}})$ is depicted on the left, and $H = ([6], A_H)$ is obtained from F by adding directed edges (red) $(2, 5)$ and $(3, 6)$.

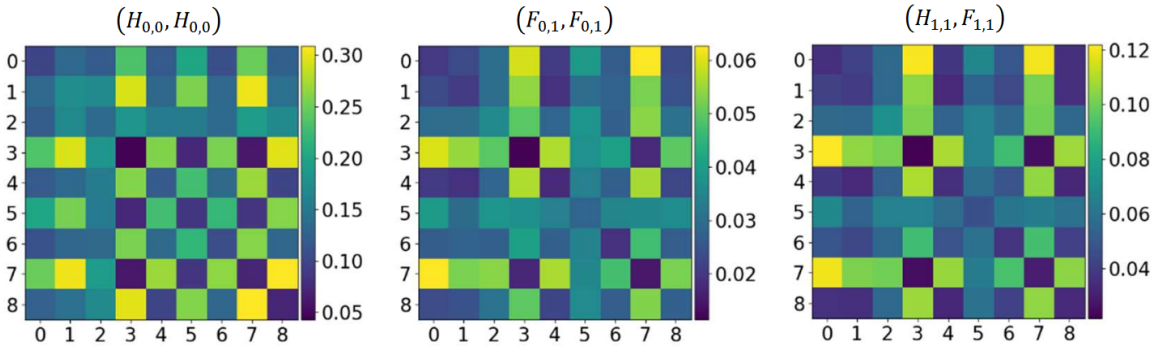

Figure 32: Heat maps of the average L^1-distance matrices between the reference (rows) and validation (columns) CHD profiles of the nine authors for the pair of motifs (H_{00}, F_{00}) (left), (H_{01}, F_{01}) (middle), and (H_{11}, F_{11}) (right).

School	Computation time (sec)	Number of nodes	Average degree
Caltech36	10.32	769	43.32
Reed98	12.78	962	39.11
Simmons81	24.2	1518	43.46
Amherst41	42.96	2235	81.39
Middlebury45	66.72	3075	81.05
Wesleyan43	82.38	3593	76.84
Bucknell39	85.76	3826	83.04
Santa74	87.3	3578	84.82
Rice31	99.9	4087	90.45
Rochester38	122.72	4563	70.74
Princeton12	212.88	6596	88.94
American75	213.68	6386	68.17
UC64	225.82	6833	45.47
Yale4	372.48	8578	94.53
Cal65	570	11247	62.48
GWU54	684.86	12193	77.02
Baylor93	774.84	12803	106.2
Harvard1	809.16	15126	109.03
UCSD34	916.24	14948	59.3
UVA16	1105.14	17196	91.8
BU10	1168.94	19700	64.72
Oklahoma97	1186.76	17425	102.44
Auburn71	1426	18448	105.59
UCLA26	1586.6	20467	73.06
NYU9	1810.44	21679	66.03
UGA50	2088.54	24389	96.28
Wisconsin87	2364.3	23842	70.12
FSU53	2723.78	27737	74.62
Texas84	5419.02	36371	87.47
Penn94	6759.94	41554	65.56

Table 30: Computation times for computing MACCs of the Facebook100 dataset shown in Figure 15 and number of nodes and the average degree of the corresponding networks. Results are shown for 30 networks randomly chosen amongst those in the Facebook100 dataset.

Networks (std of AUC)	*edeg density*	*min degree*	*max degree*	*diameter*	*degree assortativity coef*	*# cliques*	*Avg clustering coeff*	*MACC (k=5)*	*MACC (k=10)*	*MACC (k=15)*	*MACC (k=20)*
Caltech36-Simmons81	0.037	0.029	0.022	0.043	0.068	0.040	0.044	0.040	0.027	0.026	0.020
Caltech36-Reed98	0.035	0.054	0.04	0.055	0.033	0.027	0.042	0.038	0.023	0.021	0.025
Caltech36-NYU9	0.034	0.046	0.035	0.036	0.067	0.039	0.034	0.023	0.021	0.025	0.017
Caltech36-Virginia63	0.035	0.041	0.044	0.033	0.112	0.032	0.059	0.031	0.017	0.021	0.019
Caltech36-UCLA26	0.033	0.032	0.042	0.022	0.031	0.046	0.049	0.021	0.017	0.016	0.022
Caltech36-Wisconsin87	0.039	0.069	0.032	0.045	0.088	0.031	0.027	0.037	0.026	0.024	0.020
Simmons81-Reed98	0.035	0.042	0.049	0.031	0.045	0.029	0.027	0.022	0.022	0.023	0.019
Simmons81-NYU9	0.056	0.067	0.058	0.048	0.052	0.068	0.049	0.036	0.02	0.023	0.017
Simmons81-Virginia63	0.037	0.094	0.052	0.039	0.045	0.060	0.053	0.037	0.043	0.035	0.036
Simmons81-UCLA26	0.036	0.057	0.042	0.042	0.057	0.040	0.042	0.021	0.021	0.020	0.018
Simmons81-Wisconsin87	0.022	0.040	0.029	0.030	0.064	0.043	0.031	0.032	0.020	0.020	0.018
Reed98-NYU9	0.027	0.104	0.051	0.031	0.093	0.018	0.030	0.028	0.029	0.031	0.027
Reed98-Virginia63	0.037	0.042	0.043	0.034	0.064	0.043	0.047	0.028	0.023	0.022	0.020
Reed98-UCLA26	0.063	0.042	0.070	0.032	0.068	0.055	0.042	0.029	0.020	0.030	0.025
Reed98-Wisconsin87	0.042	0.041	0.062	0.031	0.065	0.055	0.046	0.025	0.013	0.012	0.013
NYU9-Virginia63	0.053	0.029	0.071	0.056	0.089	0.075	0.053	0.053	0.035	0.031	0.038
NYU9-UCLA26	0.019	0.060	0.026	0.037	0.106	0.024	0.040	0.034	0.029	0.020	0.026
NYU9-Wisconsin87	0.023	0.047	0.024	0.027	0.036	0.037	0.032	0.031	0.031	0.030	0.026
Virginia63-UCLA26	0.035	0.019	0.040	0.030	0.046	0.028	0.046	0.029	0.019	0.020	0.022
Virginia63-Wisconsin87	0.026	0.032	0.041	0.036	0.046	0.035	0.025	0.027	0.026	0.033	0.024
UCLA26-Wisconsin87	0.044	0.041	0.035	0.027	0.112	0.037	0.047	0.031	0.020	0.026	0.026

Table 31: The standard deviations of AUC scores over ten independent trials of the subgraph classification tasks. See Table 17 for more details.

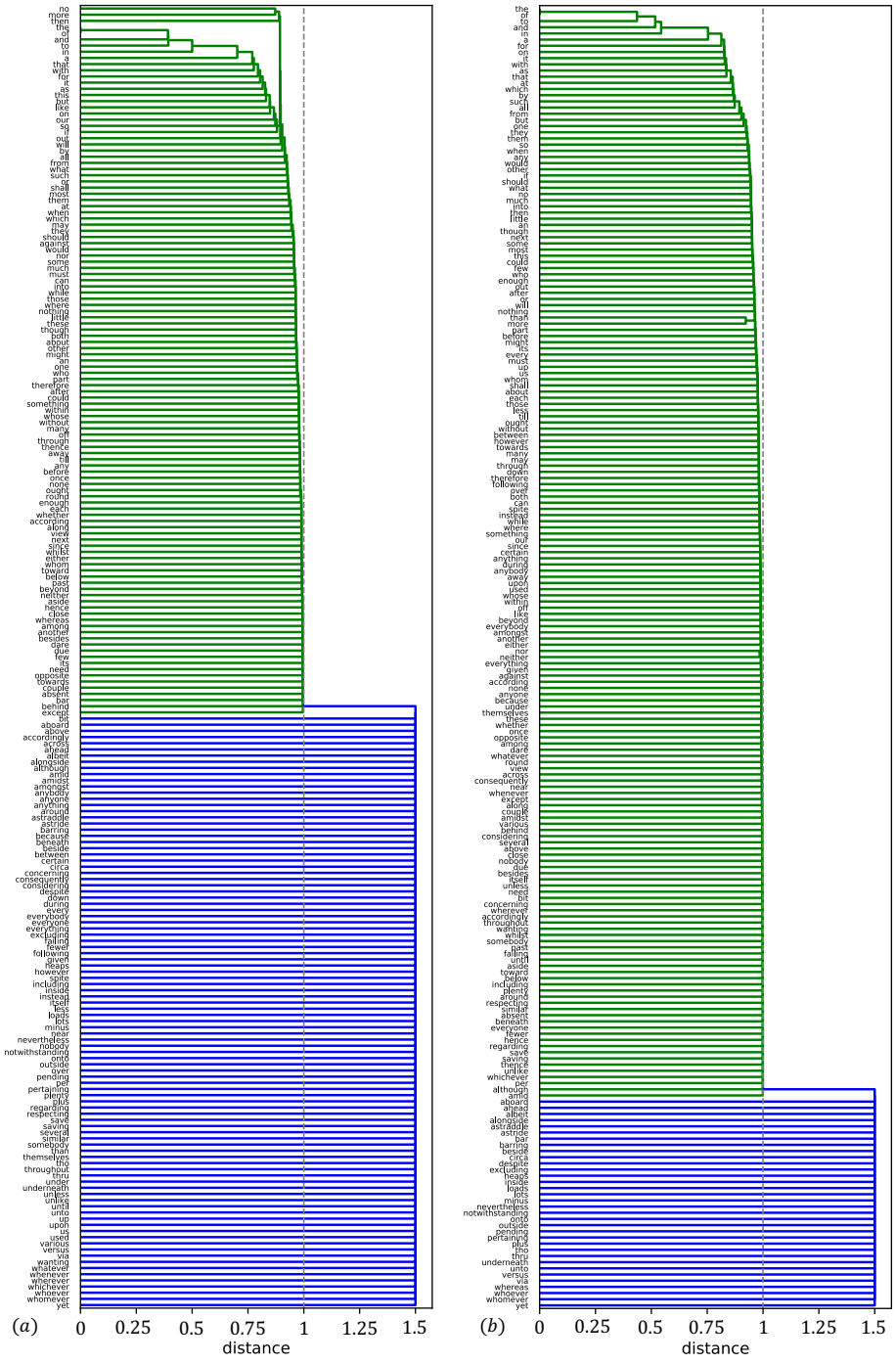

Figure 34: Single-linkage dendrogram of WANs corresponding to "Jane Austen - Pride and Prejudice" (a) and "Shakespeare - Hamlet" (b).

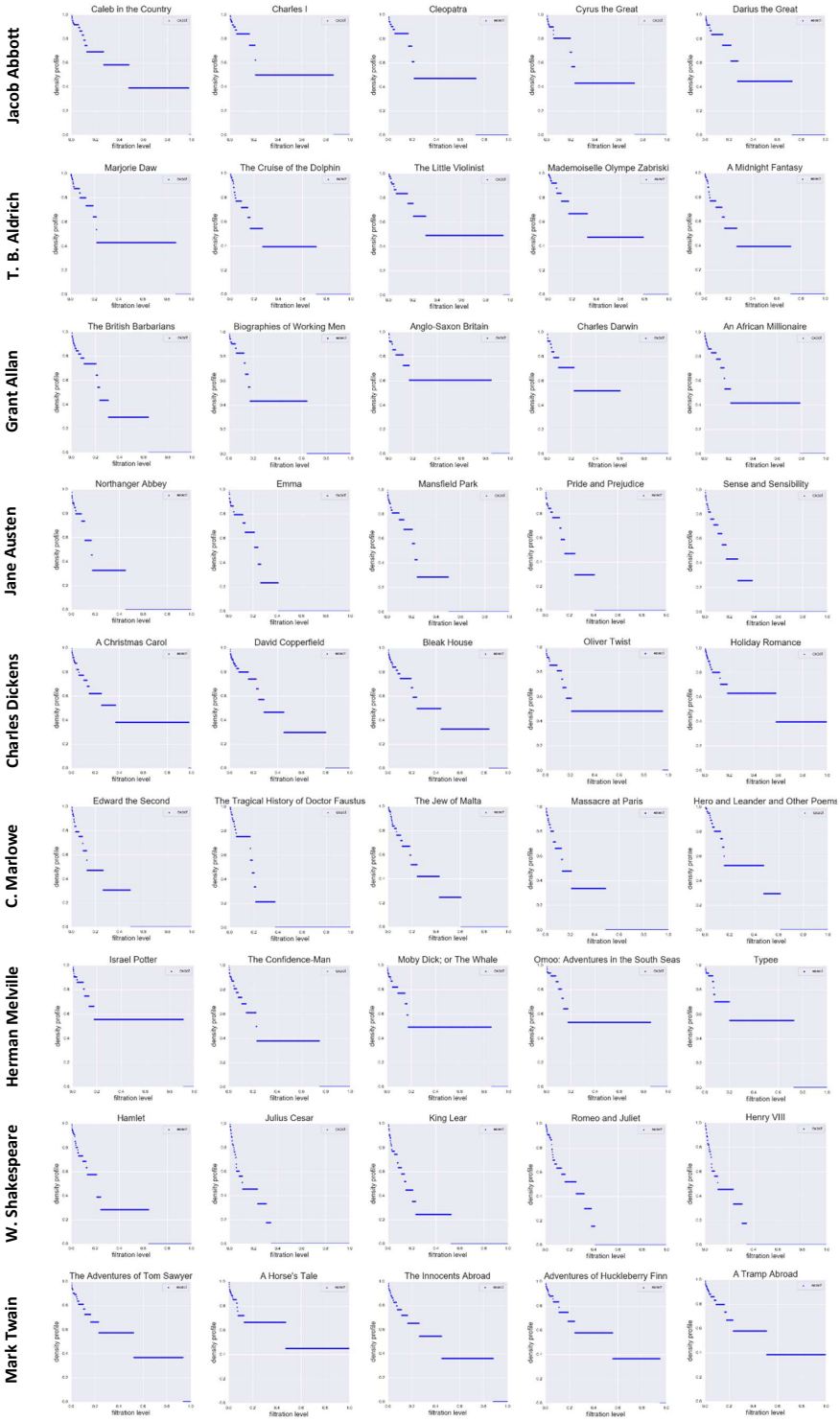

Figure 36: Exact CHD profiles of the WAN dataset for the motif pair $H_{00} = F_{00} = ([1], \mathbf{1}_{\{(1,1)\}})$.

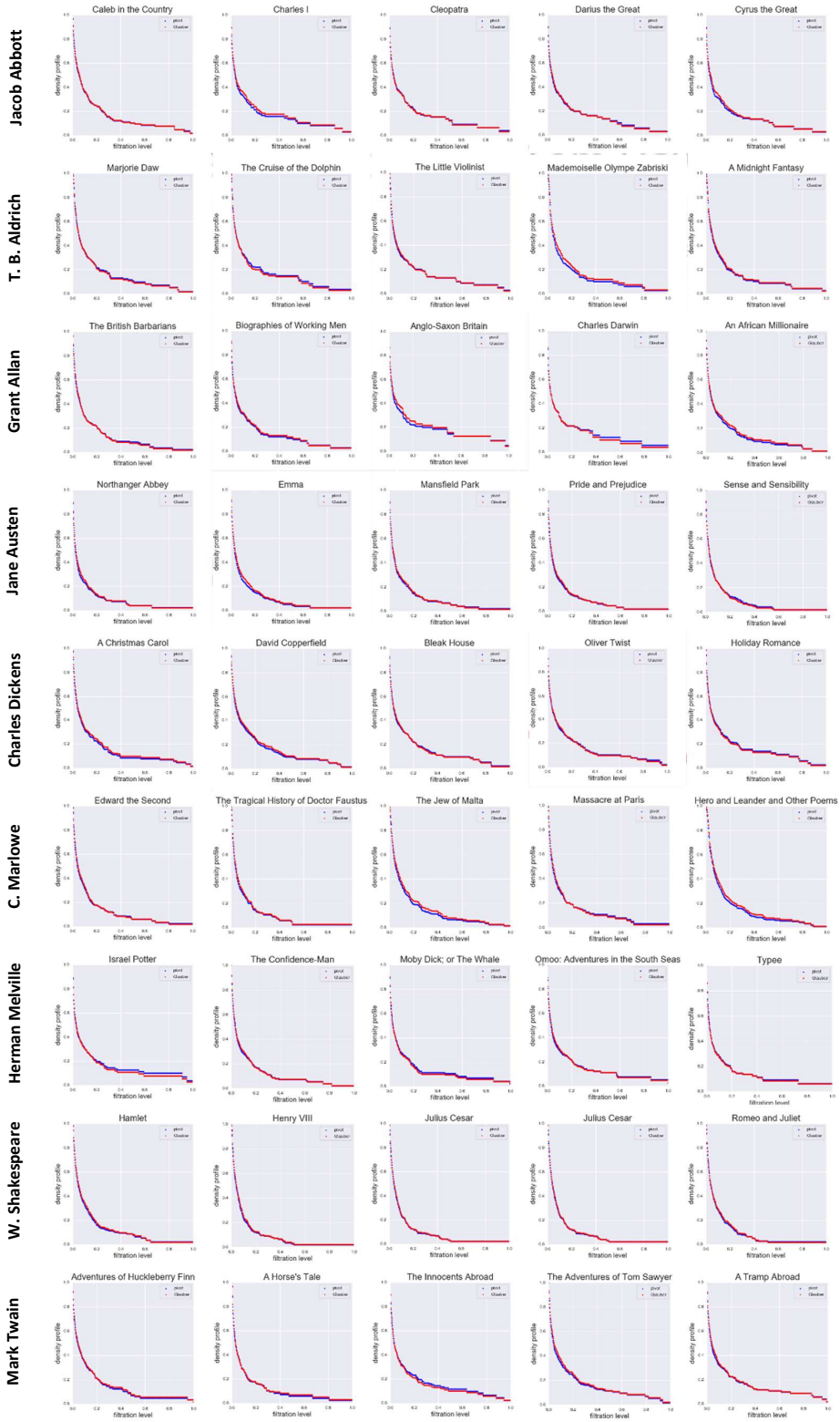

Figure 38: Approximate CHD profiles of the WAN dataset for the motif pair $H_{01} = F_{01} = ([1,2], \mathbf{1}_{\{(1,2)\}})$. Glauber chain (red) and Pivot chain (blue) for 5000 iterations.

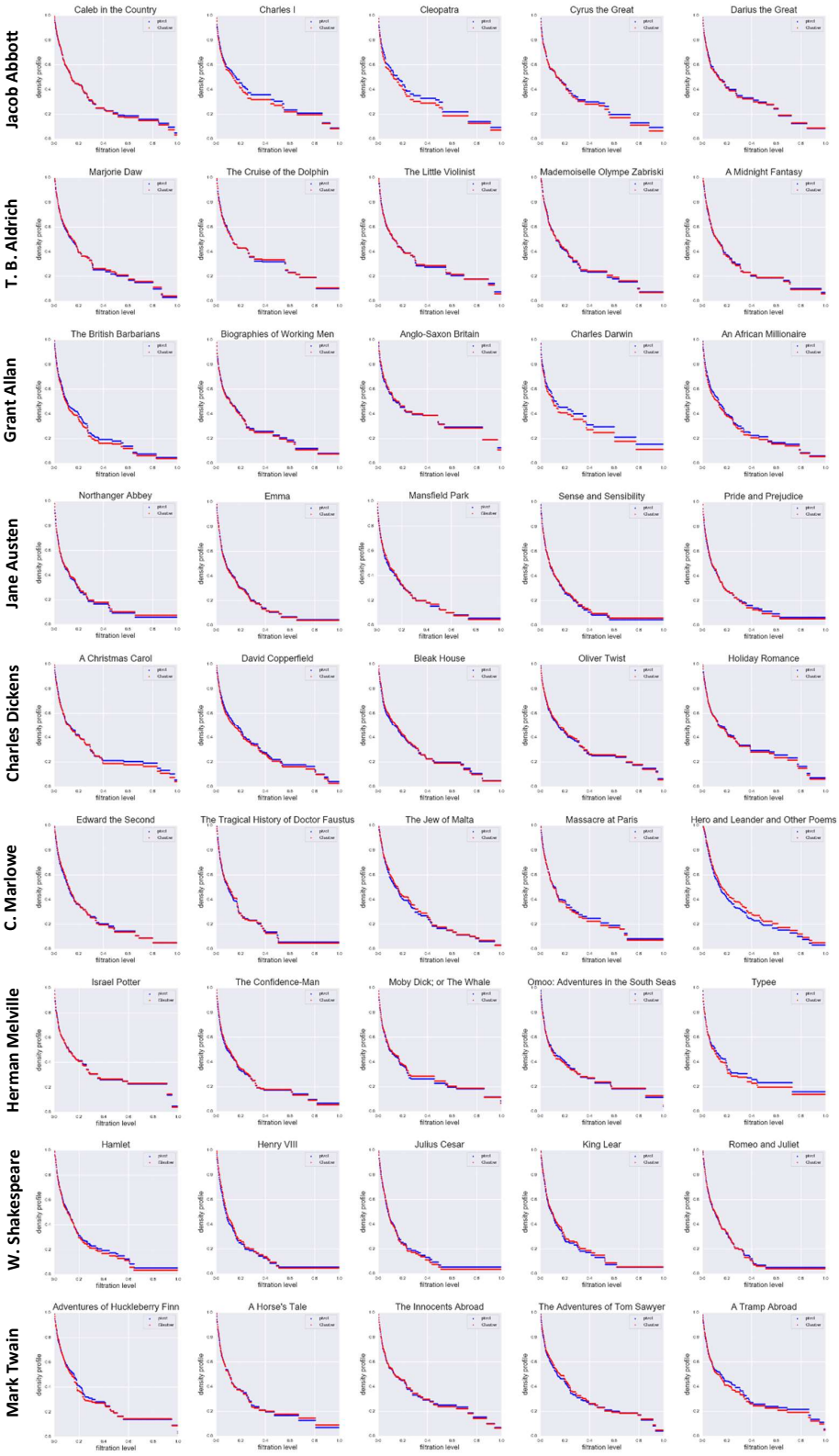

Figure 40: Approximate CHD profiles of the WAN dataset for the motif pair $H_{11} = ([1,2,3], \mathbf{1}_{\{2,3\}})$ and $F_{11} = ([1,2,3], \mathbf{1}_{\{(1,2),(1,3)\}})$. Glauber chain (red) and Pivot chain (blue) for 5000 iterations.