



Research



Cite this article: Peixoto TP. 2025 Uncertainty quantification and posterior sampling for network reconstruction. *Proc. R. Soc. A* **481**: 20250344.
<https://doi.org/10.1098/rspa.2025.0344>

Received: 18 April 2025

Accepted: 22 September 2025

Subject Areas:

statistical physics, complexity

Keywords:

network reconstruction, uncertainty quantification, posterior sampling

Author for correspondence:

Tiago P. Peixoto

e-mail: tiago.peixoto@it-u.at

Uncertainty quantification and posterior sampling for network reconstruction

Tiago P. Peixoto^{1,2}

¹Inverse Complexity Lab, IT:U Interdisciplinary Transformation University, Linz 4040, Austria

²Department of Network and Data Science, Central European University, Vienna 1100, Austria

TPP, 0000-0002-4505-0517

Network reconstruction is the task of inferring the unseen interactions between elements of a system, based only on their behaviour or dynamics. This inverse problem is in general ill-posed and admits many solutions for the same observation. Nevertheless, the vast majority of statistical methods proposed for this task—formulated as the inference of a graphical generative model—can only produce a ‘point estimate’, i.e. a single network considered the most likely. In general, this can give only a limited characterization of the reconstruction, since uncertainties and competing answers cannot be conveyed, even if their probabilities are comparable, while being structurally different. In this work, we present an efficient Markov-chain Monte-Carlo algorithm for sampling from posterior distributions of reconstructed networks, which is able to reveal the full population of answers for a given reconstruction problem, weighted according to their plausibilities. Our algorithm is general, since it does not rely on specific properties of particular generative models, and is specially suited for the inference of large and sparse networks, since in this case an iteration can be performed in time $O(N \log^2 N)$ for a network of N nodes, instead of $O(N^2)$, as would be the case for a more naïve approach. We demonstrate the suitability of our method in providing uncertainties and consensus of solutions (which provably increases the reconstruction accuracy) in a variety of synthetic and empirical cases.

© 2025 The Authors. Published by the Royal Society under the terms of the Creative Commons Attribution License <http://creativecommons.org/licenses/by/4.0/>, which permits unrestricted use, provided the original author and source are credited.

1. Introduction

Many complex systems are governed by interactions that cannot be easily observed directly. For example, while we can use testing to measure individual infections during an epidemic spreading, measuring the direct transmission contacts that caused them is significantly harder [1,2]. Similarly, we can measure the abundance of different species in an ecosystem, or the level of gene expression in a cell, with relatively simple methodologies (e.g. via qPCR DNA amplification or DNA microarrays), but determining directly the interactions between any two species (e.g. mutualism or competition) [3,4] or any two genes [5,6] is significantly more cumbersome. Another prominent example is the human brain, which can have its behaviour harmlessly probed by an fMRI scan, but its direct neuronal structure cannot be measured non-invasively. In all these cases, network reconstruction needs to be performed based on the indirect information available, if we wish to understand how the system functions.

Several different methods have been proposed for the task of network reconstruction. A significant fraction of them are heuristic in nature and attempt to determine the existence of an edge from pairwise correlations of the activities of two nodes [7–12]. These methods are fundamentally limited in two important ways. Firstly, they conflate correlation with conditional dependence or causation, since two nodes may be strongly correlated even if they are not directly connected (e.g. if they share a neighbour in common). Secondly, with these methods, the existence of an edge is decoupled from any explicit modelling of the dynamics or behaviour of the system, which severely hinders the interpretability of the reconstruction—after all, how much would we have really uncovered about a network system if we do not know how an edge contributes to its function? [13]. Another prominent class of methods is based on the definition of explicit generative probabilistic models for the behaviour of a system, conditioned on a network of interactions operating as the parameters of this model [2,14–16]. In this case, the reconstruction amounts to the statistical inference of these parameters from data. Within a Bayesian workflow [17], this inferential approach offers a series of advantages, including: (i) A more principled methodology, coupling tightly theory with data, and relying on explicit—and hence scrutinizable—modelling assumptions; (ii) non-parametric implementations [18] dispense with the need to make *ad hoc* choices, such as arbitrary thresholds, total number of inferred edges, etc.; (iii) the inherent connection with the minimum description length (MDL) principle [19,20] provides a robust framework for model selection [18], according to the combined quality of fit and parsimony of the models considered, such that different hypotheses can be directly compared; and finally, (iv) recent advances [18,21] allow for scalable, sub-quadratic reconstruction of large networks, making the overall approach practical.

However, despite these advantages, so far the literature on network reconstruction deals almost exclusively with point estimates, i.e. most of the methods proposed can only produce a single network, considered to be the most likely one,¹ and do not allow for uncertainty quantification—arguably one of the most desirable and important features of an inferential analysis. In other words, these point estimates contain no information about possible alternatives, how different and plausible they are, and hence how confident we can be about the point estimate in the first place. Besides this limitation that point estimation imposes on interpretability, its accuracy is also in general inferior to estimates that attempt to summarize the consensus over many possible solutions, weighted according to their plausibility [27].

One important reason why point estimation is predominantly employed is its relative algorithmic efficiency, when compared with approaches based on posterior averages. This is the main issue we address in this work, where we develop a scalable algorithm for posterior sampling of reconstructed networks that performs substantially better for larger problem instances than the

¹A notable exception is the literature on the reconstruction of uncertain or incomplete networks, i.e. when the data are a direct measurement of a network, but which has either been corrupted by measurement errors, or parts of it have not been measured at all. For this specific class of reconstruction problems, posterior sampling and uncertainty quantification is more commonplace [22–26]. However, despite both problems sharing the same overall conceptual framework, network reconstruction from dynamics or behaviour is algorithmically very different from the reconstruction of noisy or incomplete networks, and hence requires different computational techniques.

naïve baseline. More specifically, whereas a naïve implementation of a sampling scheme would take time $O(N^2)$ to reconstruct a sparse network of N nodes, our algorithm is capable of doing the same in time $O(N \log^2 N)$.

This paper is organized as follows. In §2, we describe our overall inferential framework and in §3 our posterior sampling approach. In §4, we compare the performance of posterior sampling with point estimates for synthetic examples. In §5, we do the same for empirical data, where we make also a comparison with correlation-based reconstructions. We finalize in §6 with a discussion.

2. Inferential framework

The inferential scenario for network reconstruction consists of some data \mathbf{X} that are assumed to originate from a generative model with a likelihood

$$P(\mathbf{X}|\mathbf{W}), \quad (2.1)$$

where $\mathbf{W} \in \mathbb{R}^{N \times N}$ is a symmetric matrix corresponding to the weights of an undirected graph of N nodes (the alternative scenario for directed networks is straightforward, so we will focus on the undirected case for simplicity). In most cases, we expect \mathbf{W} to be sparse, i.e. its number of non-zero entries scales as $O(N)$, but we do not wish to impose any strict constraints on what values it can take. In many cases, the data are represented by an $N \times M$ matrix of M i.i.d. samples, with X_{im} being a value associated with node i for sample m , such that

$$P(\mathbf{X}|\mathbf{W}) = \prod_{m=1}^M P(\mathbf{x}_m|\mathbf{W}), \quad (2.2)$$

with \mathbf{x}_m being the m th column of \mathbf{X} . Alternatively, we may have that the network generates a Markovian time series with likelihood

$$P(\mathbf{X}|\mathbf{W}) = \prod_{m=1}^M P(\mathbf{x}_m|\mathbf{x}_{m-1}, \mathbf{W}), \quad (2.3)$$

given some initial state \mathbf{x}_0 . Many other possibilities exist, but for our purposes we need only to refer to a generic posterior distribution

$$P(\mathbf{W}|\mathbf{X}) = \frac{P(\mathbf{X}|\mathbf{W})P(\mathbf{W})}{P(\mathbf{X})}, \quad (2.4)$$

which fully quantifies the reconstruction according to some specific generative model. Since the posterior ascribes a probability to every possible reconstructed network \mathbf{W} , it also quantifies the uncertainty of our inference: how sharply or broadly ‘peaked’ a distribution is around the most likely network \mathbf{W}^* means that we should have a correspondingly large or small confidence on its validity as a reconstruction.

Usually, the full posterior distribution is difficult to inspect directly due to its high-dimensional nature. If we are only interested in a particular descriptor $f(\mathbf{W})$ of the reconstructed network, we can avoid this inspection by computing the posterior mean,

$$\bar{f}(\mathbf{X}) = \int f(\mathbf{W})P(\mathbf{W}|\mathbf{X}) d\mathbf{W}, \quad (2.5)$$

or, more completely, the marginal posterior distribution

$$P(y|\mathbf{X}) = \int \delta(y - f(\mathbf{W}))P(\mathbf{W}|\mathbf{X}) d\mathbf{W}, \quad (2.6)$$

which fully quantifies the range of plausible descriptor values.

An alternative task is to summarize the posterior distribution as a whole, via a representative point estimate, and a dispersion around it. There is no unique way to obtain this summary, which will in general depend on a chosen error function $\epsilon(\mathbf{W}, \mathbf{W}')$ that we use to evaluate how close is

a reconstructed network \mathbf{W} from the true network \mathbf{W}' , with $\mathbf{W}' = \arg \min_{\mathbf{W}} \epsilon(\mathbf{W}, \mathbf{W}')$. Since our actual knowledge of the true network is given by the posterior distribution, we need to consider the error averaged over the posterior distribution,

$$\bar{\epsilon}(\mathbf{W}, \mathbf{X}) = \int \epsilon(\mathbf{W}, \mathbf{W}') P(\mathbf{W}' | \mathbf{X}) d\mathbf{W}'. \quad (2.7)$$

The representative reconstruction $\tilde{\mathbf{W}}$ is the one that minimizes this average error,

$$\tilde{\mathbf{W}}(\mathbf{X}) = \arg \min_{\mathbf{W}} \bar{\epsilon}(\mathbf{W}, \mathbf{X}). \quad (2.8)$$

If we choose the maximally strict ‘all or nothing’ error function given by

$$\epsilon(\mathbf{W}, \mathbf{W}') = \begin{cases} 0, & \text{if } \mathbf{W} = \mathbf{W}', \\ 1, & \text{otherwise,} \end{cases} \quad (2.9)$$

then [equation \(2.8\)](#) recovers the maximum *a posteriori* (MAP) point estimate $\tilde{\mathbf{W}}(\mathbf{X}) = \mathbf{W}^*(\mathbf{X})$, with

$$\mathbf{W}^*(\mathbf{X}) = \arg \max_{\mathbf{W}} P(\mathbf{W} | \mathbf{X}). \quad (2.10)$$

However, this choice only highlights the lack of nuance the MAP estimator provides in quantifying uncertainty, since its corresponding error function does not account for any amount of gradation. Instead, we may wish to account for the mean squared error

$$\epsilon(\mathbf{W}, \mathbf{W}') = \sum_{i < j} (W_{ij} - W'_{ij})^2, \quad (2.11)$$

which provides a gradation not only for the errors of individual entries W_{ij} , but also between all the entries independently. In this case, the estimator of [equation \(2.8\)](#) becomes the pairwise posterior mean, $\tilde{W}_{ij}(\mathbf{X}) = \bar{W}_{ij}(\mathbf{X})$, with

$$\bar{W}_{ij}(\mathbf{X}) = \int W'_{ij} P(\mathbf{W}' | \mathbf{X}) d\mathbf{W}'. \quad (2.12)$$

Although this estimator may seem entirely reasonable at first, there is still one remaining issue left to consider. Namely, the scenario we most often expect to encounter is one where the underlying network \mathbf{W} is sparse, i.e. most of its entries are exactly zero. However, the posterior mean $\bar{W}_{ij}(\mathbf{X})$ will not be able to convey sparsity, unless the zeros of \mathbf{W} occur with absolute certainty in the posterior distribution. Or putting it differently, the posterior mean alone cannot distinguish between having a high probability for both zero and non-zero weights, or strictly non-zero weights distributed with the same mean.

We can address the sparsity estimation by considering an auxiliary dichotomization $A(\mathbf{W})$ with entries given by

$$A_{ij}(\mathbf{W}) = \begin{cases} 1, & \text{if } W_{ij} \neq 0, \\ 0, & \text{otherwise,} \end{cases} \quad (2.13)$$

and an error function given by

$$\epsilon(\mathbf{W}, \mathbf{W}') = \sum_{i < j} (W_{ij} - W'_{ij})^2 + \alpha [A_{ij}(\mathbf{W}) - A_{ij}(\mathbf{W}')]^2, \quad (2.14)$$

where $\alpha \geq 0$ denotes the relative importance of the sparsity structure in the estimation. If we assume $\alpha \rightarrow \infty$, the estimator of [equation \(2.8\)](#) becomes $\tilde{W}_{ij}(\mathbf{X}) = \hat{W}_{ij}(\mathbf{X})$, with

$$\hat{W}_{ij}(\mathbf{X}) = \begin{cases} \bar{W}_{ij}(\mathbf{X}), & \text{if } \pi_{ij}(\mathbf{X}) > \frac{1}{2}, \\ 0, & \text{otherwise,} \end{cases} \quad (2.15)$$

where

$$\pi_{ij}(\mathbf{X}) = \int A_{ij}(\mathbf{W}') P(\mathbf{W}' | \mathbf{X}) d\mathbf{W}' \quad (2.16)$$

is the marginal posterior probability of an edge having non-zero weight. We call the estimator of equation (2.15) simply the ‘marginal posterior’ (MP) estimator from now on. Its uncertainty can be quantified jointly by $\pi(\mathbf{X})$ and the marginal distributions

$$P(W_{ij}|\mathbf{X}) = \int \delta(W_{ij} - W'_{ij})P(\mathbf{W}'|\mathbf{X}) d\mathbf{W}', \quad (2.17)$$

or more succinctly, by the posterior variances

$$\sigma_{ij}^2(\mathbf{X}) = \int [W'_{ij} - \bar{W}_{ij}(\mathbf{X})]^2 P(\mathbf{W}'|\mathbf{X}) d\mathbf{W}'. \quad (2.18)$$

(a) Monte–Carlo sampling

The above estimators require us to perform posterior means of the type

$$\int g(\mathbf{W})P(\mathbf{W}|\mathbf{X}) d\mathbf{W}, \quad (2.19)$$

for a particular function $g(\mathbf{W})$, but exact evaluations of such integrals are in general intractable. Instead, we need to approximate them as

$$\int g(\mathbf{W})P(\mathbf{W}|\mathbf{X}) d\mathbf{W} \approx \frac{1}{S} \sum_{k=1}^S g(\mathbf{W}^{(k)}), \quad (2.20)$$

where $\{\mathbf{W}^{(1)}, \dots, \mathbf{W}^{(S)}\}$ are S samples from the posterior distribution, which becomes asymptotically exact for $S \rightarrow \infty$. The central surrogate task then becomes to obtain such samples efficiently. We address the main strategy and its obstacles in the following.

3. Posterior sampling and the quadratic mixing problem

Our approach for sampling from the posterior of equation (2.4) is to employ Markov-chain Monte–Carlo (MCMC) with the Metropolis–Hastings [28,29] acceptance criterion: Given an initial weighted adjacency matrix \mathbf{W} , we propose a new matrix \mathbf{W}' by first selecting a single entry (i, j) of \mathbf{W} with probability $Q(i, j|\mathbf{W})$, and then changing its value according to a local proposal $Q(W'_{ij}|i, j, \mathbf{W})$, and finally accepting the move with probability

$$a(\mathbf{W}', \mathbf{W}, i, j) = \min \left(1, \frac{P(\mathbf{W}'|\mathbf{X})Q(W_{ij}|i, j, \mathbf{W}')Q(i, j|\mathbf{W}')}{P(\mathbf{W}|\mathbf{X})Q(W'_{ij}|i, j, \mathbf{W})Q(i, j|\mathbf{W})} \right), \quad (3.1)$$

which accounts for the reverse move probability to enforce the detailed balance condition, given by

$$P(\mathbf{W}|\mathbf{X})Q(W'_{ij}|i, j, \mathbf{W})Q(i, j|\mathbf{W})a(\mathbf{W}', \mathbf{W}, i, j) = P(\mathbf{W}'|\mathbf{X})Q(W_{ij}|i, j, \mathbf{W}')Q(i, j|\mathbf{W}')a(\mathbf{W}, \mathbf{W}', i, j). \quad (3.2)$$

This condition guarantees that a Markov chain implemented in this way will have the target posterior $P(\mathbf{W}|\mathbf{X})$ as its stationary distribution—provided it exists, i.e. the Markov chain is aperiodic, and as long as the chosen proposal distributions $Q(i, j|\mathbf{W})$ and $Q(W'_{ij}|i, j, \mathbf{W})$ are ergodic, i.e. they allow for every possible value of the weighted adjacency matrix to be obtained with a non-zero probability after a finite number of moves.

An appealing property of the MCMC approach is that it obviates the computation of the usually intractable normalization constant $P(\mathbf{X})$ that completes the definition of the posterior distribution $P(\mathbf{W}|\mathbf{X})$, since this quantity appears both in the numerator and denominator of equation (3.1), and thus does not affect the acceptance rate. Therefore, using this scheme, only the joint likelihood $P(\mathbf{X}, \mathbf{W})$ is needed to be able to asymptotically sample from the posterior $P(\mathbf{W}|\mathbf{X})$.

However, the efficacy of the overall approach hinges crucially on the choice of the proposal distributions $Q(i, j|\mathbf{W})$ and $Q(W'_{ij}|i, j, \mathbf{W})$, since not all valid choices will lead to the same mixing time, i.e. the number of steps needed to reach the stationary distribution given some initial state. An efficient proposal distribution will result in fast mixing, allowing for sufficiently many

independent samples from the target distribution to be obtained with relatively short MCMC runs.

Perhaps the simplest overall scheme is to select the entry (i, j) to be updated uniformly at random, i.e.

$$Q_u(i, j) = \frac{\mathbb{1}_{i < j}}{\binom{N}{2}}. \quad (3.3)$$

Unfortunately, this simple idea will be extremely inefficient in the most empirically relevant scenarios, even if the local weight proposal $Q(W'|i, j, \mathbf{W})$ is chosen ideally. This will happen whenever the marginal distribution π defined in [equation \(2.16\)](#) is sufficiently concentrated on a sparse set of typical edges, with the remaining entries having $\pi_{ij} < \epsilon$, for some small probability ϵ . In this case, the total number of typical edges is given by

$$|\mathcal{E}| = \sum_{i < j} \mathbb{1}_{\pi_{ij} > \epsilon}. \quad (3.4)$$

If, for example, this number grows only linearly as $|\mathcal{E}| = O(N)$, then the uniform proposal of [equation \(3.3\)](#) will choose an atypical entry (i, j) , i.e. one for which $\pi_{ij} < \epsilon$, with a probability $1 - O(1/N)$, hence tending to one for $N \rightarrow \infty$. For such atypical entries, a move that changes its weight from zero to any non-zero value will be accepted with probability at most ϵ , meaning that the vast majority of moves will be wasted on vain attempts at placing unlikely edges. In this scenario, the average time needed to propose a single update to all $|\mathcal{E}|$ typical edges will scale as $O(N^2)$, which will be a lower bound to the overall mixing time of the Markov chain.

Instead, an efficient proposal would choose entries according to their probability to lead to a move being successful. A successful move proposal is one that combines two properties: 1. It gets accepted; 2. The new value for W'_{ij} is sufficiently different from the previous value W_{ij} —in particular if $W_{ij} = 0$ then $W'_{ij} \neq 0$, and vice versa. This means that an efficient entry proposal needs to be able to estimate the typical edge set—in other words, we need to be able to estimate, beforehand, which entries of the marginal posterior π have sufficiently high values. If this succeeds, we would be able to update all typical edges in time $O(N)$, significantly reducing the mixing time when compared to the uniform entry proposal of [equation \(3.3\)](#). We describe our approach to achieve this in the following.

(a) Estimating the typical edge set

Our basic idea to estimate the typical edge set is to exploit the information used to obtain the MAP estimate of [equation \(2.10\)](#), as described in [18,21]. More specifically, the algorithm for this purpose consists of iteratively improving the estimate for \mathbf{W}^* , starting from an initial $\mathbf{W} = \mathbf{W}^{(0)}$ at $t = 0$ containing all zeros, and proceeding as:

- (i) At iteration $t + 1$, given an initial $\mathbf{W} = \mathbf{W}^{(t)}$, we find the set $\mathcal{E}^{(t+1)}$ containing the κN entries of \mathbf{W} that most increase or least decrease the posterior $P(\mathbf{W}|\mathbf{X})$, with κ being a parameter of the algorithm.
- (ii) The entries of $\mathcal{E}^{(t+1)}$ are updated in sequence to maximize $P(\mathbf{W}|\mathbf{X})$, yielding a new estimate $\mathbf{W}^{(t+1)}$.
- (iii) If the difference between $\mathbf{W}^{(t+1)}$ and \mathbf{W}^t falls below some tolerance value, we return $\mathbf{W}^* = \mathbf{W}^{(t+1)}$, otherwise we continue from step 1.

A naïve implementation of step 1 would exhaustively search through all entries, taking time $O(N^2)$. Instead, as described in [21], it is possible to estimate $\mathcal{E}^{(t+1)}$ in subquadratic time, typically

$O(\kappa^2 N \log^2 N)$, using a recursive second-neighbour search. Our estimate $\hat{\mathcal{E}}$ for the typical set is then the union of all candidate entries encountered during the above algorithm, i.e.

$$\hat{\mathcal{E}} = \bigcup_{t=1}^T \mathcal{E}^{(t)}, \quad (3.5)$$

where T is the total number of iterations. Note that we are not interested only in the last set of candidate edges, nor in the non-zero entries of the final MAP estimate \mathbf{W}^* , since we want edges with a non-negligible marginal probability, not only the most likely ones.

Since T is typically a constant with respect to N , the total size of the typical set is $\hat{\mathcal{E}} = O(N)$. With our estimate $\hat{\mathcal{E}}$ at hand, we propose entries for the MCMC according to

$$Q(i, j) = \frac{w_t Q_t(i, j | \hat{\mathcal{E}}) + w_u Q_u(i, j)}{w_t + w_u}, \quad (3.6)$$

with

$$Q_t(i, j | \hat{\mathcal{E}}) = \frac{\mathbb{1}_{(i, j) \in \hat{\mathcal{E}}}}{|\hat{\mathcal{E}}|}, \quad (3.7)$$

and with w_t and w_u being the relative propensities of choosing entries in the set $\hat{\mathcal{E}}$ and uniformly, respectively. Note that we need $w_u > 0$ to guarantee ergodicity, but we expect $Q_t(i, j | \hat{\mathcal{E}})$ to yield the most successful proposals.

The above algorithm does not guarantee that all members of the typical set are found. To increase our chances of finding the entire set, we initialize the MCMC with the MAP estimate \mathbf{W}^* , and after a sweep comprised of N consecutive proposals, we compute a new set \mathcal{E}' according to the same algorithm used in step 1 of the above algorithm, and add it to our typical set estimate $\hat{\mathcal{E}}$. Note that since this changes the proposal probabilities that depend on $\hat{\mathcal{E}}$, this procedure will invalidate detailed balance, and therefore will not lead to a correct sampling of the target distribution. Because of this, we perform this update only for τ initial sweeps, and afterwards we continue sampling with final set $\hat{\mathcal{E}}$ fixed.

In figure 1, we demonstrate the behaviour of this algorithm on a the reconstruction of an Erdős–Rényi network of $N = 5000$ nodes and average degree $2E/N = 5$, and weights sampled from a normal distribution with mean $1/5$ and standard deviation 0.01 , serving as the couplings of a kinetic Ising model (see appendix E), after $M = 500$ parallel transitions from a random initial state. Figure 1a shows the cumulative recall of the typical set, i.e. the fraction of all entries with a posterior probability π_{ij} above a particular value that have been found in $\hat{\mathcal{E}}$, for several values of the search period τ . Although for $\tau = 0$ the recall is already 95% for the entire range of typical posterior probabilities, it increases continuously to 100% for $\tau = 10^3$, indicating that further posterior samples can improve the estimate of the initial greedy algorithm. In figure 1b is shown the evolution of the Jaccard similarity

$$s(\mathbf{W}', \mathbf{W}) = 1 - \frac{\sum_{i < j} |W'_{ij} - W_{ij}|}{\sum_{i < j} |W'_{ij} + W_{ij}|}, \quad (3.8)$$

between samples \mathbf{W}' generated by the MCMC and the true value \mathbf{W} —note that the similarity decays since the MCMC is initialized with a MAP estimate, which in this case has a larger similarity to the true network than typical samples from the posterior. Despite the search time τ being barely visible in the time-span considered, its longer-term effect is noticeable, since the MCMC run with $\tau = 10^3$ converges significantly faster than the one with $\tau = 0$, despite the cumulative recall being already 95% in the latter case. This is due to the fact that the remaining 5% of the typical edge set needs to be found by uniform sampling, which will still take an $O(N)$ number of sweeps. In the same figure, we also show the result with $w_t = 0$, i.e. using only uniform entry samples, which displays a much slower convergence. In figure 1c, we show the results of the same algorithms, but starting from an empty network (i.e. all entries being zero), where we can see that the uniform sampling takes a time at least two orders of magnitude larger to converge. Finally, in figure 1c we show the autocorrelation function of the similarity, discarding

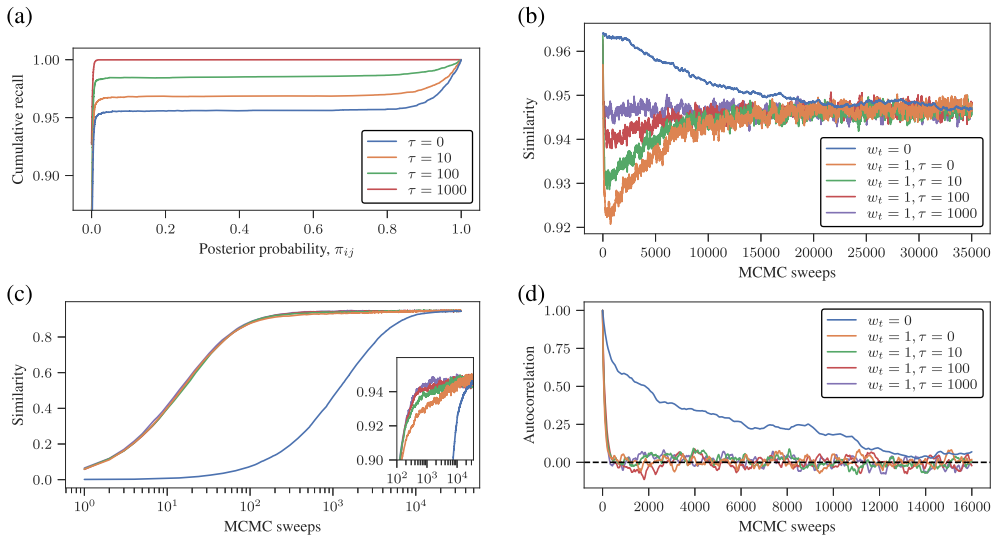


Figure 1. Results of MCMC runs for the reconstruction of an Erdős–Rényi network of $N = 5000$ nodes and average degree $2E/N = 5$, and weights sampled from a normal distribution with mean $1/5$ and standard deviation 0.01 , serving as the couplings of a kinetic Ising model (see appendix E), based on $M = 500$ parallel transitions from a random initial state. Panel (a) shows the cumulative recall of the typical set, i.e. the fraction of all entries with a posterior probability π_{ij} above a particular value that have been found in $\hat{\mathcal{E}}$, for several values of the search period τ . Panel (b) shows the Jaccard similarity $s(\mathbf{W}', \mathbf{W})$ between samples \mathbf{W}' generated by the MCMC and the true value \mathbf{W} , with ($w_t = 1$) and without ($w_t = 0$) the estimation of the typical edge set, and various search periods τ . Panel (c) shows the same kinds of MCMC runs, but with an initial state consisting of an empty network (the inset shows a zoom in the high similarity region). Panel (d) shows the autocorrelation function for the values of similarity of the runs in panel (b), discarding the initial transient before equilibration.

the transient towards equilibration, for the same runs as before. The runs with $w_t = 1$ yield autocorrelation times ranging from 300 ($\tau = 10^3$) to 600 ($\tau = 0$) sweeps, whereas runs with $w_t = 0$ have a significantly higher autocorrelation time of around 21 000 sweeps. This demonstrates how this scheme can have a substantial impact on the efficiency of drawing samples from the posterior distribution via MCMC.

(i) Searching for ‘nearby’ edges

The protocol described previously relies on a pre-processing phase aimed at determining the typical edge set, before the MCMC proper is run. Here, we present and evaluate an additional strategy which aims to continuously improve our estimate on the typical edge set during the MCMC, which consists of selecting preferentially entries that are ‘close’ to the current edges of the network (i.e. the non-zero entries of the current state of the MCMC). More specifically, we choose a node i uniformly at random, and the second node j uniformly from the set that is reachable from i in the dichotomized network $A(\mathbf{W})$ at a distance at most d , i.e.

$$R(i, j | \mathbf{W}, d) = \begin{cases} \frac{\mathbb{1}_{j \in \Lambda(i, d)}}{|\Lambda(i, d)|N'} & \text{if } |\Lambda(i, d)| > 0, \\ \frac{1}{N(N-1)} & \text{otherwise,} \end{cases} \quad (3.9)$$

where $\Lambda(i, d)$ is the set of nodes in $A(\mathbf{W})$ that are reachable from i at a distance at most d . Note that in general this proposal is asymmetric, $R(i, j | \mathbf{W}, d) \neq R(j, i | \mathbf{W}, d)$, so the final probability becomes

$$Q_n(i, j | \mathbf{W}, d) = \mathbb{1}_{i < j} [R(i, j | \mathbf{W}, d) + R(j, i | \mathbf{W}, d)]. \quad (3.10)$$

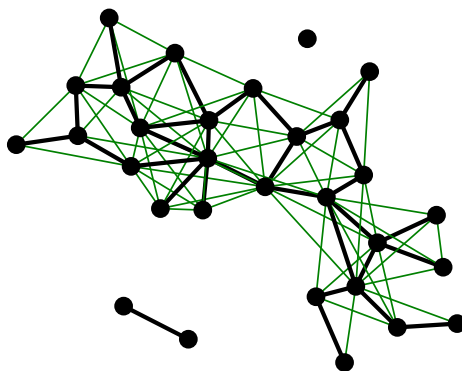


Figure 2. Illustration of the proposed ‘nearby’ updates according to [equation \(3.10\)](#). The black edges correspond to the non-zero entries of \mathbf{W} at some point of the algorithm, and the green edges are entries with $Q_n(i, j | \mathbf{W}, d) > 0$ for $d = 2$, which would be proposed for an update. Edges between the different components will never be proposed for any value of d .

By itself this proposal will not lead to an ergodic Markov chain, so it needs to be used together with the proposal of [equation \(3.6\)](#).

An illustration of the entries that are preferentially sampled in this manner is shown in [figure 2](#). The intuition behind this idea is that if the edges of $A(\mathbf{W})$ are already in the typical edge set \mathcal{E} , then the entries connecting indirect neighbours are likely to be in this set as well. This should happen with reconstruction problems with some degree of transitivity, i.e. when direct connections and those between second and third neighbours of their endpoints might have comparable or at least decaying posterior probabilities.

This approach will fail in two scenarios: (i) When the transitivity property is not applicable; (ii) When the current graph $A(\mathbf{W})$ is sufficiently disconnected, such that entries between different components are never preferentially proposed. We illustrate the behaviour of this kind of proposal on a target distribution given by $\hat{\pi} = \prod_{i < j} \hat{\pi}_{ij}$, with

$$\hat{\pi}_{ij} = p^{G_{ij}} \epsilon^{1-G_{ij}}, \quad (3.11)$$

where G is a random graph with an increased abundance of triangles, generated by first sampling an Erdős–Rényi network with E edges, removing $En/(n+1)$ edges uniformly at random, and then employing the following procedure n times in succession: Of all open triads in G —i.e. entries (i, j) such that $G_{ij} = 0$ and $G_{iu}G_{uj} = 1$ for some node $u \notin \{i, j\}$ — $E/(n+1)$ of them are selected uniformly at random and closed, i.e. $G_{ij} \rightarrow 1$. This guarantees that the final graph will have exactly E edges, and a significantly higher fraction of triangles than would be expected in an ER network. In [figure 3](#), we show the autocorrelation time for our proposed MCMC as a function of the number of nodes N , for $E = 5N/2$, considering different combinations of the move proposals so far considered, in the situation where the typical network is connected ($p = 0.9$) and where it is disconnected ($p = 0.1$). In the connected case, the nearby moves have no noticeable effect on the autocorrelation time when the initial estimate of the typical edge set is being used ($w_t = 1$), but it improves significantly the mixing when it is used on its own (in addition to the uniform moves)—in this case the autocorrelation does not grow linearly with N as in the case of using only uniform proposals. In the disconnected case, as expected, the nearby moves lose significantly their efficacy: when used on their own, the autocorrelation also increases linearly with N . However, even in this case, its use reduces the mixing time by a constant factor, even when combined with the initial estimation of the typical set. This approach is, therefore, potentially useful in situations where the typical edge set cannot be accurately estimated with the protocol described previously.

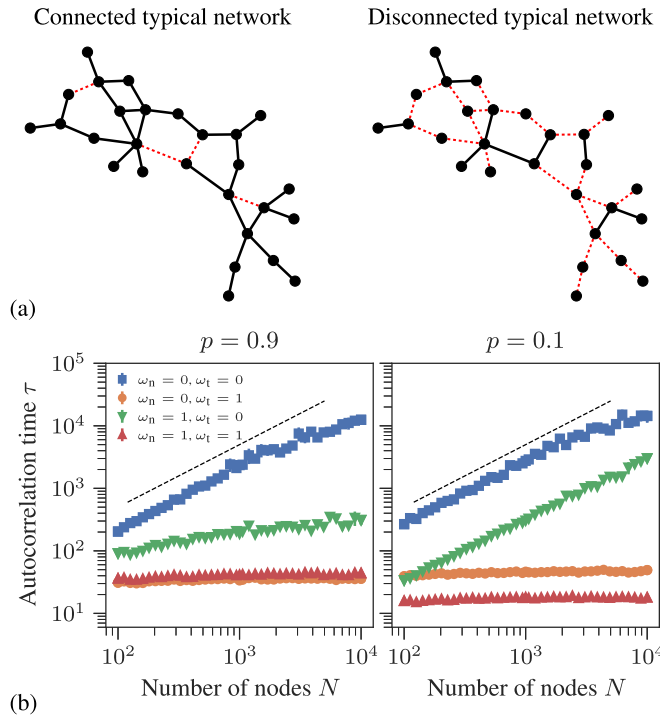


Figure 3. Panel (b) shows the autocorrelation time as a function of the number of nodes N , for a target distribution according to equation (3.11), with \mathbf{G} generated as described in the text, with $E = 5N/2$ edges, and considering different combinations of the move proposals, as indicated in the legend, in the situation where the typical network is connected ($p = 0.9$) and where it is disconnected ($p = 0.1$), in both cases with $\epsilon = 10^{-8}$. The dashed line indicates a linear slope. Panel (a) shows an illustration of the connected and disconnected cases, with black edges representing those in \mathbf{G} that are currently being sampled, and the dashed edges those in \mathbf{G} that are not.

(ii) Edge weights, node values and community structure

In the previous sections, we have focused on the move proposals $P(i, j | \mathbf{W})$ that involve the selection of entries in the matrix \mathbf{W} to be updated, but not on the proposals $P(\mathbf{W}' | i, j, \mathbf{W})$ to update the actual value of the entry selected, since the former is the most crucial for the algorithmic performance. For the value updates, conventional choices can in principle be used, such as sampling from a normal distribution. In appendix B we describe an alternative approach based on bisection sampling that we found to be efficient, and also works well with regularization schemes that rely on discretization, such as the minimum description length (MDL) formulation of [18], which we summarize in appendix A.

One feature of the MDL regularization is that it includes the stochastic block model [30] as a prior, and therefore it performs community detection as part of the reconstruction, which has been shown previously to improve the overall accuracy [31].

Furthermore, most models also include an additional set of parameters θ on the nodes, that also need to be updated. We have not included these parameters in our discussion so far, since they can be handled completely separately, by selecting one of them at random, and using the same kinds of updates as used for the entries of \mathbf{W} . Differently from \mathbf{W} , there is no inherent algorithmic challenge in sampling these node parameters, since their number scales only linearly with the number of nodes.

Finally, in appendix C we also describe an extension of the algorithm which allows for edge replacements and swaps, that can potentially move across likelihood barriers present when discretized regularization schemes are used.

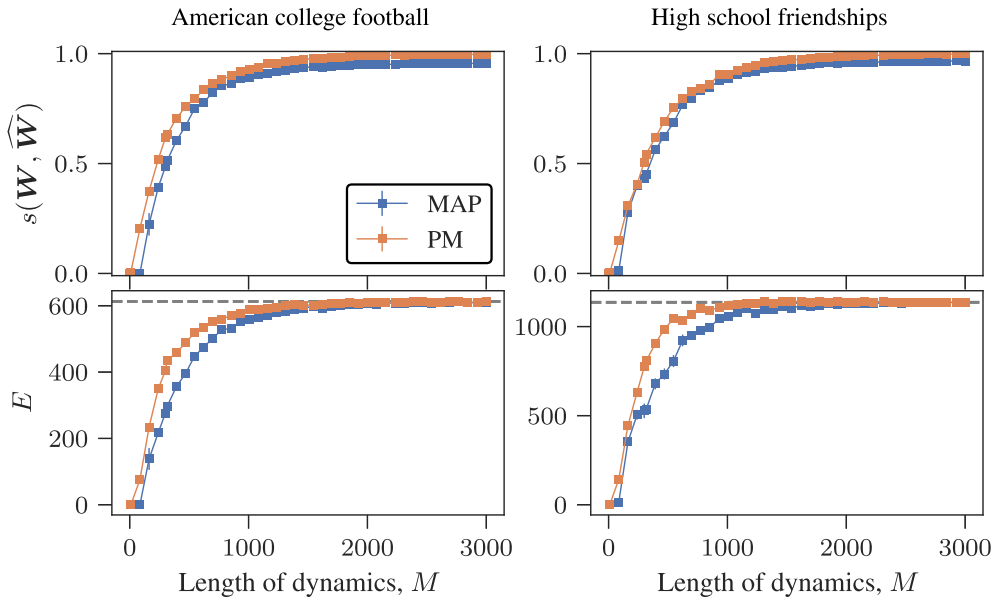


Figure 4. Reconstruction performance based on the dynamics generated by the kinetic Ising model (see appendix E) on two empirical networks, where the weights are sampled from a normal distribution with mean $1/\langle k \rangle$ and standard deviation 0.01, with $\langle k \rangle = 2E/N$ being the average degree. The left panels show the results for a network of American football teams [33] (with $N = 115$ and $E = 613$), and on the left for a network of friendship between high school students [34] (with $N = 291$ and $E = 1136$). The panels on the top show the similarity $s(\mathbf{W}, \hat{\mathbf{W}})$ between the inferred and true networks, according to the MAP and MP estimators, as indicated in the legend, as a function of the length M of the dynamics. The bottom panels show the number of edges of the inferred networks in each case. The dashed horizontal lines indicate the true value.

We provide a reference C++ implementation of the algorithms described here, together with documentation, as part of the `graph-tool` Python library [32].

4. MAP versus MP estimation with synthetic dynamics

In figure 4 we show a comparison between the MAP and MP estimates for synthetic dynamics, i.e. M transitions of the kinetic Ising model, on empirical networks, using the MDL regularization of [18], described in appendix A. For sufficient data, both estimates yield the same reconstruction. However, as data become more scarce, the MP estimator shows a systematically better performance, although the difference is not very large in these examples. The difference in performance is unsurprising, given that the derivation of the MP estimator results from the optimization of the mean squared error, as we discussed previously, and therefore it cannot be surpassed by MAP. Nevertheless, it serves as a good demonstration that obtaining the consensus over the posterior distribution can improve the accuracy of point estimates.

Besides the increased accuracy, posterior estimation can provide uncertainty quantification. We focus on this aspect when analysing the reconstruction based on empirical dynamics, in the following.

5. Empirical dynamics

In order to investigate the uncertainty information that posterior sampling can provide for network reconstruction, we first consider the voting dynamics in the lower house of the Brazilian congress, during the legislative period from 2007 to 2011, involving 623 deputies who voted ‘no’, ‘abstain’, or ‘yes’ on 619 voting sessions. We modelled this dynamics according to an

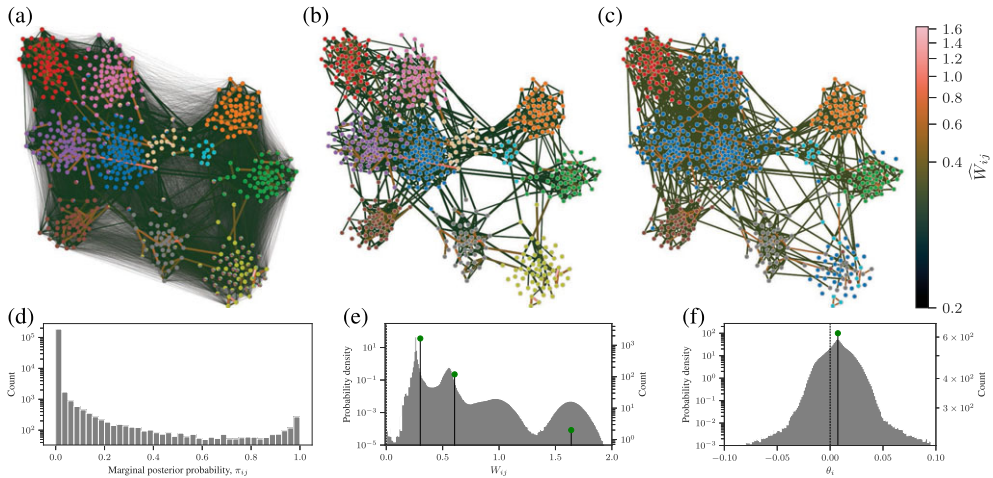


Figure 5. Reconstruction of a zero-added Ising model based on $M = 619$ votes of $N = 623$ deputies of the lower house of the Brazilian congress. (a) Marginal edge probabilities π indicated as edge thickness and the posterior mean \hat{W} as edge colors. The node pie charts indicate the marginal group memberships, inferred according to the SBM incorporated in the reconstruction, as described in [18]. (b) MP estimate \hat{W} according to equation (2.15). (c) MAP point estimate W^* according to equation (2.10). (d) Distribution of marginal posterior probability values π_{ij} across all node pairs. (e) Posterior distribution of non-zero weight values W_{ij} across all node pairs. (f) Distribution of node biases θ_i across all nodes i . In (e) and (f) the vertical lines correspond to the distribution obtained with the MAP point estimate.

equilibrium Ising model, modified to include the states $\{-1, 0, 1\}$, corresponding, respectively, to the aforementioned vote outcomes. The results are shown in figure 5.

The reconstruction uncovers a network ensemble that is divided into 11 groups of nodes who tend to vote in similar ways. As shown in figure 7, the divisions coincide very well with known party affiliations. The existence of non-zero couplings between deputies have uncertainties that vary in the entire $\pi_{ij} \in [0, 1]$ range, indicating a very heterogeneous mixture of certain and uncertain edges. The coupling strengths themselves are distributed around four typical values, whereas the node biases are centred closely around a typically small, but positive value, indicating that deputies have only a very small tendency to vote ‘yes’ in the absence of any interaction with their neighbours. The increased accuracy that the marginal estimate provides is noticeable when compared to the MAP estimate of figure 5c, for which only eight groups can be identified, with three groups in the government coalition being merged together (corresponding to the four groups in the upper left of figure 5b). The tenuous intra-coalition organization is only visible when the more detailed analysis from posterior sampling is performed, and implies that the observed dynamics cannot be well captured by a single network—at least not with the dynamical model used. The similarity between both estimates is $s(\hat{W}, W^*) = 0.72$, showing that, while there is a substantial agreement between both estimates, the disagreement is not negligible (unlike the sufficient data limit in figure 4), and indicates how posterior sampling can be important to uncover uncertainties in the analysis of empirical data.

Our approach allows us to query the individual marginal distributions $P(W_{ij}|X)$ for every pair (i, j) , giving a substantial amount of information on the reconstruction, when compared to the MAP point estimate, as can be seen in figures 5e and 5f.

We move now to another, larger dataset composed of $M = 2516$ log-returns of $N = 6369$ stocks in the US market, corresponding to 10 years from 2014 to 2024, obtained from Yahoo finance.² We performed a reconstruction using a multivariate Gaussian distribution (see appendix E), with W corresponding to the precision matrix, so that if $W_{ij} = 0$ it means that i and j are

²Retrieved from the API to <https://finance.yahoo.com>.

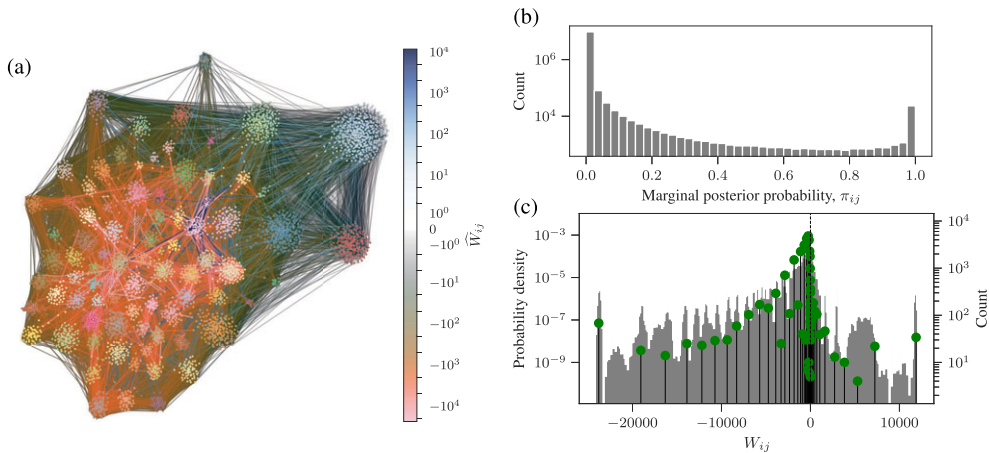


Figure 6. Reconstruction of a multivariate Gaussian model based on $M = 2516$ log-returns of $N = 6369$ US stocks in the period between 2014 to 2024. (a) Marginal edge probabilities π indicated as edge thickness and the posterior mean \hat{W} as edge colors. The node colors indicate the maximum marginal group memberships, inferred according to the SBM incorporated in the reconstruction, as described in [18]. (b) Distribution of marginal posterior probability values π_{ij} across all node pairs. (c) Posterior distribution of non-zero weight values W_{ij} across all node pairs. The vertical lines correspond to the distribution obtained with the MAP point estimate.

conditionally independent. The results are shown in figure 6. Similarly to the previous example, the reconstruction uncovers a modular network, with edge uncertainties spanning a wide range. As seen in figure 7, the groups found correlate moderately with the industry sector, although not as clearly as the correlation with party affiliation in the Brazilian congress example, considered previously. In this case, the correspondence between the MP and MAP estimates is higher, with a similarity $s(\hat{W}, W^*) = 0.83$, but the discrepancy is still not negligible, indicating a somewhat more concentrated posterior distribution (this can also be seen in figure 6b, which shows a larger abundance of edges with $\pi_{ij} \approx 1$).

(a) Comparison between posterior probabilities, weight magnitudes, and pairwise correlations

We take the opportunity to compare the outcome of our probabilistic reconstruction with commonly used heuristics for this task, based on pairwise correlations between the observable behaviour of nodes. The biggest disadvantage of this type of heuristic is the conflation it makes between direct and indirect neighbours, since if two connected nodes have a high correlation value, the same is also likely to be true between one of the endpoints involved and any of the neighbours of the other endpoints. For example, for any three vectors x , y and z , the Pearson correlation coefficient must fulfil

$$\text{corr}(x, z) \geq \text{corr}(x, y) \text{corr}(y, z) - \sqrt{[1 - \text{corr}(x, y)^2][1 - \text{corr}(y, z)^2]}. \quad (5.1)$$

Hence, e.g. if $\text{corr}(x, y) = \text{corr}(y, z) = 0.99$, then $\text{corr}(x, z) \geq 0.96$, regardless if x and z correspond to nodes that are truly connected or not. Since the covariance is related to the Pearson correlation via $\text{corr}(x, z) = \text{cov}(x, y) / \sqrt{\text{cov}(x, x) \text{cov}(y, y)}$, the same kind of inherent constraint also affects it. Similarly, mutual information satisfies

$$\text{MI}(x, z) \geq \text{MI}(x, y) + \text{MI}(y, z) - H(y), \quad (5.2)$$

where $H(y)$ is the entropy of y . Hence, if $\text{MI}(x, y) = \text{MI}(y, z) = H(y) - \epsilon$, then $\text{MI}(x, z) \geq H(y) - 2\epsilon$. Therefore, the idea of simply thresholding these quantities cannot be reconciled with the

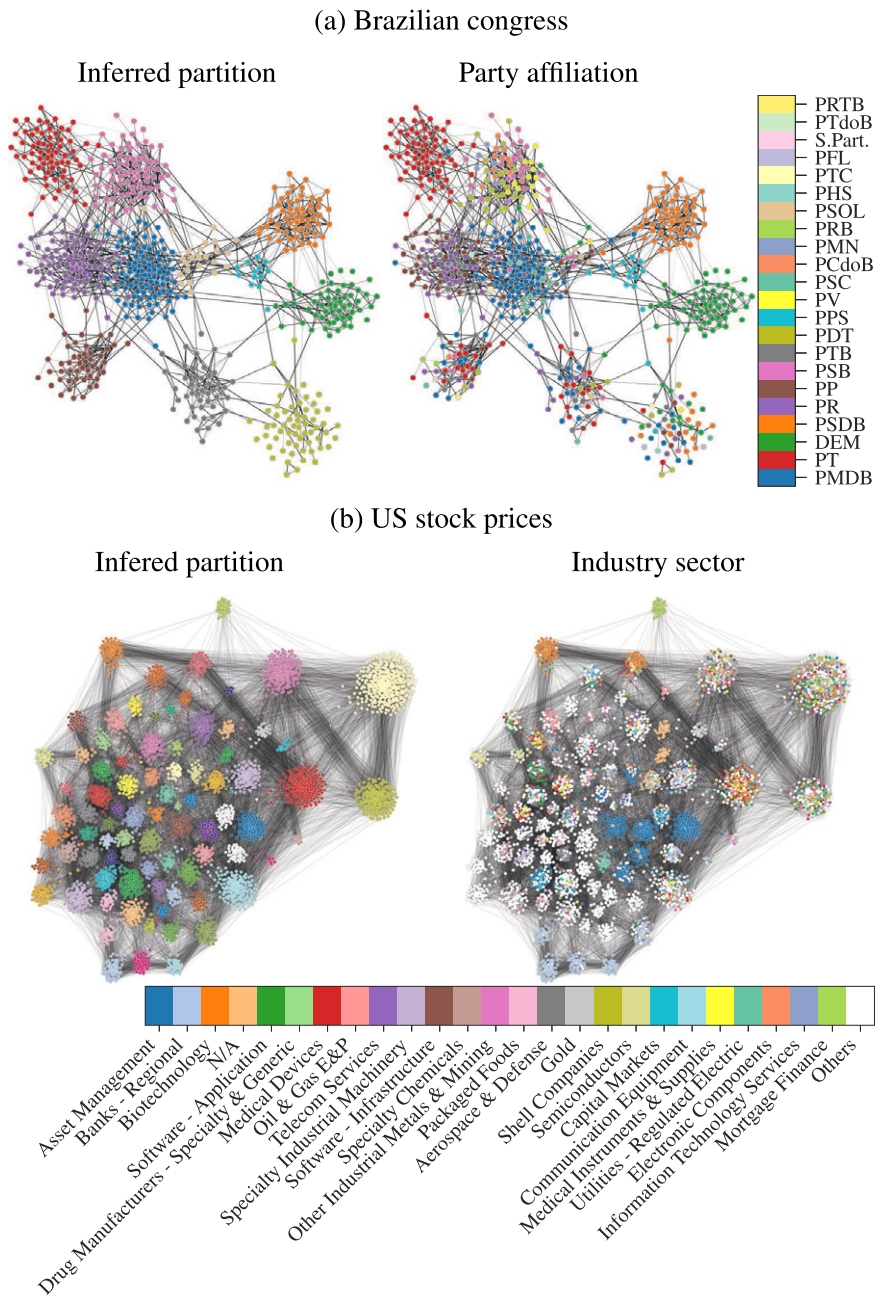


Figure 7. Correspondence between the inferred partition using the built-in SBM in our reconstruction (left) with available metadata on the nodes (right), for (a) the Brazilian congress, with the metadata being the party affiliation of the deputies, and (b) US stock prices, with the metadata being the industrial sector, in both cases as indicated in the legend.

distinction between direct and indirect neighbours, at least not in the general case. This contrasts markedly with the inferential approach considered in this work, for which such inherent constraints are inexistent.

Nevertheless, we might posit that there are situations where these reconstruction approaches yield similar results. For example, for a sparse, homogeneous true network, with all edges having the exact same weight, and all nodes having the same degree—such that the observed correlation between all true neighbours is approximately the same—it could be that the small drop in

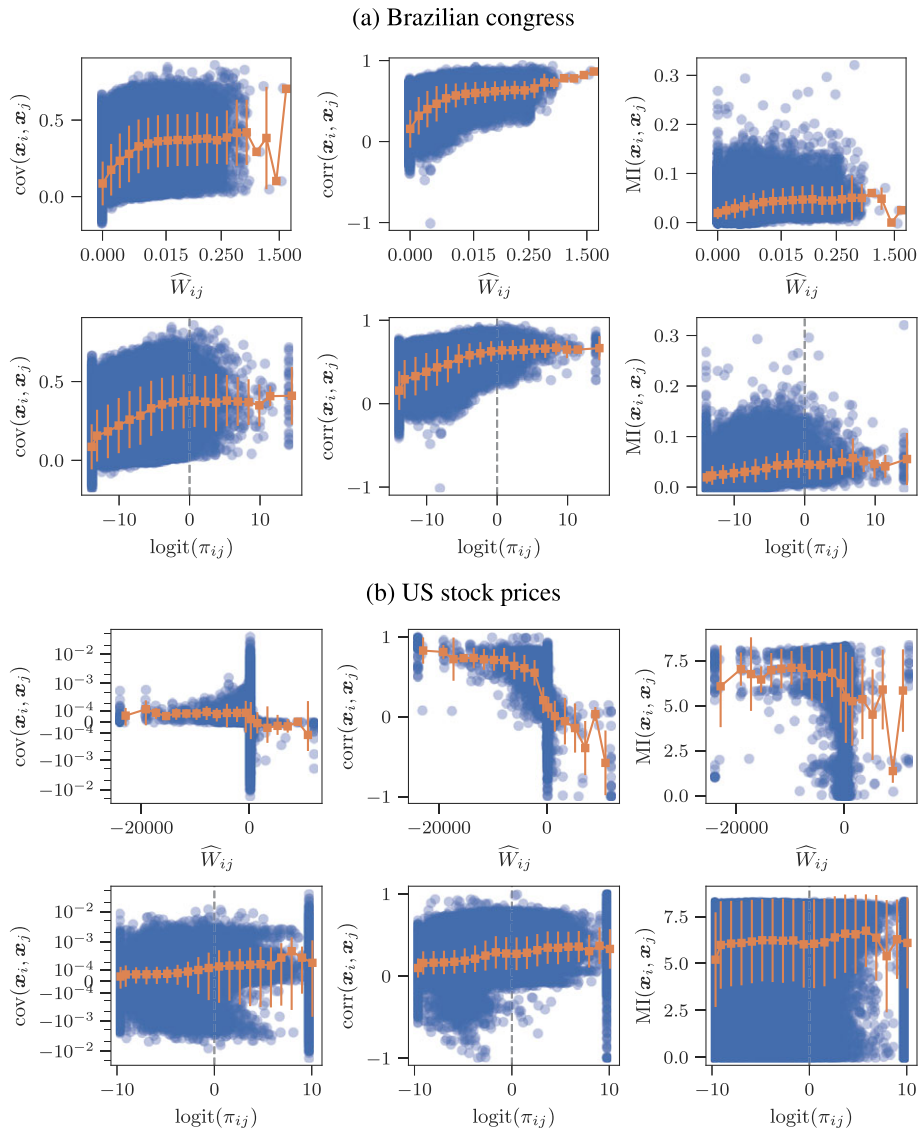


Figure 8. Scatter plot between mean posterior weights \widehat{W}_{ij} or posterior probabilities π_{ij} and a type of pairwise correlation, i.e. either the covariance $\text{cov}(x_i, x_j)$, Pearson correlation $\text{corr}(x_i, x_j)$, or mutual information $MI(x_i, x_j)$, for every node pair (i, j) , for (a) the Brazilian congress data, and (b) the US stock prices data. The connected orange points correspond to binned averages.

correlation between first and second neighbours is sufficient to discriminate between true and false edges.

In order to investigate the quantitative discrepancies outside such an idealized scenario, in figure 8 we show the correspondence between either the inferred weights or marginal edge probabilities and the three aforementioned correlation functions, for the two datasets considered so far. In all cases, although some positive correlations can be detected, they are very weak, meaning that these correlations are very inefficient predictors of both the presence of an edge and its weight magnitude. Importantly, the lack of correspondence occurs even at the extremes: we very often observe edge pairs with close to maximal correlation, but which nevertheless have a close to zero marginal edge probability, and conversely, nodes with very high marginal probability or inferred weight, but which have very low correlation values. This demonstrates that the inferences that are obtained via our reconstruction approach are leveraging much more

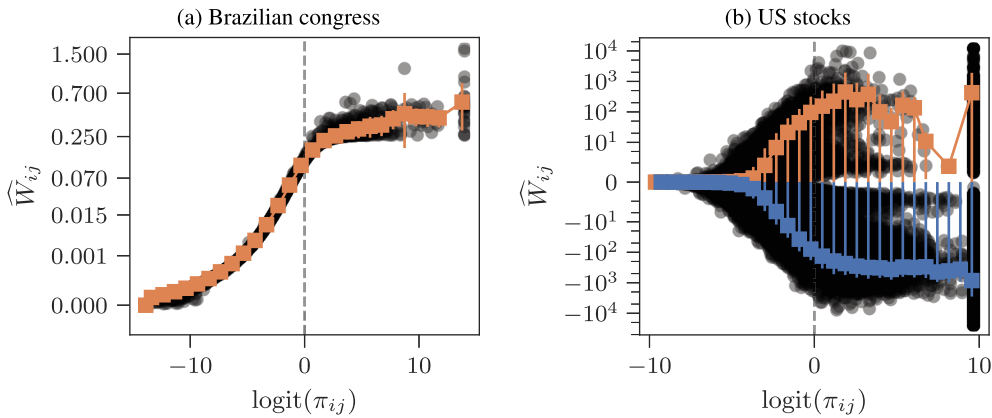


Figure 9. Scatter plot of mean posterior weights \hat{W}_{ij} versus posterior probabilities π_{ij} , for every node pair (i, j) , for (a) the Brazilian congress data, and (b) the US stock prices data. The connected orange points correspond to binned averages for positive weights, and the blue points for negative weights.

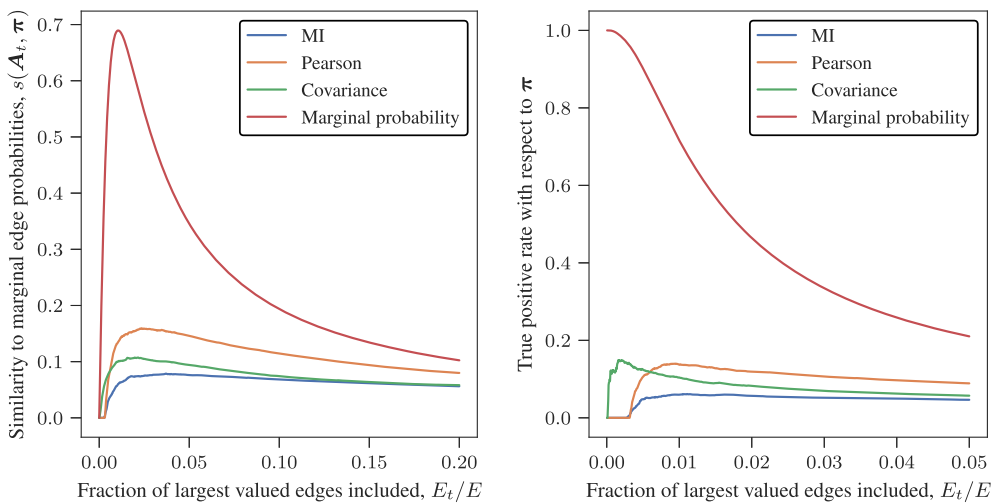


Figure 10. Accuracy according to the fraction of largest values included in the reconstruction, for the Brazilian congress data, for different kinds of ‘scores’ attributed to the edge pairs. The left plot shows the Jaccard similarity, while the right shows the ‘true positive’ rate, taking the marginal probability as a reference.

nuanced information in the data than simply whether the pairwise node correlations are either large or small.

Incidentally, we also investigated the correlation between the inferred weights and edge probabilities. Naïvely, one might expect that a large inferred weight magnitude is synonymous with a large marginal probability, but in reality the situation is more nuanced. It can be, for example, that a node accepts two other nodes as equally plausible neighbours with high weight magnitudes, but *not simultaneously*, i.e. it is either one node or the other, but not both. In this case, each of those edges will have a large weight, but a marginal posterior probability of only 50%. As can be seen in figure 9, in the case of the Brazilian congress data we do observe a positive correlation between weight and marginal probability, but it becomes significantly weaker above $\pi_{ij} = 1/2$, meaning that while a sufficiently low weight magnitude implies low probability, large weights do not necessarily have correspondingly high probabilities. On the other hand, for the US stocks data the correlation variance is much stronger, meaning that, while on average a larger weight implies higher probability, there is an abundance of exceptions, even at the extremes.

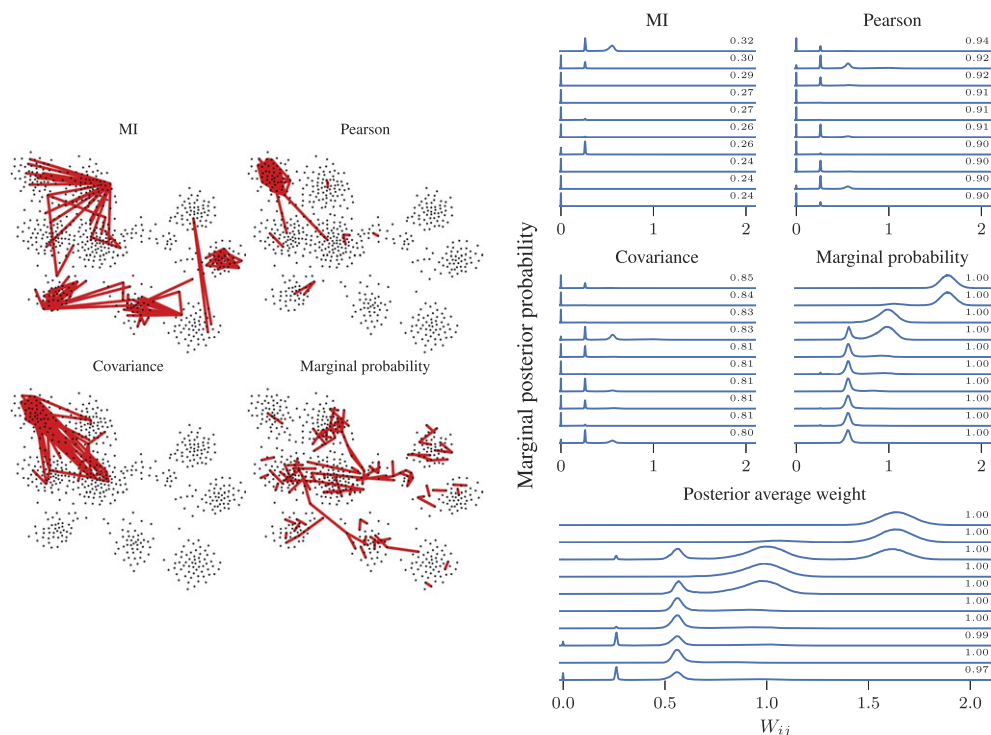


Figure 11. Left: First 100 edge pairs with the largest values of mutual information, Pearson correlation, covariance and marginal probability, for the Brazilian congress data. The layout of the nodes is the same as in figure 5. Right: Marginal weight distribution of the 10 highest ranking node pairs according to the same scores as in the top panel, as well as the posterior average weight. The upper right corners show the corresponding scores.

When we compare the reconstructed networks using correlation thresholds with the inferred ones, as we might expect from the above analysis, we obtain extreme discrepancies. In figure 10, we show the Jaccard similarity between the threshold-based reconstructions and the marginal probabilities for the Brazilian congress data, which peaks at around 0.16 for the Pearson correlation, representing the closest result overall. Even when considering only the true positive rate—which ignores the inclusion of spurious edges (false positives) in the reconstruction—the maximum value reaches only similar low ranges. Importantly, the different correlation functions also disagree significantly between themselves, as can be seen in figure 11, which shows the highest scoring node pairs in each case. The same figure also shows the marginal posterior distribution of weights for the same pairs, illustrating the lack of agreement between high correlation among nodes and the weights inferred.

From these comparisons, we can conclude that posterior sampling not only provides valuable uncertainty quantification, but also a completely different, and more accurate, reconstruction result than comparatively crude, but often employed heuristics based on thresholding of correlations.

6. Conclusion

We have described an efficient method to sample from posterior distributions of networks that allows us to perform uncertainty quantification for the problem of network reconstruction, as well as to produce consensus estimates from marginal distributions.

Our method does not rely on specific properties of particular generative models used for reconstruction, nor on the prior distribution used for their parameters. We showed how our

method can be used together with a sophisticated regularization scheme that uncovers the most appropriate number of edges and weight distribution in a manner consistent with the statistical evidence available in the data.

We have demonstrated on synthetic and empirical examples how posterior sampling can improve the accuracy of network reconstructions, and uncovers the entire range of possible reconstructions weighted according to their plausibility as an account of how the data has been generated.

A comparison with heuristics based on the thresholding of pairwise correlations revealed the relative advantage of performing an inferential reconstruction, since besides providing a generative model, uncertainty estimates, and significantly increased accuracy, it is capable of distinguishing between the probability of existence of an edge and its weight magnitude, which otherwise would be conflated.

Since our methodology is easily adaptable to other generative models, it remains to be explored how it can be employed with models more realistic than the relatively simple ones considered here, and how the underlying Bayesian framework can be leveraged to perform model selection, to investigate the fundamental limits of network reconstruction, and to obtain predictive statements about the unseen behaviour and the outcome of interventions in network systems, based solely on indirect non-network data.

Data accessibility. This article has no additional data.

Declaration of AI use. No, I have not used AI-assisted technologies in creating this article.

Authors' contributions. T.P.: conceptualization, formal analysis, funding acquisition, investigation, methodology, project administration, resources, software, supervision, validation, visualization, writing—original draft, writing—review and editing

Conflict of interest declaration. I declare I have no competing interests.

Funding. This work has been funded by the Vienna Science and Technology Fund (WWTF) and by the State of Lower Austria (grant no. 10.47379/ESS22032).

Appendix A. MDL regularization and joint SBM inference

Following [18], we consider a formulation of the edge weight priors based on a sparse, adaptive quantization of the allowed values, which amounts to an implementation of the minimum description length (MDL) principle. More specifically, we first sample an auxiliary unweighted multigraph A , specifying the placement of non-zero weights, according to the degree-corrected stochastic block model (DC-SBM) [35], here in its microcanonical formulation [36], with a likelihood

$$P(A|\mathbf{b}, \mathbf{k}, \mathbf{e}) = \frac{\prod_{r < s} e_{rs}! \prod_r e_{rr}! \prod_i k_i!}{\prod_{i < j} A_{ij} \prod_i A_{ii}! \prod_r e_r!}, \quad (\text{A } 1)$$

where $\mathbf{b} = \{b_i\}$ is the node partition, with $b_i \in \{1, \dots, B\}$ being the group membership of node i , $\mathbf{k} = \{k_i\}$ is the degree sequence, with k_i being the degree of node i , and $\mathbf{e} = \{e_{rs}\}$ is the group affinity matrix, with e_{rs} being the number of edges between groups r and s , or twice that if $r = s$. Based on the multigraph A , a simple graph G is obtained by ‘erasing’ the edge multiplicities [37],

$$P(G|A) = \prod_{i < j} (1 - \delta_{A_{ij}, 0})^{G_{ij}} \delta_{A_{ij}, 0}^{1 - G_{ij}}. \quad (\text{A } 2)$$

Conditioned on G , we sample the non-zero weights from a finite set of K values $\mathbf{z} = \{z_1, \dots, z_K\}$, conditioned on their exact counts $\mathbf{m} = \{m_k\}$, where $m_k = \sum_{i < j} \delta_{W_{ij}, z_k}$, and otherwise uniformly, according to

$$P(\mathbf{W}|\mathbf{z}, \mathbf{m}, G) = \frac{\prod_k m_k!}{E!} \times \prod_k \delta_{m_k, \sum_{i < j} \delta_{W_{ij}, z_k}} \times \prod_{i < j} \delta_{W_{ij}, 0}^{1 - G_{ij}}, \quad (\text{A } 3)$$

with the non-zero counts themselves sampled uniformly according to

$$P(\mathbf{m}|\mathbf{K}, A) = \frac{\delta_{\sum_k m_k, E(A)} \prod_k \mathbb{1}_{m_k > 0}}{\binom{E(A)-1}{K-1}}, \quad (\text{A } 4)$$

where $E(A)$ is the number of non-zero entries in A . In [18], the weight categories were sampled according to a discrete Laplace distribution. Instead, here we propose a slight variation, where only the extreme values z_1 and z_K are sampled jointly as

$$P(z_1, z_K | \lambda, \Delta) = \mathbb{1}_{z_1 \leq z_K} (2 - \delta_{z_1, z_K}) P(z_1 | \lambda, \Delta) P(z_K | \lambda, \Delta), \quad (\text{A } 5)$$

where

$$P(z | \lambda, \Delta) = \begin{cases} e^{-\lambda|z|} (e^{\lambda\Delta} - 1)/2, & \text{if } z = \Delta \lceil \frac{z}{\Delta} \rceil, \text{ and } z \neq 0 \\ 0, & \text{otherwise,} \end{cases} \quad (\text{A } 6)$$

is a quantized zero-excluded Laplace distribution, with decay and quantization parameters, λ and Δ , respectively, each sampled uniformly from the set of all strictly positive real numbers representable by q bits, i.e. $P(\Delta|q) = P(\lambda|q) = 2^{-q}$, for which we pragmatically choose $q = 64$. Conditioned on these extreme values, we sample the remaining $K - 2$ distinct values uniformly as

$$P(z_2, \dots, z_{K-1} | \Delta, K, z_1, z_K) = \frac{\prod_{k=2}^{K-1} \delta_{z_k, \Delta \lfloor z_k / \Delta \rfloor} \times \prod_{k=1}^{K-1} \mathbb{1}_{z_k < z_{k+1}}}{\binom{(z_K - z_1)/\Delta - 1 - \mathbb{1}_{z_1 z_K < 0}}{K-2}}. \quad (\text{A } 7)$$

Lastly, the number K of discrete weight values is sampled uniformly inside the allowed range according to

$$P(K | \Delta, z_1, z_K) = \frac{\mathbb{1}_{1 \leq K \leq (z_K - z_1)/\Delta + 1 - \mathbb{1}_{z_1 z_K < 0}}}{(z_K - z_1)/\Delta + 1 - \mathbb{1}_{z_1 z_K < 0}}. \quad (\text{A } 8)$$

Putting it all together, we have

$$\begin{aligned} P(W | A, \lambda, \Delta) &= \sum_G P(G | A) \left[\sum_{z, K} P(W | z, G) P(z_2, \dots, z_{K-1} | \Delta, K, z_1, z_K) P(K | z_1, z_K, \Delta) P(z_1, z_K | \lambda, \Delta) \right]^{1 - \delta_{E(G), 0}} \\ &= \prod_{i < j}^{\delta_{A_{ij}, 0}} \delta_{W_{ij}, 0} \left[\frac{(\prod_k m_k!) e^{-\lambda(|z_1| + |z_K|)} (e^{\lambda\Delta} - 1)^2 (2 - \delta_{z_1, z_K})}{4 \times E! \binom{E-1}{K-1} \binom{(z_K - z_1)/\Delta - 1 - \mathbb{1}_{z_1 z_K < 0}}{K-2}} \right]^{1 - \delta_{E(A), 0}}, \end{aligned} \quad (\text{A } 9)$$

where the remaining quantities m , z , K and E in equation (A 9) should be interpreted as being functions of W .

With this prior at hand, we can formulate the problem of reconstruction according to the joint posterior

$$P(W, A, b, \lambda, \Delta | X) = \frac{P(X | W) P(W | A, \lambda, \Delta) P(A | b) P(b) P(\lambda) P(\Delta)}{P(X)}, \quad (\text{A } 10)$$

where the marginal distribution $P(A | b) = \sum_{k, e, E} P(A | b, k, e) P(k | e) P(e | E) P(E)$ is computed using the priors described in [36], in particular those corresponding to the hierarchical (or nested) SBM [38]. The prior for the total number of multiedges, $P(E) = [\mu/(\mu + 1)]^E / (\mu + 1)$, is a geometric distribution with mean $\mu = \binom{N}{2}$, and comparable standard deviation $\sigma_E = \sqrt{\mu(\mu + 1)} \approx \binom{N}{2}$, for $N \gg 1$.

The proposals for the partitions b are done according to the merge-split algorithm described in [39]. Although it is straightforward to introduce move proposals for both λ and Δ , we found that the results are often indistinguishable from simply choosing $\lambda = 1$ and $\Delta = 10^{-8}$, since these are not very sensitive hyperparameters.

For generative models which have additional node parameters, e.g. local fields of the Ising model (see appendix E), almost identical priors can be used for them, with the only exception being that zero values are allowed. See [18] for details.

Appendix B. Edge weight proposals via bisection and linear interpolation (BLI)

In the main text, we focused on selecting which node pairs to update, but gave no details about how the edges should be updated, i.e. what should be the move proposal $Q(W'_{ij}|W, i, j)$ after we have selected the node pair (i, j) . A standard approach in this case would be to choose for this a normal distribution centred on the previous value, and with some user-defined variance. However, this has as a drawback that the variance needs to be carefully chosen, which in general requires a substantial degree of experimentation and fine-tuning. Here we describe an alternative bisection and linear interpolation (BLI) approach, that is self-adaptive and does not require fine-tuning. We start with a triplet (W_a, W_b, W_c) , with $W_a < W_b < W_c$, that ‘brackets’ a maximum in the conditional posterior $f(W) = P(W_{ij} = W | W \setminus W_{ij}, X)$, i.e.

$$f(W_a) < f(W_b), \quad f(W_b) > f(W_c). \quad (\text{B } 1)$$

If this condition is fulfilled, then there is at least one local maximum in the interval $[W_a, W_c]$. Such a triplet can be found by considering an initial $(W_a^{\text{init}}, y, W_c^{\text{init}})$, with W_a^{init} and W_c^{init} being initial guesses that bound the typical range of weight values, and y is sampled uniformly at random in the interval enclosed by these values. If this initial choice does not bracket a maximum, the boundary W with the largest $f(W)$ is multiplied by a factor 2. This procedure is repeated until a bracketing interval is found, and the difference between $\log f(W_b) - \log \max(f(W_a), f(W_c))$ is sufficiently large, e.g. more than 200 or so, such that values outside this range can be neglected as having a vanishingly small probability. Having obtained this bracketing interval, we proceed with a random bisection search:

- (i) We sample y uniformly at random between either $[W_a, W_b]$ or $[W_b, W_c]$, depending on which interval is larger.
- (ii) The new bracketing interval is updated to include y as its midpoint and the old midpoint W_b as one of the boundaries if $f(y) > f(W_b)$, otherwise the midpoint is preserved and the corresponding boundary is updated to y .
- (iii) If $\log f(W_b) - \log \max(f(W_a), f(W_c)) < \epsilon$, the search stops. Otherwise, we go back to step 1.

The above algorithm will converge to a *local* maximum of $f(W)$ after $O(\log(1/\epsilon))$ iterations on average. The fact we select the midpoint uniformly at random—instead of deterministically like in the golden section search method [40]—means that we can, in principle, obtain any local maximum contained in the initial interval.

Our objective is to produce a sample proposal from $f(W)$, not to optimize it. Hence, we construct a distribution formed by a linear interpolation between all the points considered during the random bisection algorithm above, which by necessity involves the neighbourhood of at least one local maximum, and therefore probes regions of relatively high probability from the target distribution. This interpolation requires a number of points $n = O(\log(1/\epsilon))$, and a single sample from it can be generated in time $O(n)$, by first computing the relative probability mass for each linear segment, then sampling a linear segment according to these probabilities (e.g. with the alias method [41,42], requiring time $O(n)$), and finally sampling the final value inside the interval in time $O(1)$ by an inverse transform. An example run of this scheme is shown in figure 12 for a multimodal target distribution. As can be seen in figure 12b, which shows an average of many such proposals, the proposals tend to concentrate around the modes of the target distribution, and, in this example, more than four bisections does not bring noticeable improvements—therefore only very few likelihood evaluations are needed. In figure 12c is also shown the average Metropolis–Hastings (MH) acceptance rate as a function of the number of bisections, demonstrating the same saturation at around four bisections for this particular example.

For the specific generative models considered in the main text and in appendix E, their corresponding conditional likelihood $f(W)$ is convex, which means that a deterministic bisection could be used instead. However, in the interest of generality, our algorithm does not rely on the

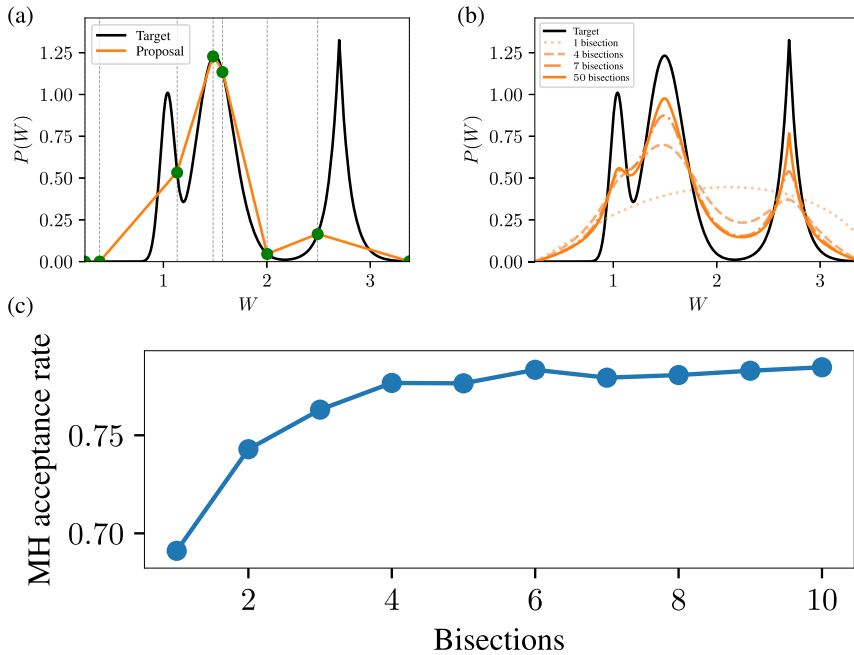


Figure 12. (a) Example target distribution and the proposal generated via the algorithm described in the main text. The circle markers and the vertical lines mark the random bisection points. (b) Average proposal distribution for increasing number of bisection steps, as shown in the legend. (c) Metropolis–Hastings (MH) acceptance rate as a function of the number of bisections.

convexity of the conditional likelihood, nor on other usually desirable properties such as it being differentiable or even continuous.

We note also that when computing the MH acceptance probability, it is not necessary to include the probability of choosing the bisection points themselves, nor the marginal probability averaged over all of them. We notice this by considering the detailed balance condition

$$f(W')T(W'|\boldsymbol{\gamma})P(\boldsymbol{\gamma}) = f(W)T(W|\boldsymbol{\gamma})P(\boldsymbol{\gamma}),$$

with $\boldsymbol{\gamma}$ being the random bisection points chosen with the above algorithm. If this condition is fulfilled, then the marginal detailed balance is also trivially fulfilled, i.e. $f(W')T(W') = f(W)T(W)$, with $T(W) = \int T(W|\boldsymbol{\gamma})P(\boldsymbol{\gamma})d\boldsymbol{\gamma}$, and the MH acceptance is computed as

$$\begin{aligned} a &= \min \left(1, \frac{f(W')P(W|\boldsymbol{\gamma})P(\boldsymbol{\gamma})}{f(W)P(W|\boldsymbol{\gamma})P(\boldsymbol{\gamma})} \right) \\ &= \min \left(1, \frac{f(W')P(W|\boldsymbol{\gamma})}{f(W)P(W|\boldsymbol{\gamma})} \right), \end{aligned}$$

which is independent of $P(\boldsymbol{\gamma})$ and depends only on the probability $P(W|\boldsymbol{\gamma})$ of sampling the final value according to the bisection points $\boldsymbol{\gamma}$, which is easily computed from the linear interpolation.

(a) Discrete values

When dealing with the discretized values for W considered in appendix A, special considerations are needed. Although we can easily adapt the above BLI sampling to values which are multiples of the quantization parameter Δ , this may not yield proposals which are accepted, since most of the time the proposal will yield a new value of z_k , increasing the number K of discrete categories,

which, per design, exerts a penalty to the likelihood. Because of this, we consider the following move types:

- (i) New categories: BLI moves constrained to values which are multiples of Δ .
- (ii) Old categories: BLI moves constrained to the existing categories, z .
- (iii) Collective category moves: BLI moves of a single category z_k with $k \in \{1, \dots, K\}$, to a new value which is a multiple of Δ , distinct from the other categories.

Move types 1 and 2 are mutually required to fulfil detailed balance, since, if the current category has more than one count, the move to a new category can only be reversed by a move to a previously existing category, and vice versa for the vanishing of an existing category with a single count. Move type 3 will simultaneously involve all the edges that belong to the same category, and thus can be seen as a non-local move that can speed up the MCMC convergence, and is an inherent advantage offered by our discretized approach.

Furthermore, we also employ the merge-split of [39] for the distribution of the weight categories on the edges, since this can remove likelihood barriers that exist when moving one edge at a time. The only modification we use for that algorithm is that when weight categories are split and merged, the respective category values z_k , both for old and new categories, are sampled according to the BLI algorithm described previously.

Appendix C. Updating multiple entries simultaneously: edge replacements and swaps

The move proposals considered in the main text all involve the update of a single entry of the matrix W at a time. In the presence of non-convex regularization schemes that penalize the excessive abundance of edges, we can encounter scenarios where the respective removal and addition of edges in two different entries of W would be individually rejected, but if these are performed at the same time their combined move would be accepted. In this way, the regularization can introduce ‘barriers’ in the posterior landscape that slow down the mixing of the Markov chain. In order to avoid this, here we consider also updates that involve two entries simultaneously. The first type of move is an edge replacement, performed as follows:

- (i) A node i is sampled uniformly at random.
- (ii) A neighbour of j is sampled uniformly at random with probability

$$P_e(j|i) = \begin{cases} \frac{A_{ij}}{\sum_u A_{iu}}, & \text{if } \sum_u A_{iu} > 0, \\ \frac{1}{N}, & \text{otherwise,} \end{cases}$$

where we account also for nodes with degree zero.

- (iii) A node v is sampled with probability $P_f(v|i)$.
- (iv) If $|\{i, j, v\}| < 3$, i.e. at least one of the nodes is repeated, the proposal is skipped.
- (v) Otherwise, the values of the entries W_{ij} and W_{iv} are swapped.

In the above, the new potential neighbour is sampled in step 3 with probability

$$P_f(j|i) = pP_{\hat{\varepsilon}}(j|i) + (1 - p)P_{\Lambda}(j|i), \quad (\text{C1})$$

where p is the probability of sampling according to the typical edge set estimate, i.e.

$$P_{\hat{\varepsilon}}(j|i) = \begin{cases} \frac{G_{ij}}{\sum_u G_{iu}}, & \text{if } \sum_u G_{iu} > 0, \\ \frac{1}{N}, & \text{otherwise,} \end{cases} \quad (\text{C2})$$

where G is the adjacency matrix corresponding to the edges in $\hat{\mathcal{E}}$, or otherwise they are sampled according to the nodes reachable from i , i.e. with probability

$$P_{\Lambda}(j|i) = \begin{cases} q \frac{\mathbb{1}_{j \in \Lambda(i,d)}}{|\Lambda(i,d)|} + \frac{1-q}{N}, & \text{if } |\Lambda(i,d)| > 0, \\ \frac{1}{N}, & \text{otherwise,} \end{cases} \quad (\text{C3})$$

and where $\Lambda(i,d)$ is the set of nodes reachable from i at a distance at most d , and $1-q$ is the probability of choosing v uniformly at random.

Note that the above proposal will either move an edge from (i,j) to (i,v) , if $W_{iv} = 0$, or simply swap their weights otherwise. However, if a move is performed, it will change the number of neighbours of v and j . Because of this, we can also consider a swap proposal that can preserve the degrees of all nodes involved, namely, we select four nodes $\{i, j, u, v\}$ according to

$$P(i, j, u, v) = P(i)P_e(j|i)P_f(u|j)P_e(v|u), \quad (\text{C4})$$

and if $|\{i, j, u, v\}| < 4$, i.e. at least one of the nodes is repeated, we skip the proposal, otherwise we swap W_{ij} with W_{iv} , and W_{uv} with W_{uj} . Note that this will preserve the node degrees only if W_{ij} and W_{uv} are both non-zero, and W_{iv} and W_{uj} are both zero.

We do not analyze the effect of these move proposals in detail, but they are included in our reference implementation, and we have observed a positive effect in the mixing time of empirical networks.

Appendix D. Parallelization

The algorithm of [21] used here to estimate the typical edge set can be performed in parallel, which often yields significant runtime improvements in multiprocessor environments. Unfortunately, MCMC algorithms in general, including the one we present to perform posterior sampling, are inherently serial, since we need to consider one move before the next one can be contemplated. Nevertheless, in our particular case partial parallelization can in fact be achieved by noting that if edges (i, j) and (u, v) are considered in sequence, and if all their endpoint nodes are different, then their individual contributions to the model likelihood (i.e. excluding prior terms) are completely independent. Since this contribution is the most computationally demanding, taking time $O(M)$ where M the total number of data samples, then we can benefit from parallelization as follows:

- (i) A set of edge candidate moves of size L is proposed according to the current state.
- (ii) For the subset of edges that form an independent set, i.e. those that do not share endpoints, their likelihood contributions are computed in parallel.
- (iii) The MCMC proceeds through the proposed moves sequentially, using the pre-computed likelihood changes if they are available, otherwise they are computed as needed.

This way of proceeding preserves the detailed balance of the MCMC algorithm (as long as we compute the forward and backward probabilities according to the above), since it is in fact being performed sequentially, but benefits from parallelization by pre-computing the most demanding parts of the likelihood computation in parallel, as long as most edges proposed belong to an independent set. In practice, we observe that this relatively simple strategy can improve significantly the MCMC run times when multiple processors are available—although it tends to be more subject to thread contention than the algorithm of [21], which does not require any thread synchronization.

Appendix E. Generative models

In our examples, we use three generative models: the equilibrium Ising model [16], the kinetic Ising model and a multivariate Gaussian.

The kinetic Ising model is a Markov chain on N binary variables $\mathbf{x} \in \{-1, 1\}^N$ with transition probabilities given by

$$P(\mathbf{x}(t+1)|\mathbf{x}(t), \mathbf{W}, \boldsymbol{\theta}) = \prod_i \frac{e^{x_i(t+1)(\sum_j W_{ij}x_j(t) + \theta_i)}}{2 \cosh(\sum_j W_{ij}x_j(t) + \theta_i)} \quad (\text{E } 1)$$

with θ_i being a local field on node i .

The equilibrium Ising model is the $t \rightarrow \infty$ limiting distribution of the above dynamics, with a likelihood given by

$$P(\mathbf{x}|\mathbf{W}, \boldsymbol{\theta}) = \frac{e^{\sum_{i<j} W_{ij}x_ix_j + \sum_i \theta_i x_i}}{Z(\mathbf{W}, \boldsymbol{\theta})}, \quad (\text{E } 2)$$

with $Z(\mathbf{W}, \boldsymbol{\theta}) = \sum_{\mathbf{x}} e^{\sum_{i<j} W_{ij}x_ix_j + \sum_i \theta_i x_i}$ being a normalization constant. Since this normalization cannot be computed analytically in a closed form, we make use of the pseudolikelihood approximation [43],

$$\begin{aligned} P(\mathbf{x}|\mathbf{W}, \boldsymbol{\theta}) &= \prod_i P(x_i|\mathbf{x} \setminus x_i, \mathbf{W}, \boldsymbol{\theta}) \\ &= \prod_i \frac{e^{x_i(\sum_j W_{ij}x_j + \theta_i)}}{2 \cosh(\sum_j W_{ij}x_j + \theta_i)} \end{aligned} \quad (\text{E } 3)$$

—which essentially approximates equation (E 2) as the probability of a transition of the global state of the kinetic Ising model onto itself—since it gives asymptotically correct results and has excellent performance in practice [16,44].

In the case of the zero-valued Ising model with $\mathbf{x} \in \{-1, 0, 1\}^N$, the normalization of equations (E 3) and (E 1) changes from $2 \cosh(\cdot)$ to $1 + 2 \cosh(\cdot)$.

Finally, the (zero-mean) multivariate Gaussian is a distribution on $\mathbf{x} \in \mathbb{R}^N$ given by

$$P(\mathbf{x}|\mathbf{W}) = \frac{e^{-1/2\mathbf{x}^\top \mathbf{W} \mathbf{x}}}{\sqrt{(2\pi)^N |\mathbf{W}^{-1}|}}, \quad (\text{E } 4)$$

where \mathbf{W} is the precision (or inverse covariance) matrix. Unlike the Ising model, this likelihood is analytical—nevertheless, the evaluation of the determinant is computationally expensive, and, therefore, we make use of the same pseudolikelihood approximation [45],

$$P(\mathbf{x}|\mathbf{W}, \boldsymbol{\theta}) = \prod_i \frac{e^{-(x_i + \theta_i^2 \sum_{j \neq i} W_{ij}x_j)^2 / 2\theta_i^2}}{\sqrt{(2\pi)\theta_i}}, \quad (\text{E } 5)$$

where we parameterize the diagonal entries as $\theta_i = 1/\sqrt{W_{ii}}$.

References

1. Netrapalli P, Sanghavi S. 2012 Learning the graph of epidemic cascades. In *Proceedings of the 12th ACM SIGMETRICS/PERFORMANCE Joint International Conference on Measurement and Modeling of Computer Systems SIGMETRICS '12, London, UK, 11–15 June 2012*, pp. 211–222. New York, NY, USA: ACM. (doi:10.1145/2254756.2254783)
2. Braunstein A, Ingrosso A, Muntoni AP. 2019 Network reconstruction from infection cascades. *J. R. Soc. Interface* **16**, 20180844. (doi:10.1098/rsif.2018.0844)
3. Faust K, Raes J. 2012 Microbial interactions: from networks to models. *Nat. Rev. Microbiol.* **10**, 538–550. (doi:10.1038/nrmicro2832)
4. Guseva K, Darcy S, Simon E, Alteio LV, Montesinos-Navarro A, Kaiser C. 2022 From diversity to complexity: microbial networks in soils. *Soil Biol. Biochem.* **169**, 108604 (doi:10.1016/j.soilbio.2022.108604)
5. Wang YXR, Huang H. 2014 Review on statistical methods for gene network reconstruction using expression data. *J. Theor. Biol.* **362**, 53–61. (doi:10.1016/j.jtbi.2014.03.040)

6. Pratapa A, Jaliha AP, Law JN, Bharadwaj A, Murali TM. 2020 Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data. *Nat. Methods* **17**, 147–154. (doi:10.1038/s41592-019-0690-6)
7. Bullmore E, Sporns O. 2009 Complex brain networks: graph theoretical analysis of structural and functional systems. *Nat. Rev. Neurosci.* **10**, 186–198. (doi:10.1038/nrn2575)
8. Zhang B, Horvath S. 2005 A general framework for weighted gene co-expression network analysis. *Stat. Appl. Genet. Mol. Biol.* **4**. (doi:10.2202/1544-6115.1128)
9. Horvath S. 2011 *Weighted network analysis: applications in genomics and systems biology*, 1st edn. SpringerLink Bücher. New York, NY: Springer Science+Business Media, LLC.
10. Tumminello M, Lillo F, Mantegna RN. 2010 Correlation, hierarchies, and networks in financial markets. *J. Econ. Behav. Organ.* **75**, 40–58. (doi:10.1016/j.jebo.2010.01.004)
11. Zhou D, Gozolchiani A, Ashkenazy Y, Havlin S. 2015 Teleconnection paths via climate network direct link detection. *Phys. Rev. Lett.* **115**, 268501. (doi:10.1103/PhysRevLett.115.268501)
12. Becker M *et al.* 2023 Large-scale correlation network construction for unraveling the coordination of complex biological systems. *Nature Comput. Sci.* **3**, 346–359. (doi:10.1038/s43588-023-00429-y)
13. Peel L, Peixoto TP, De Domenico M. 2022 Statistical inference links data and theory in network science. *Nat. Commun.* **13**, 6794. (doi:10.1038/s41467-022-34267-9)
14. Dempster AP. 1972 Covariance selection. *Biometrics* **28**, 157–175. (doi:10.2307/2528966)
15. Friedman J, Hastie T, Tibshirani R. 2008 Sparse inverse covariance estimation with the graphical Lasso. *Biostatistics* **9**, 432–441. (doi:10.1093/biostatistics/kxm045)
16. Nguyen HC, Zecchina R, Berg J. 2017 Inverse statistical problems: from the inverse Ising problem to data science. *Adv. Phys.* **66**, 197–261. (doi:10.1080/00018732.2017.1341604)
17. Gelman A, Vehtari A, Simpson D, Margossian CC, Carpenter B, Yao Y, Kennedy L, Gabry J, Bürkner PC, Modrák M. 2020 Bayesian workflow. (<http://arxiv.org/abs/10.48550/arXiv.2011.01808>)
18. Peixoto TP. 2025 Network reconstruction via the minimum description length principle. *Phys. Rev. X* **15**, 011065. (doi:10.1103/PhysRevX.15.011065)
19. Rissanen J. 1978 Modeling by shortest data description. *Automatica* **14**, 465–471. (doi:10.1016/0005-1098(78)90005-5)
20. Rissanen J. 2010 *Information and complexity in statistical modeling*, 1st edn. 2007 edn. New York, NY: Springer.
21. Peixoto TP. 2024 Scalable network reconstruction in subquadratic time. (<http://arxiv.org/abs/10.48550/arXiv.2401.01404>)
22. Butts CT. 2003 Network inference, error, and informant (in)accuracy: a Bayesian approach. *Soc. Netw.* **25**, 103–140. (doi:10.1016/S0378-8733(02)00038-2)
23. Guimerà R, Sales-Pardo M. 2009 Missing and spurious interactions and the reconstruction of complex networks. *Proc. Natl Acad. Sci. USA* **106**, 22 073–22 078. (doi:10.1073/pnas.0908366106)
24. Newman MEJ. 2018 Network structure from rich but noisy data. *Nat. Phys.* **14**, 542–545. (doi:10.1038/s41567-018-0076-1)
25. Peixoto TP. 2018 Reconstructing networks with unknown and heterogeneous errors. *Phys. Rev. X* **8**, 041011. (doi:10.1103/PhysRevX.8.041011)
26. Young JG, Cantwell GT, Newman MEJ. 2021 Bayesian inference of network structure from unreliable data. *J. Complex Netw.* **8**, cnaa046. (doi:10.1093/comnet/cnaa046)
27. Jaynes ET. 2003 *Probability theory: the logic of science*. Cambridge, UK: Cambridge University Press.
28. Metropolis N, Rosenbluth AW, Rosenbluth MN, Teller AH, Teller E. 1953 Equation of state calculations by fast computing machines. *J. Chem. Phys.* **21**, 1087. (doi:10.1063/1.1699114)
29. Hastings WK. 1970 Monte Carlo Sampling methods using Markov chains and their applications. *Biometrika* **57**, 97–109. (doi:10.1093/biomet/57.1.97)
30. Holland PW, Laskey KB, Leinhardt S. 1983 Stochastic blockmodels: first steps. *Social Netw.* **5**, 109–137. (doi:10.1016/0378-8733(83)90021-7)
31. Peixoto TP. 2019 Network reconstruction and community detection from dynamics. *Phys. Rev. Lett.* **123**, 128301. (doi:10.1103/PhysRevLett.123.128301)
32. Peixoto TP. 2014 The graph-tool python library. Figshare. (doi:10.6084/m9.figshare.1164194)
33. Girvan M, Newman MEJ. 2002 Community structure in social and biological networks. *Proc. Natl Acad. Sci. USA* **99**, 7821–7826. (doi:10.1073/pnas.122653799)

34. Moody J. 2001 Peer influence groups: identifying dense clusters in large networks. *Social Netw.* **23**, 261–283. (doi:10.1016/S0378-8733(01)00042-9)
35. Karrer B, Newman MEJ. 2011 Stochastic blockmodels and community structure in networks. *Phys. Rev. E* **83**, 016107. (doi:10.1103/PhysRevE.83.016107)
36. Peixoto TP. 2017 Nonparametric Bayesian inference of the microcanonical stochastic block model. *Phys. Rev. E* **95**, 012317. (doi:10.1103/PhysRevE.95.012317)
37. Peixoto TP. 2020 Latent Poisson models for networks with heterogeneous density. *Phys. Rev. E* **102**, 012309. (doi:10.1103/PhysRevE.102.012309)
38. Peixoto TP. 2014 Hierarchical block structures and high-resolution model selection in large networks. *Phys. Rev. X* **4**, 011047. (doi:10.1103/PhysRevX.4.011047)
39. Peixoto TP. 2020 Merge-split Markov chain Monte Carlo for community detection. *Phys. Rev. E* **102**, 012305. (doi:10.1103/PhysRevE.102.012305)
40. Press WH, Teukolsky SA, Vetterling WT, Flannery BP. 2007 *Numerical recipes*, 3rd edn: the art of scientific computing. Cambridge, UK: Cambridge University Press.
41. Walker A. 1974 New fast method for generating discrete random numbers with arbitrary frequency distributions. *Electron. Lett.* **10**, 127–128. (doi:10.1049/el:19740097)
42. Vose M. 1991 A linear algorithm for generating random numbers with a given distribution. *IEEE Trans. Software Eng.* **17**, 972–975. (doi:10.1109/32.92917)
43. Besag J. 1974 Spatial interaction and the statistical analysis of lattice systems. *J. R. Stat. Soc.: Ser. B (Methodological)* **36**, 192–225. (doi:10.1111/j.2517-6161.1974.tb00999.x)
44. Mozeika A, Dikmen O, Piili J. 2014 Consistent inference of a general model using the pseudolikelihood method. *Phys. Rev. E* **90**, 010101 (doi:10.1103/PhysRevE.90.010101)
45. Khare K, Oh SY, Rajaratnam B. 2015 A convex pseudolikelihood framework for high dimensional partial correlation estimation with convergence guarantees. *J. R. Stat. Soc. Ser. B: Stat. Methodol.* **77**, 803–825. (doi:10.1111/rssb.12088)