



Topology estimation method for telecommunication networks

Miika Rajala¹ · Risto Ritala²

Published online: 7 February 2018
© Springer Science+Business Media, LLC, part of Springer Nature 2018

Abstract

Network topology specifies the interconnections of nodes and is essential in defining the qualitative network behaviour which is known to be universal with the similar phenomena appearing in many complex networks in diverse application fields. Topology information may be uncertain, or several pieces of inconsistent topology information may exist. This paper studies a method for estimating the network topology directly from node data, and is motivated by mobile telecommunications networks (MTNs). Mutual information based dependency measure is first used to quantify the statistical node dependencies, and the topology estimate is then constructed with multidimensional scaling and distance thresholding. The topology estimate defines the graph structure of a Markov random field (MRF) model, and after model parameter identification, the MRF model can then be used e.g. in analyzing the effect of disturbances to the overall network state of MTN. The method is evaluated with MCMC generated data and is found to work in qualitative network behaviour situations that are practical from the application perspective of MTNs. With the same data, the method yields at least as good results as a typical constrained-based graph estimation method.

Keywords Topology estimation · Graph estimation · Markov random field model · Ising model · Multidimensional scaling (MDS) · Telecommunications networks

1 Introduction

Networked systems exist in great diversity of application fields such as human social networks, business relationship networks, neural networks, the Internet, and many technical networks, from power grids and supply chains to mobile telecommunications networks (MTNs). Through co-operative interaction of the nodes, networks perform operations such as delivering goods (e.g. supply chains), or exchanging (e.g. MTNs, Internet, social networks) and processing (e.g. neural networks, social networks) information. The performance of such coherent networked systems may be extremely effective and robust, but on the other hand, may be vulnerable for failures even in single nodes.

Topology of the network defines the node-to-node interactions, from which the complex behaviour and the properties

of the network emerge. Complex network behaviour may arise as spontaneous organisation [60] leading to coherence in node states, as hysteresis with discontinuous transitions of the overall network state, or as scale invariant coherence in state fluctuations [57]. These qualitative phenomena are known to be similar, universal, for many complex networks [3,42] in diverse application fields.

Information of topology in many networks is usually uncertain, or there may exist several pieces of topology information, perhaps inconsistent and overlapping. As an example, in MTNs there exist topologies both due to the physical locations and the logical relations of the nodes. In such cases one would have to know which topology information to exploit. However, topology estimated from data directly captures the best combination of the domain-based topologies. On the other hand, mobile ad hoc networks are an example of MTNs that does not have a fixed topology at all, instead the topology may change dynamically and hence can be tracked by estimating from data.

In the literature, methods exist for estimating the topology from data for both directed and undirected graphs. Directed graphs define the structure of a Bayesian network while undirected graphs define the structure of a Markov random field (MRF). One of the most straightforward approaches for con-

✉ Miika Rajala
miika.rajala@live.fi

Risto Ritala
risto.ritala@tut.fi

¹ Metsä Group, P.O. Box 10, 02020 Metsä, Finland

² Laboratory of Automation and Hydraulics, Tampere University of Technology, P.O. Box 527, Tampere, Finland

structuring a graph is simply to threshold node dependencies modelled e.g. with mutual information; see e.g. [14]. Score-based methods are used mainly to estimate directed graphs [19,22] but have also been applied for undirected graphs [32]. Constrained-based methods are generally used for both types of graphs [34] and are based on performing a set of conditional independence tests for nodes [51]. Examples include the Grow-Shrink (GS) [38] methods and the algorithm for undirected graphs is called the Grow-Shrink Markov Network (GSMN) [10]. Also some other topology estimation methods exist, mainly aimed for some specific application (see, e.g. [9,43]).

In this paper, we will consider a rather simple and straightforward topology estimation scheme, aiming for the identification of the topology of MTNs. The method is based on an assumption that the states of two nodes depend statistically on their interaction which is the weaker the further apart the nodes are. Mutual information (MI) [17] based measure is first employed to quantify the statistical dependencies between the network nodes and then multidimensional scaling [21] is applied to transform the node dependencies into a 2D map of node locations. Finally, the node location map is thresholded into a graph representation, which describes the node neighbourhoods, and also defines the structure of a MRF model [8]. In this paper, we apply a binary state Ising model [27,33].

The proposed topology estimation scheme is validated with Markov Chain Monte Carlo (MCMC) [36] generated data under a wide range of network parameters leading to a varying coherence in the overall network state. In particular, the test cases are selected according to their relevance for the MTNs. We will study how the performance of the method is affected by the sizes of the network, the node neighbourhoods, the generated data set, and the type of node loadings used. Finally, the proposed topology estimation method is compared to the GSMN method, and we will also discuss the advantages and limitations of our topology estimation approach.

The method we are considering in this paper was first proposed in [47,49], where we introduced the basic concept of the method but without any extensive tests with data in varying situations which is the focus of this paper. In [49] we have shown the qualitative phenomena the underlying Ising model is capable of describing. All the results presented here are previously published in the dissertation thesis of one of the author of this paper [46], where we have also applied the method to real telecommunications network data.

The rest of this paper is organised as follows. In Sect. 2 we introduce the general topology estimation scheme for networked systems. Section 3 discusses MRFs as models for networked systems, represents the Ising model, and shortly reviews MCMC for data generation purposes. Methods to quantify the goodness of a topology estimate are represented

in Sect. 4. In Sect. 5 the topology estimation scheme is analyzed and validated with synthetic data. Section 6 concludes the paper.

2 Graph structure estimation

This section presents general algorithms for estimating graph structure of a networked system based on the state and load data of the network nodes. First the statistical dependencies of nodes are measured with an information theoretic dependency measure. Then a node location map is obtained by employing multidimensional scaling, and finally, a graph structure estimate is formed by thresholding the node distances in the node location map.

2.1 State and load data of network nodes

Let us assume that the nodes of a networked system considered are known but the graph structure is unknown. Each node has a state which is a random variable assuming either discrete or continuous values and an external load assuming continuous values. Node state as a random variable is denoted by S_m , its value by s_m , the external load by h_m and the location coordinate vector by \mathbf{x}_m . Subscript m labels the nodes; $m = 1, \dots, M$, where M is the total number of nodes.

Network observations are labelled with superscript l ($l = 1, \dots, L$): $\{s_m^l, h_m^l\}_{m=1}^M$. We assume that the graph structure of the network and the network parameters are not varied within the data set and that node states and loads are observed without measurement uncertainty. In real network data the load variation is dictated by external conditions.

2.2 Node dependency measures

Mutual information (MI) is based on the concept of entropy and is a measure of the amount of information one random variable contains about another random variable [17]. In this paper, MI is used for measuring the statistical dependencies (similarities) of node pairs. In the literature MI is used as a similarity measure in many applications, including biomedical image registration [15], statistical language translation [13] and in research of networked systems interactions [20].

MI is defined for two random variables S_i and S_j as [17]:

$$\begin{aligned} I(S_i; S_j) &= H(S_i) - H(S_i|S_j) \\ &= \sum_{s_i} \sum_{s_j} p(s_i, s_j) \log \frac{p(s_i, s_j)}{p(s_i) p(s_j)}. \end{aligned} \quad (1)$$

Here p_i, p_j , and p_{ij} denote the marginal and joint probability distributions of the variables. $H(S_i)$ is the entropy of S_i and $H(S_i|S_j)$ is the conditional entropy of S_i given S_j .

MI is uncertain when estimated from a finite set of observations. Furthermore, the uncertainty depends on the number of observations. Therefore, we use a dependency measure which is less sensitive to the amount of data, the statistical significance of MI (SSMI). SSMI is thus more appropriate in comparing node pair dependencies when the data set size varies.

SSMI is defined through a null hypothesis that S_i and S_j , are statistically independent: $p_{ij}^{(0)} = p_i^{(obs)} p_j^{(obs)}$. We obtain $p_i^{(obs)}$ and $p_j^{(obs)}$ from $p_{ij}^{(obs)}$ which is the observed joint distribution. Under the null hypothesis, MI estimated from any finite data set of L observations is always positive. Its distribution, $f^{(0)}(I|L)$ can be described as a histogram estimate of N MI values. The values for the histogram are obtained by generating N sets of L state pairs according to $p_{ij}^{(0)}$, and calculating MI estimate for each set. Denoting the observed MI by $I(S_i; S_j|L)$, the SSMI, $\sigma_{MI}(S_i, S_j)$, is now obtained as

$$\sigma_{MI}(S_i, S_j) = \int_0^{I(S_i; S_j|L)} f^{(0)}(I|L) dI, \tag{2}$$

where $0 < \sigma_{MI} < 1$. The probability that the null hypothesis is erroneously discarded is $1 - \sigma_{MI}(S_i, S_j)$.

As SSMI is calculated from simulated random observations, it is itself a random variable, the uncertainty of which depends both on N and L . However, this uncertainty can be reduced at fixed L by increasing N , whereas uncertainty in MI is determined by L .

We have previously studied another measure called the statistical significance of χ^2 -statistics (SSCSS), where the χ^2 -statistics is an approximation of MI [40,45]. The advantage of SSCSS over SSMI is that, in the case of the null hypothesis, the analytical form of its distribution is always known, and it is thus much less computationally demanding than the SSMI. We have also found SSCSS to give results very similar to SSMI [47], and hence it may be more suitable for practical applications; for more details on SSCSS, see e.g. [46]. However, in this paper we will only use the exact measure, the SSMI, for not to introduce any additional uncertainty to the results.

2.3 Multidimensional scaling

Multidimensional scaling (MDS) is a method to transfer dissimilarity measures of variables into a low dimensional location map, where the distances between the nodes (variables) represent dissimilarities of the variables. In the literature MDS has been applied e.g. to visualize genes [29]

and databases [4], and for estimating position and velocity of mobile stations [25]; see also [61]. MI and MDS have also been used together for searching a spatial configuration to model dependencies in speech and music data [1] and to analyse word relations [39].

Let us consider a set of variables V with any symmetric dissimilarity: $\delta_{ij} = \delta_{ji}$ (here: $\delta_{ij} = \delta_{ji} = 1 - \sigma_{ij}$), defined for each pair of variables $(i, j) \in V$. Our goal is to find a 2-dimensional location map in which the node distances optimally describe the dissimilarities of the variables. On the location map, we denote the coordinate vectors of S_i and S_j by \mathbf{x}_i and \mathbf{x}_j and their Euclidean distance by $d_{ij}(\mathbf{x}_i, \mathbf{x}_j)$. To construct the node location map, we minimize Kruskal's stress-1 criterion [30,31]:

$$K_1 = \sqrt{\frac{\sum_{(i,j) \in V} [d_{ij}(\mathbf{x}_i, \mathbf{x}_j) - \hat{d}_{ij}]^2}{\sum_{(i,j) \in V} d_{ij}(\mathbf{x}_i, \mathbf{x}_j)^2}}, \tag{3}$$

where disparities (or target distances) \hat{d}_{ij} are monotonically related to the observed similarities δ_{ij} : $\hat{d}_{ij} < \hat{d}_{kl} \iff \delta_{ij} < \delta_{kl}$. While metric MDS utilizes the absolute values of the dissimilarities, non-metric MDS uses only the rank information of the dissimilarities and is thus more robust and more practical than the metric MDS for real observed dissimilarity values containing measurement uncertainties and distortions.

Iterative Shepard-Kruskal scaling algorithm can be used for minimizing K_1 [30,31]. In this iterative algorithm first the node coordinates in the location map are initialised randomly and the corresponding node distances are determined. Secondly, monotone regression is employed to relate the current distances to the original dissimilarities, producing a new set of dissimilarities, called disparities. Thirdly, the coordinates are revised by minimising K_1 for the distances to better match the disparities. Steps 2 and 3 are repeated until the fit is satisfying.

2.4 Graph structure estimation method

A graph structure estimate is obtained by defining a threshold value d_{thr} for the node distances, uniform through the estimated location map. Then if $d_{ij}(\mathbf{x}_i, \mathbf{x}_j) < d_{thr}$ for nodes i and j , $i \in N(j)$ and $j \in N(i)$, where N denotes the set of neighbours. For convenience, we will abbreviate this graph structure estimation method as the MGMN method (MDS-based Graph estimation for Markov Networks). Sketch of this algorithm is presented as Algorithm 1.

Algorithm 1 Sketch of the MGMN method, modified from [46]

1. **Initialise** U as the set of all node variables: $\forall S_i; i = 1, \dots, M: S_i \in U$
 2. **Initialise** the adjacency matrix G as an unconnected graph: $\forall S_i, S_j: G(S_i, S_j) = 0$
 3. **Estimate** $\sigma(S_i, S_j)$ for all node pairs $(S_i, S_j); S_i \in U, S_j \in U$
 4. **Calculate** $\delta(S_i, S_j)$ for all node pairs $(S_i, S_j); S_i \in U, S_j \in U$ as $\delta(S_i, S_j) = 1 - \sigma(S_i, S_j)$
 5. **Form** $M \times M$ dissimilarity matrix D with $D(i, j) = \delta(S_i, S_j)$
 6. **Apply** MDS to D to obtain a low-dimensional map with node locations \mathbf{x}_i for all nodes $S_i; i = 1, \dots, M: S_i \in U$, and with internode distances $d_{ij}(\mathbf{x}_i, \mathbf{x}_j)$ for all node pairs $(S_i, S_j); S_i \in U, S_j \in U$
 7. **Specify** a threshold distance parameter d_{thr}
 8. **For** all node pairs $(S_i, S_j); S_i \in U, S_j \in U$
 9. **If** $d_{ij}(\mathbf{x}_i, \mathbf{x}_j) < d_{\text{thr}}$ **then**
 10. Set $G(S_i, S_j) = 1$ and $G(S_j, S_i) = 1$
 11. **End**
-

The uncertainty of the graph estimate is related mainly to the size of the data set, but also to the low-dimensional approximation of the original node dissimilarities with the node location map. Furthermore, the smaller the value $|d_{\text{thr}} - d_{ij}(\mathbf{x}_i, \mathbf{x}_j)|$, the more uncertain the corresponding graph link.

If d_{thr} is selected too small for a coherent system, coherence may be lost because of the lost connections in the network model. Selecting d_{thr} too large for a system of almost independent nodes can result in false coherent behaviour.

If available, partial information of graph structure can be helpful when choosing the threshold. For example, if the total number of node connections is known d_{thr} can be chosen accordingly. If some of the nodes are known to be connected, these connections can be verified in the estimated graph, and, if missing, added. In this paper we assume the total number of network connections known and thus determining the threshold value.

There exist two limiting cases where the location map cannot be directly obtained with MDS. The first is when the network node states are highly coherent, resulting in similarity close to 1 for all node pairs. We may interpret all the nodes interacting explicitly and hence define d_{thr} large enough for all the nodes to be neighbours to one another. The second is the case when the nodes are statistically independent with similarity values near 0. Hence, we may choose $d_{\text{thr}} = 0$, resulting in a set of disconnected nodes.

2.5 Other graph structure estimation methods

Constrained-based (CB) graph estimation methods are based on conditional independence tests, where each node at a time is conditioned on subsets of the other nodes [51]. Node pairs being conditionally dependent according to the tests are concluded to be neighbours. MGMN method considers conditional independencies implicitly via constructing a spatial configuration from all the node dependency values.

The most straightforward CB method is the SGS algorithm [55], applied for both Bayes and Markov networks. SGS algorithm does not scale for large systems because its computation time grows exponentially in the number of nodes. In PC algorithm (see, e.g. [28]) conditional independence tests are only conducted for subsets of nodes that have fewer nodes than some threshold [55]. Hence, the PC algorithm scales as the network size to the power of maximum subset size and is suitable for systems consisting of some hundreds of nodes. There exist some modifications for the PC algorithm [2], and also other CB algorithms are available (see, e.g., [10,16,28,34,38]).

As an alternative to the MGMN method, we will study the Grow-Shrink (GS) algorithm [38]. GS algorithm is a CB method that was originally developed for directed graphs, but a version called the Grow-Shrink Markov Network (GSMN) is available for Markov networks [10]. There are some improvements done to the GSMN algorithm, mostly of computational efficiency but less for accuracy [10,12,23]. Other methods also include the Particle Filter Markov Network (PFMN) algorithm [11,37]. However, only the GSMN algorithm is discussed here in detail because of its simplicity, and because computational efficiency is not crucial. The GSMN algorithm used here is represented in detail in [10] where the Pearson's chi-square test is applied with the test statistics specified by the χ^2 -statistics.

In general, CB is suitable only for estimating relatively sparse graphs, because with dense graphs the subsets of nodes in the conditional dependency tests become large. Pearson's chi-square dependency test required the construction of a frequency table of size q^2 . Conditioning on a subset of W nodes, frequencies need to be calculated for q^W instances [10]. This means that the number of conditioning instances grows exponentially as a function of the number of conditioning nodes. Additionally, a large amount of data is required to get accurate results. For example, one observation for each table cell requires q^W observations when conditioning on W nodes [10]. However, the more developed versions of this algorithm [10,12,23], somewhat alleviate these problems by reducing the number of tests needed. For more details of the algorithms and their properties, see [51].

3 MRF models for networked systems

Markov Random Fields (MRFs) are stochastic models consisting of interacting components, and hence are ideal for modelling networked systems. MRF is defined through a joint probability distribution (JPD) of the network node states, which are either discrete or continuous. MRF satisfies a set of conditional independence properties defined by a graph structure, and which can thus be estimated with the methods proposed in Sect. 2.

3.1 General structure of a MRF model

The most general form of MRF JPD with a given graph is defined through a collection of cliques [8] which is a subset of graph nodes that are all neighbours to one another. A maximal clique is a clique which is not a subset of any other clique. A potential function (PF) of a clique is any positive definite function of the node states of a clique.

The most general form of JPD can be written as a product of PFs of maximal cliques. However, we will only consider exponential PFs of node pair cliques and single node cliques. Interactions of the nodes are defined by node pair cliques, whereas the local effects of nodes, e.g., due to external forces, are defined by the single node cliques.

The global structure of the MRF JPD is defined by a graph structure as a set of conditional independence properties. If two nodes are not neighbours on the graph, they are conditionally independent. The local properties are defined by the potential functions.

Let us denote the set of all node pairs by V , and let subscript $m = 1, \dots, M$ be the index for the nodes. For nodes i and j ($(i, j) \in V$) we denote the PF of a node-pair clique by $\psi_V(s_i, s_j)$, and the PF of a single node clique by $\psi(s_m)$. The MRF JPD for a vector of node state variables \mathbf{s} is now defined as

$$p(\mathbf{s}) = Z^{-1} \prod_{(i,j) \in V} \psi_V(s_i, s_j) \prod_{m=1}^M \psi(s_m), \tag{4}$$

Z is a normalisation constant, or a partition function, and is given by

$$Z = \sum_{\mathbf{s}} \prod_{(i,j) \in V} \psi_V(s_i, s_j) \prod_{m=1}^M \psi(s_m), \tag{5}$$

where the sum is taken over all combinations of node states.

Varying model types are obtained by altering the potential functions. Examples of models include the binary state Ising model [27,33], Potts model [44] which is an extension of the Ising model to arbitrary number of node states, and a Gaussian model [50].

General approach for parameter estimation is the maximum likelihood (ML) method [8]. However, ML requires Z

to be known and Z is extremely difficult to calculate in practice as the sum in (5) runs over all possible node states [60]. Instead, we apply the pseudolikelihood (PL) [6,7,46] method based on the likelihoods of conditional node probabilities.

3.2 Binary state Ising model

Ising model [27,33,41] is a binary state model with each node assuming either state -1 or $+1$. Ising model has its origin in statistical physics [52], but has also been applied e.g. in image analysis [5,59], in studying the spread of viruses [54], and for analyzing stock market crashes [58]. Although at the node level Ising model is very simple, it is phenomenologically rich for studying the collective behaviour of complex networks; see [41,46,48].

By denoting an external load of node m by h_m , the JPD of the Ising model in a form similar to (4) can be written as

$$\begin{aligned} p(\mathbf{s}) &= Z^{-1} \prod_{(i,j) \in V} \exp(J_{ij} s_i s_j) \prod_{m=1}^M \exp[H s_m (h_m - h_0)] \\ &= Z^{-1} \exp \left[\sum_{(i,j) \in V} J_{ij} s_i s_j + H \sum_{m=1}^M s_m (h_m - h_0) \right]. \end{aligned} \tag{6}$$

The first of the exponential factors is a product of PFs of node-pair cliques and the second one is a product of single node cliques. Here J_{ij} , H , and h_0 are the model parameters. In our studies we will assume that $J_{ij} = J$, uniform throughout the MRF structure.

3.3 MCMC to generate synthetic data

Markov Chain Monte Carlo (MCMC) [36] is used to generate synthetic network data to study the accuracy and limitations of the topology estimation method, and in our other studies to investigate the behaviour and sensitivity of the network in various situations.

In the literature MCMC is applied extensively in Ising model simulations (see e.g. [35,56]). We apply Gibbs sampling [8] to generate node state data distributed according to the JPD of the Ising model of (6). First an initial network state configuration is chosen randomly. Then we randomly select a single network node while fixing the states of its neighbours and select state -1 or $+1$ according to their conditional probabilities. After iterating this MCMC scheme many times the node states become distributed according to (6); see e.g. [18].

MCMC needs a “burn in” period at the beginning of the simulation during which the simulation should converge to the desired JPD to produce valid data. However, recording data from a single long simulation is not appropriate for non-ergodic processes. Although the Ising model with finite number of states is ergodic, there can be long time constants

until the entire state space is covered when running a single long simulation starting from some initial state. Hence despite the cost as increased computation time, we use an ensemble scheme where each simulation run with varying random initial state and an appropriate “burn in” period only generates a single sample. In random initialization, each node state is selected either -1 or $+1$ independently on the others. For more details, see [46].

4 Graph estimate evaluation methods

In order to evaluate a graph estimate we need measures for comparing its similarity to a true graph structure. Similarity measures are required also when the true graph structure is not known, e.g. when analyzing the difference between two graph candidates.

4.1 Frobenius scaling and procrustes analysis

As a node location map estimate resulting from MDS is unique up to translation, rotation, reflection, and scaling, comparing it to another estimate based on a different data set or to a known true location map is difficult. Procrustes transformation is a combined translation, rotation, reflection, and scaling operation. Procrustes analysis (see, e.g. [21,24,53]) searches for the best match between two node location maps by performing the Procrustes transformation on one map with respect to the other [21]. The final value of the Procrustes criterion is a measure of the similarity of the two maps.

When a network consists of tightly bound node groups, the node map may be divided into subnetworks, where the states of nodes inside the same subnetwork are tightly coupled, but nearly independent on nodes belonging to other subnetworks. There may not exist a good global match to be found with the Procrustes analysis, even if the subnetworks were identical. In particular, the scaling component goes to zero as the translation, rotation, and reflection operations are unable to provide a satisfying solution. Hence, we exclude the Procrustes scaling component and instead first scale the node coordinates of both maps with their Frobenius matrix norms [26].

We denote by \mathbf{V} and \mathbf{Q} the node coordinates ($M \times 2$ matrices) of two 2-dimensional location maps, and by size 1×2 vectors \mathbf{v}_m and \mathbf{q}_m the x - and y -coordinates of a node m . The notation for the respective Frobenius scaled node coordinates is as follows: \mathbf{V}_F , \mathbf{Q}_F , $\mathbf{v}_{F,m}$ and $\mathbf{q}_{F,m}$. The mean coordinates are denoted by size 1×2 vectors $\bar{\mathbf{v}}$ and $\bar{\mathbf{q}}$. After subtracting the mean coordinate values from the coordinates of each node, the respective location maps are denoted by \mathbf{V}_0 and \mathbf{Q}_0 . The Frobenius scaled node coordinates are now obtained as

$$\{\mathbf{V}_F, \mathbf{Q}_F\} = \left\{ \mathbf{V}_0 \left[\sum_{m=1}^M (\mathbf{v}_m - \bar{\mathbf{v}}) (\mathbf{v}_m - \bar{\mathbf{v}})^T \right]^{-\frac{1}{2}}, \right. \\ \left. \mathbf{Q}_0 \left[\sum_{m=1}^M (\mathbf{q}_m - \bar{\mathbf{q}}) (\mathbf{q}_m - \bar{\mathbf{q}})^T \right]^{-\frac{1}{2}} \right\}. \quad (7)$$

As the node coordinates are scaled and translated to the origin, Procrustes analysis reduced to finding the minimum of the sum of squared residuals (SSR) criterion [21]:

$$R_1^2 = \min_{\mathbf{Y} \in R(\theta) \times \Phi} \sum_{m=1}^M \left[\mathbf{v}_{F,m} - \mathbf{Y}(\theta, \varphi)^T \mathbf{q}_{F,m} \right]^T, \\ \left[\mathbf{v}_{F,m} - \mathbf{Y}(\theta, \varphi)^T \mathbf{q}_{F,m} \right] \quad (8)$$

where matrix \mathbf{Y} is a size 2×2 orthogonal matrix defining rotation R and a possible reflection $\varphi \in \Phi$ (Φ is the set of all possible reflections). The optimal value of SSR is obtained from a singular value decomposition of $\mathbf{Q}_F^T \mathbf{V}_F$ [21], and acts as a similarity measure of the two maps.

4.2 Node and graph distance correlation

As weak dependencies are difficult to estimate, it is highly improbable that Procrustes analysis yields a good global match between two location maps based on different data. Hence we apply a local similarity measure by comparing internode distances between matching nodes on two location maps. Let us denote by $d_{ij}^A(\mathbf{x}_i, \mathbf{x}_j)$ and $d_{ij}^B(\mathbf{x}_i, \mathbf{x}_j)$ the internode distances on two maps, A and B , with mean values \bar{d}^A and \bar{d}^B . The linear (Pearson) correlation coefficient of the distances is defined as

$$C_d(A, B) \\ = \frac{\sum_{(i,j) \in V} [d_{ij}^A(\mathbf{x}_i, \mathbf{x}_j) - \bar{d}^A] [d_{ij}^B(\mathbf{x}_i, \mathbf{x}_j) - \bar{d}^B]}{\sqrt{\sum_{(i,j) \in V} [d_{ij}^A(\mathbf{x}_i, \mathbf{x}_j) - \bar{d}^A]^2 \sum_{(i,j) \in V} [d_{ij}^B(\mathbf{x}_i, \mathbf{x}_j) - \bar{d}^B]^2}}, \quad (9)$$

where the internode distances need not be Frobenius-scaled. We call $C_d(A, B)$ as the node distance correlation (NDC). Graph distance correlation (GC) is obtained by replacing the node distances with their respective node graph distances $d_{g,ij}^A$ and $d_{g,ij}^B$.

5 Evaluation with synthetic data

In this section, the MGMN topology estimation method using non-metric MDS and SSMI is tested with data generated from the Ising model. In particular, we will study the accuracy and limitations of the method when the qualitative network behaviour is varied by changing the Ising model parameters. The effect of data characteristics on topology estimation

is studied by analysing the effects of the number of network observations, network size, node neighbourhood size, and node loading distribution. Finally the MGMN method is compared to the GSMN method.

5.1 MCMC generated synthetic data

To generate synthetic data, we will consider the reference network of 30 nodes shown in Fig. 1. The respective graph structure is obtained by first generating a node location map on a 2D plane with the x - and y -coordinates of the nodes drawn independently from a $\text{Uni}[0,1]$ -distribution and then applying a uniform threshold for the internode distances chosen so that the average number of nodes per node, A , equals 8.8. As the topology affects the qualitative properties of MRF models, the synthetic topology tries to mimic an MTN topology. However, this is difficult as in MTNs both the logical topology and that of the physical node locations are expected to affect the MTN behaviour, and hence such a combined topology is unknown.

As the interaction parameter J of the Ising model largely determines the qualitative model behaviour, in the validation tests J is varied between 0 and 0.2 with 0.01 interval, resulting in 21 model parameterisations. The other two Ising model parameters are kept constant; $H = 0.6$ and $h_0 = 0.7$. With each parameterization, a set of $L = 270$ network observations is generated. Here A and L are chosen such that they are similar to the corresponding values of a real MTN considered in our other studies; see [46].

The synthetic node state data $\{s_m^{(l)}\}_{m=1}^M$ for each observation l ($l = 1, \dots, L$) is then generated with the Gibbs sampling method from the random-field Ising model. The set of node loadings is the same for each parameterization; $h_m^{(l)}$ is i.i.d. according to $\text{Uni}[0,1]$. Each ensemble observation is then generated with a burn-in period of 500 updates to node states (in each, each node is updated once).

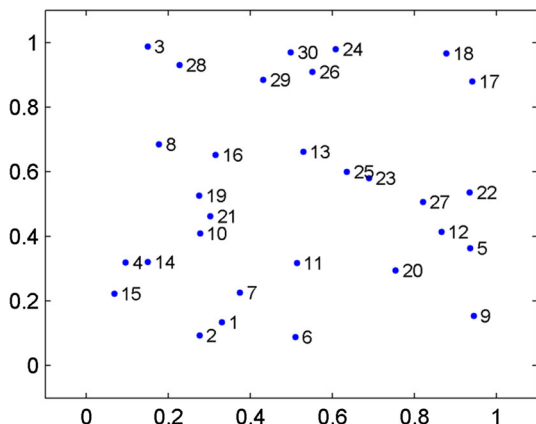


Fig. 1 Synthetic network of 30 nodes [46]

To reduce variation in the results due to the sample set and the stochastic aspects of SSMI estimation, in all studies three generated data sets are used and the results are given in medians over the three sets, unless stated otherwise. In addition, to avoid local minima, MDS is always run 20 times from varying initial node coordinates, and the node location map giving the smallest stress-1 value is chosen.

5.2 Coherence in network data

Although with the Ising model coherence is largely determined by J , also other coherence measures must be considered because various network connectivities affect the level of coherence. One overall coherence measure, abbreviated here as ASSMI is the average of the SSMI values over all node pairs. For the Ising model we can define a more specific measure for a given data set as the ratio of the interaction and the external load terms:

$$R = \frac{\sum_{k=1}^K \left| \sum_{n \in N(k)} s_n \right|}{\sum_{k=1}^K |h_k - h_0|}, \tag{10}$$

where $k = 1, \dots, K$ ($K = L \times M$) enumerates all M nodes in all L observations. Loading of node k is denoted by h_k , and $N(k)$ denotes the estimated neighbourhoods. When we use an estimated parameter value, such that $h_0 = h'_0$, the coherence measure is denoted by R' .

Figure 2 shows the relationship of the various coherence measures as a function of J , and the mean and median values of SSMI as a functions of the true graph distance of nodes. The functional relationships of the coherence measures are similar to each other; non-linear and nearly monotonic. Also the mean and median SSMI values are similar.

5.3 Location map and graph structure estimates

In all 21 model parameterisations, the estimated node location map is first scaled with the Frobenius norm and then Procrustes-transformed without the scaling component with respect to the true location map shown in Fig. 1. Node location maps can then be compared visually and analysed quantitatively with the SSR criterion. Location map estimates are shown in Fig. 3 with selected parameterisations.

Graph estimates are obtained by thresholding the node location map estimates so that $A = 8.8$. The final Procrustes SSR criterion, the node distance (NDC) and graph correlation (GC) measures with each J are given in Fig. 4. All these measures give similar information, which is that for MGMN to perform well, nodes cannot act as a group of independent nodes (very small R'), or as a single network entity (very large R'). However, the MGMN is at its best with the practical situation when the network acts coherently but not as a

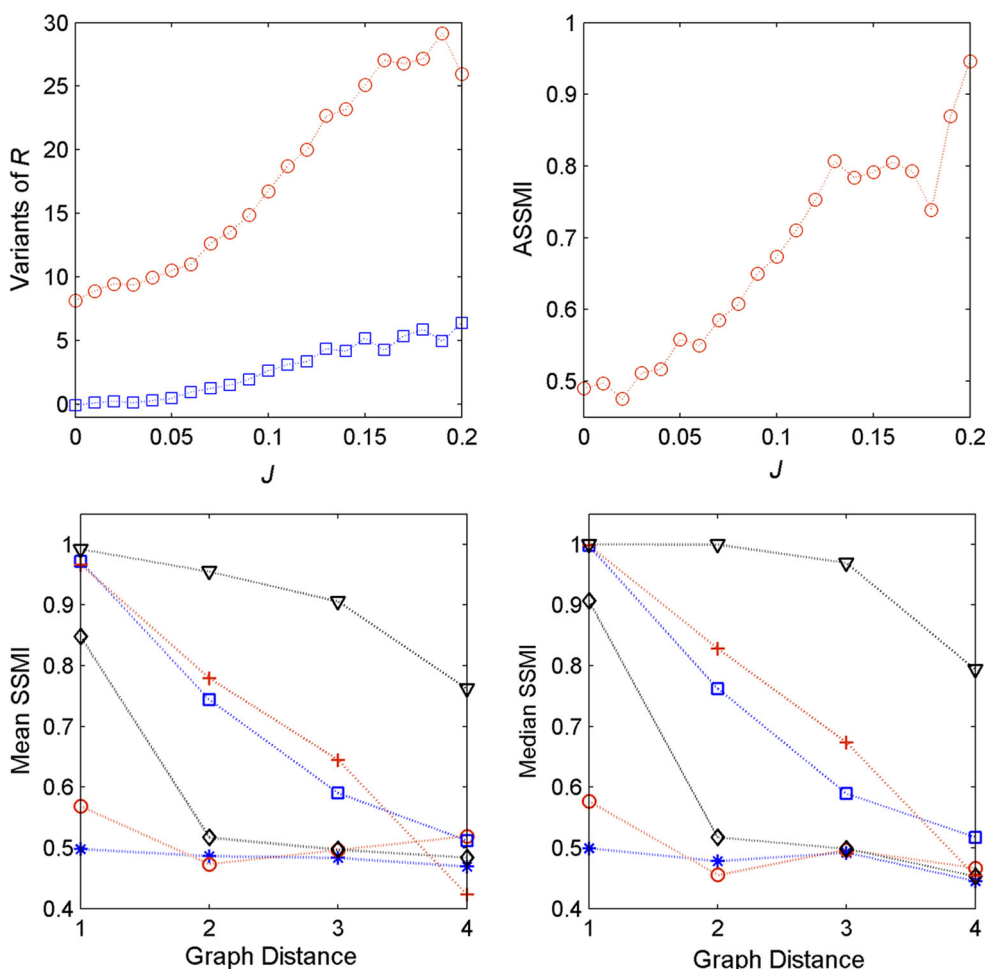


Fig. 2 Network coherence in the reference case. Top-left plot shows R' (circles) and R_N (squares) as functions of J ; the top-right plot shows ASSMI as a function of J . Mean (bottom-left) and median (bottom-right) SSMI values are given as functions of graph distance of nodes with six J values: 0.02 (asterisks), 0.04 (circles), 0.08 (diamonds), 0.12

(squares), 0.16 (plus signs), and 0.20 (triangles). Mean and median SSMI values are obtained by first taking the median SSMI over the three ensembles for each node pair with that graph distance and then taking the mean and median over those node pair SSMI values [46]

single unit. To compare the results to Fig. 2, topology estimation is most successful when the coherence of true graph neighbours most distinctly differs from coherence between non-neighbours. This is the case when the SSMI is the steepest function of the graph distance.

We have also compared the distributions of internode distances in estimated node location maps between the true graph neighbours and of all nodes, and have also studied the histograms of true graph distances of estimated graph neighbours. In both studies we have found the results to support our conclusions; see [46].

5.4 Effect of data characteristics

Here we examine the effect of data characteristics to the topology identification, in particular, the type of the node load distribution, the node neighbourhood size, the data set

size, and the network size. First, however, the quality of the synthetic data needs to be checked by changing the number of steps in the burn-in period of the MCMC. With the 30-node network the number of MCMC steps was varied from the reference case's 500×30 to 250×30 and to 1000×30 . In all three cases the results were similar, and hence the samples of the reference case are generated from the stationary distribution.

Figure 5 represents the results when three node load distribution types are studied: uniform distribution with Uni [0, 1] (the reference case), normal distribution with $N(0.5, 0.25^2)$, and exponential distribution with Exp (0.58). With the exponentially distributed loadings, the functional form of ASSMI is a bit different to others and also R' values are smaller. One explanation for the differences can be the smaller median value (0.4) with the Exp (0.58) distribution. The distribution type does not seem to affect the performance of the MGMN

Fig. 3 Estimated node location maps. From top-left to bottom-right, the maps correspond to the following J values: 0.04, 0.08, 0.16, and 0.20 [46]

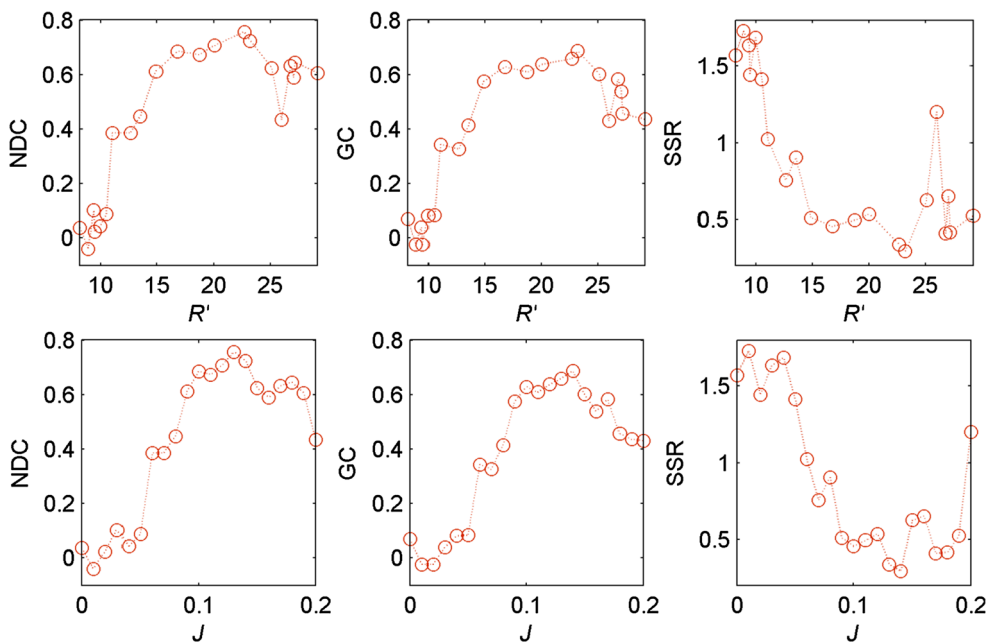
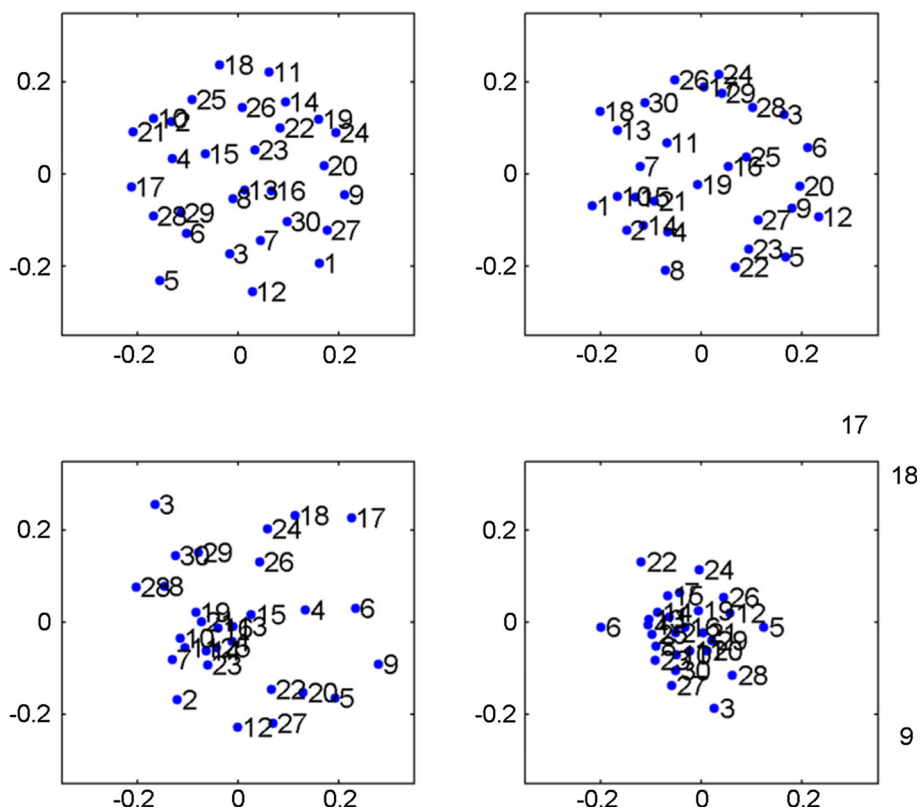


Fig. 4 Similarity measures between estimated and true node location maps and their respective graph structures. NDC, GC, and SSR are shown as functions of R' (top row) and J (bottom row) [46]

method, indicated by the large and similar graph correlation values in all cases.

The following neighbourhood sizes are tested: $A = 6.8$, $A = 8.8$ (reference), and $A = 10.8$. In the resulting figure,

the range for R' is a bit different from other figures. The larger the network connectivity is, the larger the coherence is, indicated by the large values of R' and ASSMI. With $A = 10.8$ and a few largest J values, the network acts nearly

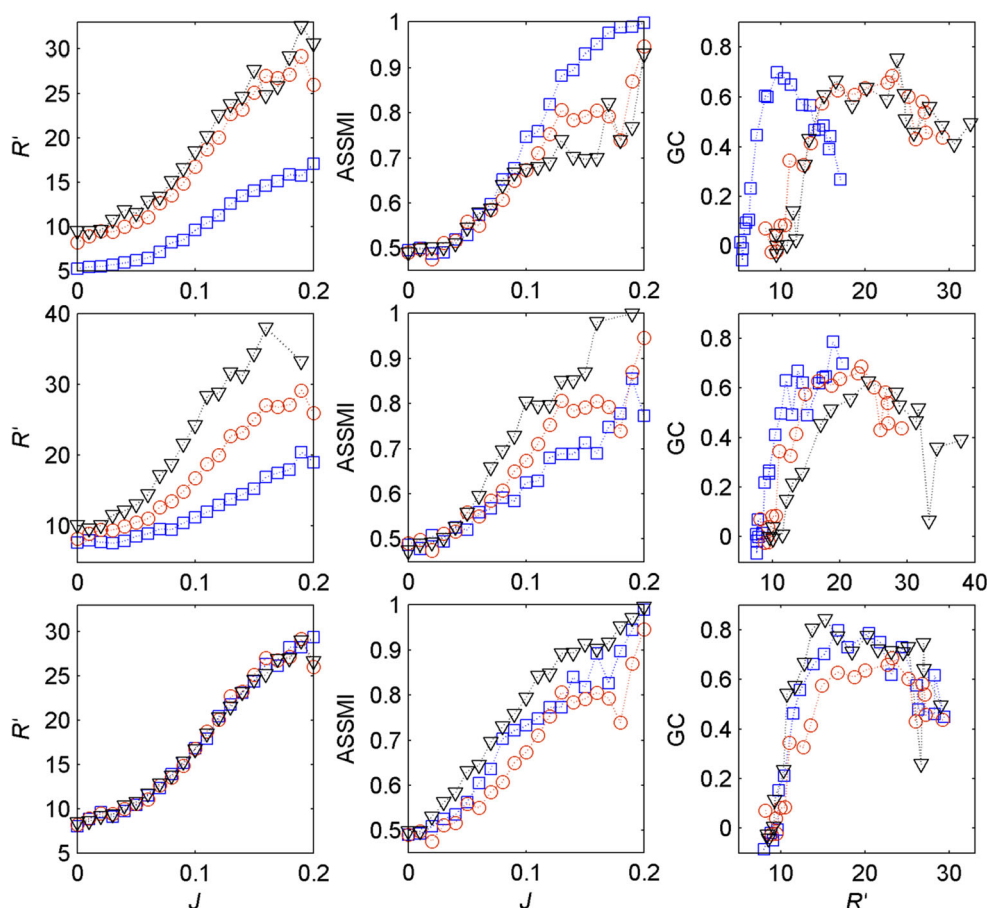


Fig. 5 Effect of load distribution type (top row), node neighbourhood size (middle row), and data set size (bottom row) on topology identification. R' (left column) and ASSMI (centre column) are shown as functions of J , and GC (right column) as a function of R' . Top row:

exponential (squares), uniform (circles), and normal (triangles) node load distributions. Middle row: $A = 6.8$ (squares), $A = 8.8$ (circles), and $A = 10.8$ (triangles). Bottom row: $L = 270$ (circles), $L = 540$ (squares), and $L = 1080$ (triangles) [46]

as a single entity with nearly all nodes appearing simultaneously in equal states in the generated data, causing problems to the topology estimation. For example, having $A = 10.8$ and $J > 0.16$, only one case at $J = 0.19$ produces a topology estimate, with the corresponding GC value at $R' \approx 33$ being clearly distinct from other values. Otherwise, topology identification is successful and small A yields somewhat better results.

The following data sizes are studied: $L = 270$ (reference) $L = 540$, and $L = 1080$. Both ASSMI and GC values are affected, but R' remains practically unchanged as changing L does not change average node states or loadings. As large data sets are more informative about node dependencies, both the ASSMI and GC increase with L ; the change in GC from $L = 270$ to $L = 540$ is particularly large. As a conclusion, the smallest data set seems too small to estimate SSMI and thereby the topology accurately.

Testing the effect of larger network sizes needs to be coupled with a simultaneous increase in the size of the data. The following network sizes are studied: $M = 30$ (reference),

$M = 60$, and $M = 120$. Three tests are done: L is constant for each M , L is increased linearly, and finally quadratically in M . The last case is tested because the number of node pairs grows quadratically in M . For the quality of the data to be the same for each network size, steps in the MCMC burn-in period are increased linearly in M ($500 \times M$). The neighborhood size is fixed to $A = 8.8$.

The results are represented in Fig. 6. With constant data size, $L = 270$, both the ASSMI and GC assume clearly larger values with the smallest M than with the two larger M values, but R' is nearly unaffected. Hence, L needs to be increased to have reasonable data quality. As L is increased linearly in M , coherence measures behave like in the previous case, but GC values are similar for all the network sizes, which indicates that the linear increase is proper.

Due to computational reasons with the case of L increasing quadratically in M , only the network with $M = 60$ is tested. Worst results are obtained when $L = 270$, however, ignoring the rather heavy fluctuations, GC is similar to results with the two larger data sets and nearly similar to the results

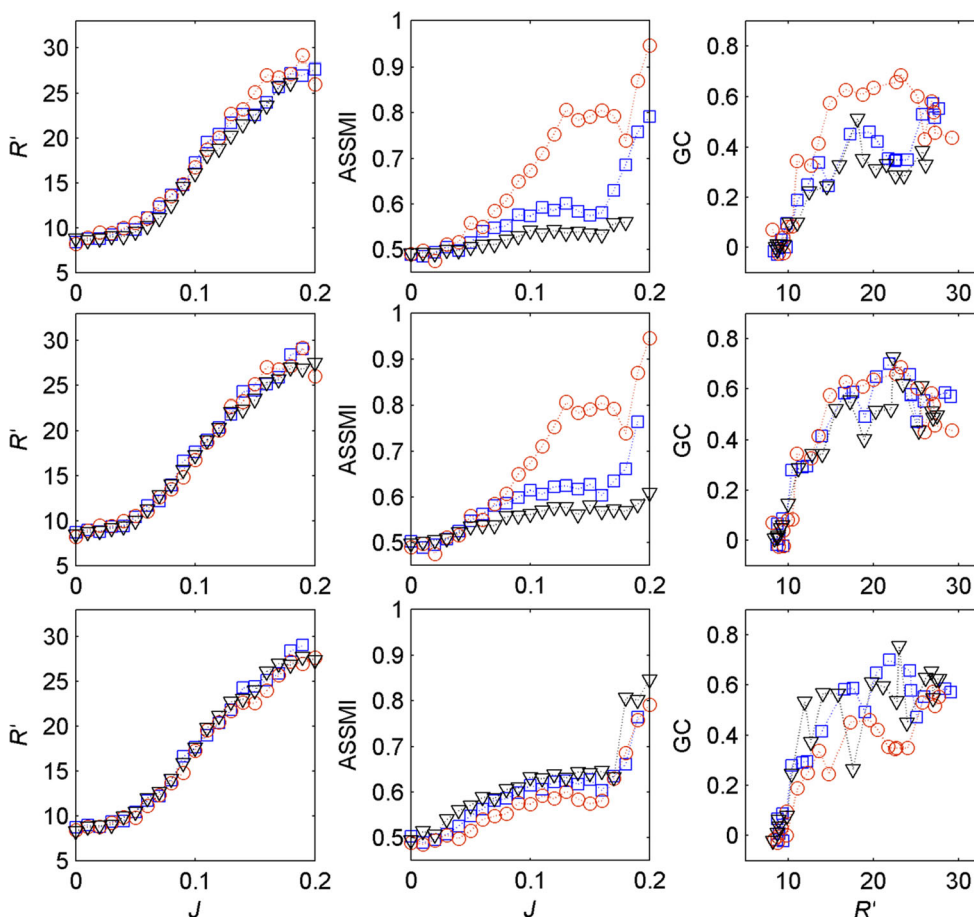


Fig. 6 Effect of network size on topology identification. R' (left column) and ASSMI (centre column) are shown as functions of J , and GC (right column) as a function of R' . Top row: $M = 30$ (circles), $M = 60$ (squares), and $M = 120$ (triangles), each with $L = 270$. Middle row: $M = 30, L = 270$ (circles), $M = 60, L = 540$ (squares),

and $M = 120, L = 1080$ (triangles). Bottom row: $L = 270$ (circles), $L = 540$ (squares), and $L = 1080$ (triangles), each with $M = 60$. For the other network sizes than $M = 30$, measures are calculated from a single ensemble [46]

with the reference case. R' is again nearly constant in all cases, which means that the data is similar in all the cases. This is not surprising, as both the graph and the parameters of the Ising model are unaltered.

ASSMI values are similar between the two larger data sizes, but smaller with $L = 270$. ASSMI seems to depend on the network size, or at least it is very different between the two larger networks with $M = 30$. The reason may be in the particular properties of the randomly generated network. Furthermore, as the network is small, even a few highly connected nodes may have large effect on the network properties.

5.5 Comparison to other methods

In this section, the MGMN method is compared to other graph estimation methods, in particular the GSMN method. As our main interest in graph estimation is the estimation of the MTN topology, the methods are tested with graphs derived from two-dimensional spatial configurations.

As a measure for comparing the graph estimates, we apply the percentage of properly recovered links (PRL) in an estimated graph. PRL is independent of A when A is the same for the compared graphs. PRL is a more exact measure than GC to compare two graphs, because GC can yield large values also in cases when graph distances, but not necessarily exact neighbourhoods, are similar. However, GC is well suited for measuring the overall fitness of an estimated graph.

Figure 7 represents the results when the MGMN method is compared to direct thresholding of MI and SSMI when varying L , A , and M ; L is here increased linearly in M . For $M = 30$, the results are similar, but with larger networks ($M = 60$ and $M = 120$), MGMN is clearly better than the other two methods as only MGMN uses the conditional dependency relations in estimation. Apparently, in this particular case with binary-state nodes, node probability distributions are so similar that the MI values between node pairs are nearly comparable, and therefore MI yields result

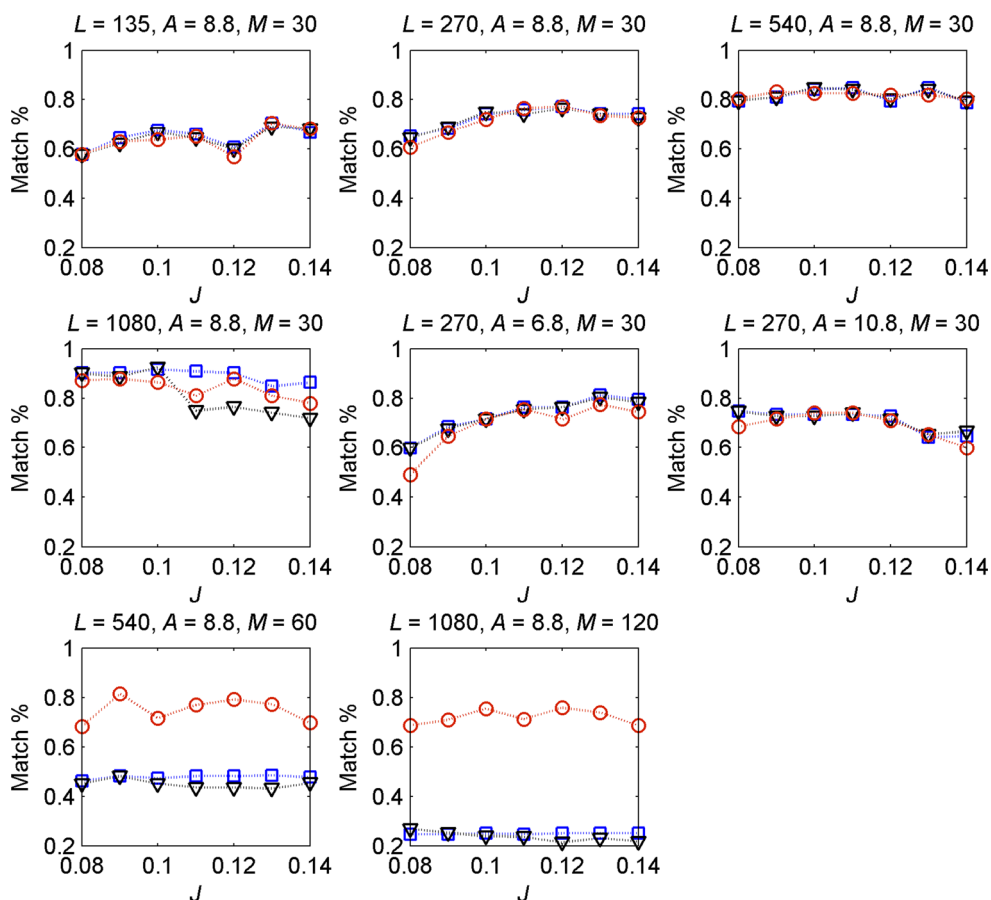


Fig. 7 Comparison of the MGMN method (circles) to direct thresholding of MI (squares) and SSMI values (triangles). Performance of the methods is tested with several data set sizes (L), neighbourhood sizes (A), and network sizes (M). In each case, results are given in percent-

ages of estimated graph links that match the true links among all true links. Results are shown for J values: 0.08, 0.10, 0.11, 0.12, 0.13, 0.14. The values with the two largest networks are based on a single ensemble [46]

similar to SSMI which represents absolute values that are comparable among different node pairs.

In the GSMN algorithm, the parameter α defines the neighbourhood size on a graph. Similar to many other constrained-based algorithms, GSMN is poor in estimating dense graphs. With the reference case having $A = 8.8$ the graph already becomes rather dense. Therefore, GSMN algorithm is also studied with α values corresponding to $A = 2.8$ and $A = 5.8$. We use mostly $L = 540$, as it yields better results than $L = 270$ as concluded previously. Figure 8 represents the results: for large α values larger intervals are used due to increased computation time.

GSMN method indeed has some difficulties in estimation of the denser graphs, which is seen as missing data points in Fig. 8. With $J = 0.08$ the results are only obtained with small A' values. With all tested J values, almost all of the estimated links are true links when A' is small, however, there are quite many true links that are missing. The rate of correct links drops with denser graphs and at $A' \sim 8.8$, MGMN clearly outperforms the GSMN method.

When varying A (2.8, 8.8, 5.8), the smaller the A' , the smaller the probability that an estimated link corresponds to a true link. However, even with the case of $A' \approx A$, GSMN gives similar but poorer results than MGMN. When varying L (270, 540, 1080), the best results are clearly obtained with the two largest data sets. With $L = 1080$, graph estimates are only obtained in a narrow range of A' values. At $A' \approx A$ and with each L , the GSMN results are not as good as with the MGMN.

Finally, M is varied and L is increased linearly in M . The following cases for (M, L) are considered: (60, 540; 30, 270; 120, 1080). With the two larger M , the estimation is only successful in a narrow range of A' values. The graph estimates are better with larger network sizes, which was not expected. This is also true for MGMN, but the variations are quite small. Overall, in the relevant cases from the application perspective of MTNs, MGMN method seems to be better for graph estimation than the GSMN method.

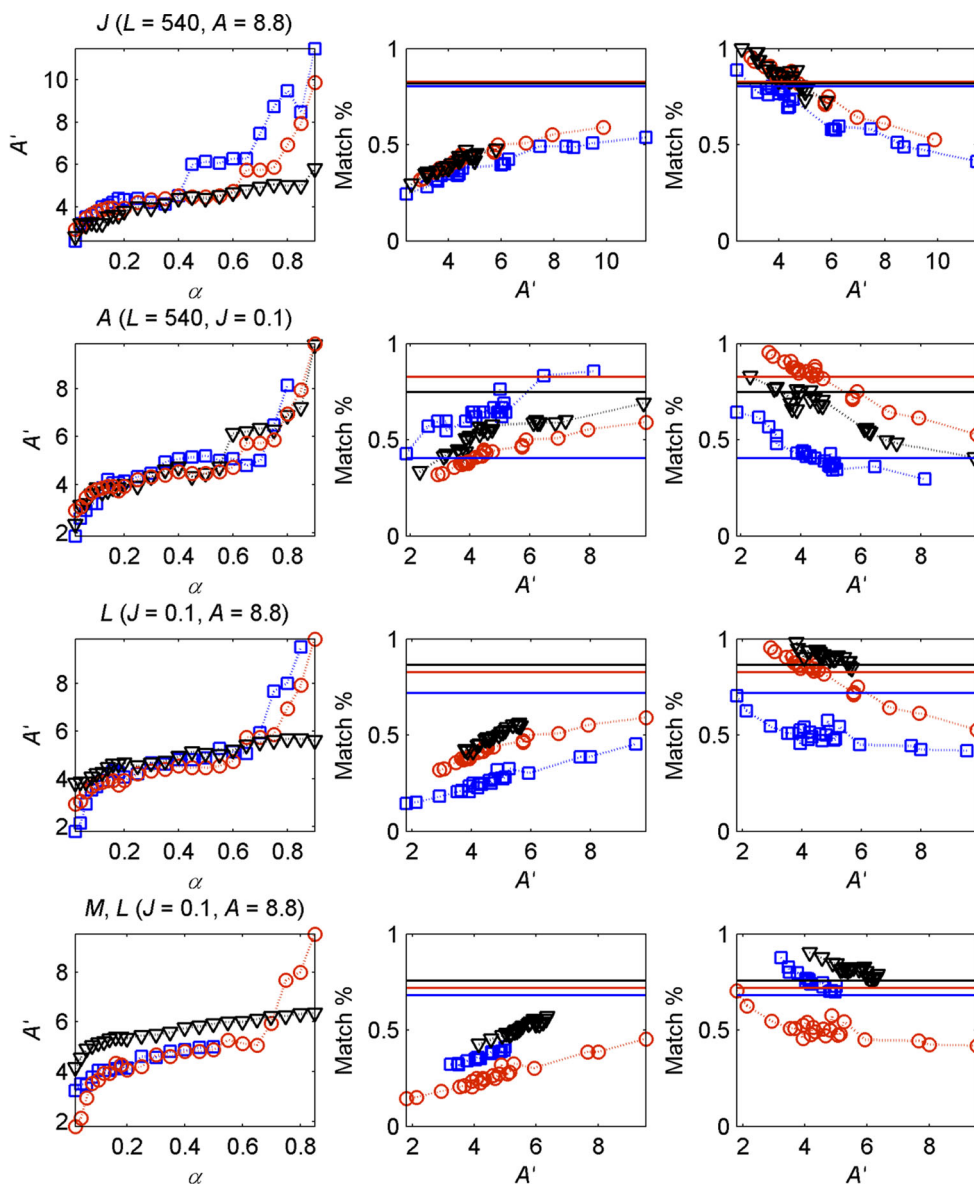


Fig. 8 Performance of the GSMN algorithm. The figure shows A' as a function of α (left column), the proportion of correctly estimated links among all true links (centre column), and the proportion of true links among all estimated links (right column). From the top to bottom row, results are given (squares; circles; triangles) for J (0.08; 0.1; 0.12), A (2.8; 8.8; 5.8), L (270; 540; 1080), and M, L (60, 540; 30, 270; 120, 1080). Results are shown with the following α values: from 0.02 to 0.2

at intervals of 0.02, and from 0.25 to 0.9 at intervals of 0.05. The two rightmost columns also show the results with the MGMN method as vertical lines in the case when $A' = A$ with the colours matching the respective cases. In the first three rows, a reference case (circles) is used ($J = 0.1, A = 8.8, L = 540, M = 30$). The results with $M = 60$ and $M = 120$ are obtained from a single ensemble [46]

6 Conclusions

In this paper we proposed a method for estimating graph structure of a networked system aimed at defining the structure of a MRF model and motivated by the mobile telecommunications networks (MTNs). The proposed estimation method was studied with several synthetic network cases corresponding to diverse network behaviour situations that are practical from the application perspective of MTNs.

The particular difficulty with MTNs is that both the logical topology and that of the physical node locations are expected to affect the behaviour of an MTN, and hence such a combined topology is unknown.

The proposed graph estimation method was found to work increasingly better as the level of coherence in the network node states was increased, except with very large coherence values when all the nodes appear in equal states and the network acts as a single entity. The type of the node load dis-

tribution was found to have only a minor effect to the graph structure estimation. The size of the data set was found essential and should be increased at least linearly as a function of the network size. In networks with large neighbourhood sizes, the estimation may be difficult since the complex node dependencies are not that well exposed in the data. Overall, the proposed graph structure estimation method works well under slightly limited, though practical, network behaviour situations and have some robustness against changing the type of node loadings, and the sizes of the network, the data set, and the node neighbourhoods.

The proposed method was compared to a constrained based graph estimation method, namely the GSMN method, and to a straightforward method based on simply thresholding the node dependency values into node neighbours. In all relevant cases from our application perspective, the proposed graph estimation method was found to give at least as good results as the GSMN method, and outperformed the straightforward method as by utilizing the conditional dependencies of the nodes when forming the graph estimate.

In our other studies the graph structure estimation method is applied together with MRF model parameter estimation methods for estimating the Ising model from a real MTN data. Identified Ising model can then be used in studying the effect of local and global node disturbance situations to the behaviour of the MTNs.

References

1. Abdallah, S. (2002). *Towards music perception by redundancy reduction and unsupervised learning in probabilistic models*. Doc. Thesis: King's College, London.
2. Abellán, J., Gómez-Olmedo, M., & Moral, S. (2006). Some variations on the PC algorithm. In *Proc third European workshop on probabilistic graphical models*.
3. Albert, R., & Barabási, A.-L. (2002). Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74, 47–97.
4. Basalaj, W. (1999). Incremental multidimensional scaling method for database visualization. In *Proc SPIE'99*, pp. 149–158.
5. Bentrem, F. W. (2010). A Q-Ising model application for linear-time image segmentation. *Central European Journal of Physics*, 8(5), 689–698.
6. Besag, J. E. (1974). Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, 36, 192–225.
7. Besag, J. E. (1975). Statistical analysis of non-lattice systems. *The Statistician*, 24, 179–195.
8. Bishop, C. M. (2006). *Pattern recognition and machine learning*. Berlin: Springer.
9. Breitbart, Y., Garofalakis, M., Jai, B., Martin, C., Rastogi, R., & Silberschatz, A. (2004). Topology discovery in heterogeneous IP networks: The NetInventory system. *IEEE/ACM Transactions on Networking*, 12(3), 401–414.
10. Bromberg, F., Margaritis, D., & Honavar, V. (2006). Efficient Markov network structure discovery using independence tests. In *Proceedings of SIAM international conference on data mining*, pp. 141–152.
11. Bromberg, F., & Margaritis, D. (2007). Efficient and robust independence-based Markov network structure discovery. In *Proc IJCAI*.
12. Bromberg, F., & Margaritis, D. (2009). Improving the reliability of causal discovery from small data sets using argumentation. *JMLR*, 10, 301–340.
13. Brown, P., Cocke, J., Della Pietra, S., Della Pietra, V., Jelinek, F., Mercer, R. (1988). A statistical approach to language translation. In *COLING-88*, Vol. 1, pp. 71–76.
14. Butte, A. J., & Kohane, I. S. (2000). Mutual information relevance networks: Functional genomic clustering using pairwise entropy measurements. In *Pacific symposium on biocomputing*, Vol. 5.
15. Chen, H., & Varshney, P. (2003). Mutual information-based CT-MR brain image registration using generalized partial volume joint histogram estimation. *IEEE Transactions on Medical Imaging*, 22(9), 1111–1119.
16. Cheng, J., Greiner, R., Kelly, J., Bell, D., & Liu, W. (2002). Learning Bayesian networks from data: An information-theory based approach. *Artificial Intelligence*, 137(1–2), 43–90.
17. Cover, T. M., & Thomas, J. A. (1991). *Elements of information theory* (pp. 5–21). Hoboken: Wiley.
18. Cressie, N. A. C. (1993). *Statistics for spatial data* (pp. 383–573). Hoboken: Wiley.
19. De Campos, L. M. (2006). A scoring function for learning Bayesian networks based on mutual information and conditional independence tests. *Journal of Machine Learning Research*, 7, 2149–2187.
20. Drees, B. L., Thorsson, V., Carter, G. W., Rives, A. W., Raymond, M. Z., Avila-Campillo, I., et al. (2005). Derivation of genetic interaction networks from quantitative phenotype data. *Genome Biology*, 6, R38.
21. Everitt, B. S., & Rabe-Hesketh, S. (1997). *Kendall's library of statistics 4: The analysis of proximity data* (pp. 11–68). London: Arnold.
22. Friedman, N., & Koller, D. (2003). Being Bayesian about network structure: A Bayesian approach to structure discovery in Bayesian networks. *Machine Learning*, 50, 95–126.
23. Gandhi, P., Bromberg, F., & Margaritis, D. (2008). Learning markov network structure using few independence tests. In *Proceedings of SIAM international conference on data mining*, pp. 680–691.
24. Gower, J. C. (1975). Generalized procrustes analysis. *Psychometrika*, 40(1), 33–51.
25. Hellebrandt, M., Mathar, R., & Scheibenbogen, M. (1997). Estimating position and velocity of mobiles in a cellular radio network. *IEEE Transactions on Vehicular Technology*, 46(1), 65–71.
26. Horn, R. A., & Johnson, C. R. (1990). *Norms for vectors and matrices. Matrix analysis* (Ch. 5). Cambridge: Cambridge University Press.
27. Ising, E. (1925). Beitrag zur Theorie des Ferromagnetismus. *Zeitschrift für Physik*, 31, 253–258.
28. Kalisch, M., & Bühlmann, P. (2007). Robustification of the PC algorithm for directed acyclic graphs. *Journal of Computational and Graphical Statistics*, 17(4), 773–789.
29. Kishino, H., & Waddell, P. J. (2000). Correspondence analysis of genes and tissue types and finding genetic links from microarray data. *Genome Informatics*, 11, 83–95.
30. Kruskal, J. B. (1964). Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29, 1–27.
31. Kruskal, J. B. (1964). Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, 29, 115–129.
32. Lee, S.-I., Ganapathii, V., & Koller, D. (2007). Efficient structure learning of Markov networks using L_1 -regularization. In *Advances in neural information processing systems*.
33. Lenz, W. (1920). Beitrag zum Verständnis der magnetischen Erscheinungen in festen Körpern. *Zeitschrift für Physik*, 21, 613–615.

34. Li, F. (2007). *Structure learning with large sparse undirected graphs and its applications*. Doc. Thesis, Carnegie Mellon University, USA.
35. Lo, W. S., & Pelcovits, R. A. (1990). Ising model in a time-dependent magnetic field. *Physical Review A*, 42(12), 7471–7474.
36. MacKay, D. (2003). *Information theory, inference and learning algorithms* (pp. 357–421). Cambridge: Cambridge University Press.
37. Margaritis, D., & Bromberg, F. (2009). Efficient Markov network discovery using particle filter. *Computational Intelligence*, 25(4), 367–394.
38. Margaritis, D., & Thrun, S. (1999). Bayesian network induction via local neighbourhoods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20, 832–834.
39. Martín-Merino, M., & Muñoz, A. (2004). A new MDS algorithm for textual data analysis. In *Proc ICONIP'04. LNCS*, Vol. 3316, pp. 860–867.
40. Mathiassen, J., Skavhaug, A., & Bø, K. (2002). Texture similarity measure using Kullback–Leibler divergence between Gamma distributions. In A. Heyden et al. (Eds.), *ECCV 2002 Part III, LNCS*, Vol. 2352, pp. 133–147.
41. McCoy, B. M., & Wu, T. T. (2014). *The two-dimensional Ising model*. Courier Corporation.
42. Newman, M. E. J. (2003). The structure and function of complex networks. *SIAM Review*, 45(2), 167–256.
43. Niu, C., & Grimson, E. (2006). Recovering non-overlapping network topology using far-field vehicle tracking data. In *Proc ICPR'06*, pp. 944–949.
44. Potts, R. B. (1952). Some generalized order-disorder transformations. *Proceedings of the Cambridge Philosophical Society*, 48, 106–109.
45. Press, W., Flannery, B., Teukolsky, S., & Vetterling, W. (1986). *Numerical recipes: The art of scientific computation* (pp. 476–481). Cambridge: Cambridge University Press.
46. Rajala, M. (2009). *Data-based modelling and analysis of coherent networked systems with applications to mobile telecommunication networks*. Doc. Thesis, Tampere University of Technology, Finland. [http://dspace.cc.tut.fi/dpub/handle/123456789/6056\(24.3.2016\)](http://dspace.cc.tut.fi/dpub/handle/123456789/6056(24.3.2016)).
47. Rajala, M., & Ritala, R. (2006). Mutual information and multidimensional scaling as means to reconstruct network topology. In *Proc SICE-ICCAS'06*, pp. 1398–1403.
48. Rajala, M., & Ritala, R. (2006). Statistical model describing networked systems phenomena. In *Proc ISCC'06*, pp. 647–654.
49. Rajala, M., & Ritala, R. (2007). A method to estimate the graph structure for a large MRF model. In J. Marques de Sá et al. (Eds.), *ICANN 2007 Part II, LNCS*, Vol. 4669, pp. 836–849.
50. Rue, H., & Held, L. (2005). *Gaussian Markov random fields: Theory and applications*. Boca Raton: CRC.
51. Schlüter, F. (2014). A survey on independence-based Markov networks learning. *Artificial Intelligence Review*, 42(4), 1069–1093.
52. Schroeder, D.V. (1999). *An introduction to thermal physics*. Reading: Addison-Wesley.
53. Seber, G. (1984). *Multivariate observations* (pp. 235–256). Hoboken: Wiley.
54. Solé, R. V., Ferrer, R., González-García, I., Quer, J., & Domingo, E. (1999). Red queen dynamics, competition and critical points in a model of RNA virus quasispecies. *Journal of Theoretical Biology*, 198, 47–59.
55. Spirtes, P., Glymour, C., & Scheines, R. (2000). *Causation, prediction, and search*. Cambridge, MA: The MIT Press.
56. Szabó, G., & Kádár, G. (1998). Magnetic hysteresis in an Ising-like dipole–dipole model. *Physical Review B*, 58(9), 5584–5587.
57. Thompson, C. J. (1972). *Mathematical statistical mechanics*. Princeton: Princeton University Press.
58. Vanderwalle, N., Boveroux, P., Minguet, A., & Ausloos, M. (1998). The crash of October 1987 seen as a phase transition: Amplitude and universality. *Physica A*, 255, 201–210.
59. Winkler, G. (2003). *Image analysis, random fields and Markov chain Monte Carlo methods* (2nd ed.). Berlin: Springer.
60. Yang, C. N. (1952). The spontaneous magnetization of a two-dimensional Ising model. *Physical Review*, 85(5), 808–816.
61. Young, F. W. (2013). *Multidimensional scaling: History, theory, and applications*. Hove: Psychology Press.



Miika Rajala received his Master's degree in information technology in 2004 and Doctoral degree in measurement information technology in 2009 from Tampere University of Technology. He is currently working as a Data analytics specialist at Metsä Group.



Risto Ritala received his Master's and Doctoral degree in engineering physics from Helsinki University of Technology in 1981 and 1986, respectively. Since 2003 he has been a professor in measurement information technology at the Laboratory of Automation and Hydraulics at Tampere University of Technology. His research interests are optimization of the operation of sensing degrees of freedom in autonomous mobile robots and related estimation, prediction, data analysis, modelling and optimal control.