TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY
Julkaisu 799 • Publication 799

Miika Rajala

Data-Based Modelling and Analysis of Coherent Networked Systems with Applications to Mobile Telecommunications Networks

Tampere 2009

Tampereen teknillinen yliopisto. Julkaisu 799
Tampere University of Technology. Publication 799

Miika Rajala

Data-Based Modelling and Analysis of Coherent Networked Systems with Applications to Mobile Telecommunications Networks

Thesis for the degree of Doctor of Technology to be presented with due permission for public examination and criticism in Sähkötalo Building, Auditorium S3, at Tampere University of Technology, on the 21st of April 2009, at 12 noon.

Tampereen teknillinen yliopisto - Tampere University of Technology
Tampere 2009

ISBN 978-952-15-2131-7 (printed)
ISBN 978-952-15-2236-9 (PDF)
ISSN 1459-2045

Abstract

Due to interactions of nodes, a networked system may behave coherently with the nodes collaborating to perform a joint task, such as transferring speech and data traffic in a mobile telecommunications network (MTN). On one hand, coherence may lead to high network performance and an ability to withstand high external node loadings caused by the traffic. On the other hand, however, because of coherence, a network may exhibit such qualitative properties that lead to a drastic sensitivity of the network to even smallest node loading changes under a high loading. At worst, such sensitivity may lead to a coherent collapse of the entire network and thereby to severe financial and quality-of-service related losses. This thesis examines interacting coherent networks with a focus on MTNs and aims to provide information through statistical modelling for supporting the decision making of network operators and helping to improve the network robustness against unexpected node failures or excessive local node loadings.

In this thesis, Markov random fields (MRFs) are applied as statistical collaborative models of networked systems, describing a network through the joint probability of network node states. Instead of trying to model the exact dynamics of complex networked systems, which is extremely difficult, MRF models provide all the uncertainty information related to the network state under given conditions. Ising model, a simple MRF model, is adopted to model the joint probability of network node states under given external node loadings. Though it describes nodes in binary states, the Ising model is highly capable of describing the behaviour of complex coherent networks and thus proper, e.g., to study the qualitative properties of MTNs.

As an MRF model, identification of the Ising model consists of two parts: identification of the graph structure of the nodes and identification of the model parameters. Graph structure defines the neighbourhood relations of the nodes, and after the graph is fixed, the model parameters then determine the coherence of the network. Both the graph structure and the model parameters hence contribute to the qualitative network properties. In this thesis, for graph structure identification, a method especially suitable for systems assuming an underlying spatial node configuration, such as MTNs, is developed. However, the method is likely to be relevant to other applications as well and, more generally, serve as a means of learning a graph structure for MRF models. For parameter identification, an approximation of the maximum likelihood method, the pseudolikelihood method, is applied. The uncertainties of the parameter estimates are then studied via approximating the parameter distribution, under the pseudolikelihood assumption, with a Gaussian distribution.

To study the qualitative properties of an identified Ising model, Markov chain Monte Carlo (MCMC) methods are used for model simulation under varying uniform node loading and local excessive node loading situations. Moreover, MCMC simulations are applied for studying the adiabatic and transient dynamics of the Ising model. By using the MCMC methods for generating samples from an identified Ising model, the identification methods are tested extensively with MCMC-generated synthetic data in varying qualitative network behaviour cases. By using real MTN data, the model identification methods are finally tested in a realistic case.

Acknowledgements

I wish to express my deepest gratitude for the supervisor of this work, Prof. Risto Ritala, for his guidance and for providing many ideas to this research work. I would also like to thank the reviewers of this work, Prof. Leo Van Biesen and Dr. Kimmo Raivio, for their valuable feedback. The financial support of Tampere University of Technology Graduate School is also acknowledged.

I am also grateful to the staff of the Department of Automation Science and Engineering of Tampere University of Technology, especially Prof. Jouko Halttunen, Mr. Heimo Ihalainen and Mr. Pekka Kumpulainen. Thanks are also due to Dr. Timo Lepistö for revising the English. Finally, I would like to thank my family and friends for their support during this work.

Table of Contents

PART I: INTRODUCTION TO NETWORKED SYSTEMS AND APPLICATION

PART II: METHODOLOGICAL BACKGROUND

PART III: DEVELOPMENT AND EVALUATION OF MODEL IDENTIFICATION METHODS

PART IV: APPLICATIONS TO MOBILE TELECOMMUNICATIONS NETWORKS

List of Abbreviations

BSC	base station controller
BTS	base transceiver station
CSS	chi-square statistics (χ^2-statistics)
GC	graph (distance) correlation
GMRF	Gaussian Markov random field
GSM	global system for mobile communications
GSMN	grow-shrink Markov network
JPD	joint probability distribution
JSD	Jensen-Shannon divergence
KLD	Kullback-Leibler divergence
KPI	key performance indicator
KT	Kendall's tau
MAP	maximum a posteriori
MCMC	Markov chain Monte Carlo
MDS	multidimensional scaling
MGMN	MDS-based graph estimation for Markov networks
MH	Metropolis-Hastings
MI	mutual information
ML	maximum likelihood
MPL	maximum pseudolikelihood
MRF	Markov random field
MTN	mobile telecommunications network
NDC	node distance correlation
SSCSS	statistical significance of CSS
SSMI	statistical significance of MI
SR	Spearman's rho
SSR	sum of squared residuals

List of Symbols

A	average number of node neighbours
A'	average number of node neighbours on an estimated graph
$A(\mathbf{s}^{(l+1)}, \mathbf{s}^{(l)})$	in MCMC, the approval probability of $\mathbf{s}^{(l+1)}$ when $\mathbf{s}^{(l)}$ is given
α	significance level in statistical dependency tests
β	parameter related to the Boltzmann distribution: $\beta = 1/kT$
$\mathbf{b}$	Procrustes translation component
c	Procrustes scaling component
$\mathbf{C}$	covariance matrix of Gaussian probability distribution
D	degrees of freedom parameter
d_{ij}	Euclidean distance between nodes i and j
$\hat{d}_{ij}$	in MDS, the estimated distance, or disparity, between nodes i and j
d_{thr}	in MGMN, the internode distance threshold
d_{ij}^{A}	distance between nodes i and j on a node location map A
$d_{g,ij}^{A}$	graph distance between nodes i and j on a graph A
δ	Kronecker delta function
δ_{ij}, $\delta(S_i, S_j)$	dissimilarity between nodes (random variables) i (S_i) and j (S_j)
∇	gradient (first-order derivative)
∇^2	Hessian (second-order derivative)
E_p	expectation value with respect to probability distribution p
ε_{ij}	in MDS, the error in the description of δ_{ij} based on d_{ij}
$\psi(s_m)$	in MRF, the single node clique potential function for s_m
$\psi_V(s_i, s_j)$	in MRF, the node pair clique potential function for s_i and s_j
H	external load parameter of the Ising model
H'	estimate of H
$H(S_i)$	entropy of random variable S_i
$H(S_i, S_j)$	joint entropy of random variables S_i and S_j
$H(S_i \vert S_j)$	conditional entropy of random variable S_i given S_j
$H[p(s_i), q(s_i)]$	cross entropy of probability distributions $p(s_i)$ and $q(s_i)$ associated with S_i
h_0	load threshold parameter of the Ising model
$h_0{}'$	estimate of h_0
h_m	external load of node m
$h_m^{(l)}$	external load of node m at observation l
h_m^{eff}	effective load of node m
$I(S_i; S_j)$	mutual information between random variables S_i and S_j
J	interaction parameter of the Ising model
J'	estimate of J
k	constant related to the Boltzmann distribution
L	number of observations
$L(\boldsymbol{\theta} \vert \{\mathbf{s}^{(l)}\}_{l=1}^{L})$	likelihood of the parameters $\boldsymbol{\theta}$ given the data
L_p	pseudolikelihood of the parameters $\boldsymbol{\theta}$ given the data

$L_{p,\log}$	log-pseudolikelihood of the parameters $\boldsymbol{\theta}$ given the data
M	number of nodes
N	number of neighbours
N_C, N_{C2}, N_{C3}	normalisation constants of probability distributions
$N(m)$	node neighbourhood of node m
$p(\mathbf{s})$	probability associated with state $\mathbf{s}$
$\tilde{p}(\mathbf{s})$	$= p(\mathbf{s})/Z$
$\mathbf{Q}$	precision matrix of Gaussian probability distribution
$q(\mathbf{s})$	probability associated with state $\mathbf{s}$
$q(\mathbf{s}^{(l+1)}\|\mathbf{s}^{(l)})$	in MCMC, the proposal distribution, the probability of $\mathbf{s}^{(l+1)}$ given $\mathbf{s}^{(l)}$
R	term ratio related to the Ising model
R'	term ratio using estimated parameters
R_N	normalised term ratio using estimated parameters
S_m	random variable associated to node m
s_m	state realisation of random variable S_m
$\mathbf{S}$	multivariate random variable (associated to a group of nodes)
$\mathbf{s}$	state realisation of random variable $\mathbf{S}$
$\mathbf{s}_{-m}$	states of nodes other than node m
$s_m^{(l)}$	state of node m at observation l
$\langle \mathrm{s} \rangle$	average over the network node states in an observation
$s_{\mathrm{c}}(m,l)$	in MTN data, a continuous valued state of node m at observation l
σ	disorder parameter in random-field Ising model, or standard deviation of a univariate Gaussian
σ_{ij}, $\sigma(S_i, S_j)$	similarity between nodes (random variables) i (S_i) and j (S_j)
T	temperature parameter related to the Ising model
$\boldsymbol{\theta}$	vector of model parameters
U	set of nodes
V	set of node pairs
$\mathbf{V}$, $\mathbf{Q}$	node coordinate matrices related to Procrustes transformation
$\mathbf{V}_{\mathrm{F}}$, $\mathbf{Q}_{\mathrm{F}}$	Frobenius normalised versions of matrices $\mathbf{V}$ and $\mathbf{Q}$
$\mathbf{V}_0$, $\mathbf{Q}_0$	matrices $\mathbf{V}$, $\mathbf{Q}$ with means removed from each column
$\mathbf{v}_m$, $\mathbf{q}_m$	vectors of coordinates of node m in $\mathbf{V}$ and $\mathbf{Q}$
$\bar{\mathbf{v}}$, $\bar{\mathbf{q}}$	vectors of mean values of $\mathbf{V}$ and $\mathbf{Q}$
$\mathbf{x}_i$	location of node m as a coordinate vector in a q-dimensional node location map
$\mathbf{Y}$	Procrustes rotation/reflection component
Z	normalisation term, or partition function, of MRF joint distribution
Z_m	normalisation term, or partition function, of conditional distribution of node m

1. Introduction

A networked system consists of a group of nodes with each node associated with a state and an external force, or loading, affecting the node state. Nodes interact through a set of neighbour relations defined by a network topology. Topological relations may also be associated with an interaction strength specifying how strongly neighbouring nodes are bound to each other. Topology being fixed, with weak interactions external force mostly determines a node state, whereas with strong interactions the node state is largely affected by the states of its neighbouring nodes. Consequently, under strong interactions and strongly connected topology, network becomes coherent with the states of neighbouring nodes showing correlations. Furthermore, the stronger the interactions and the connectivity, the larger the coherence and the correlations throughout the network.

Because of the collective node behaviour, in a coherent network complex qualitative system properties may emerge (e.g., [42]). For example, the otherwise smooth transitions under varying external forces from one extreme network state (all nodes assuming the same state) into another may occur discontinuously with all the nodes transiting simultaneously. Furthermore, due to hysteresis, the time-reversal symmetry of the system may be broken with the network state depending on its past values (e.g., [37], [111], [128]). In a real network, discontinuous transition may correspond to a sudden coherent collapse of the entire network under heavy external loading. Because of hysteresis, the original, desired, system state may not be obtained simply as by reducing the loading back to its previous value, but instead, needs to be severely reduced, and the larger the coherence, the larger the loading reduction needed.

Under varying external loadings, systemic analysis of the collective node behaviour may thus provide valuable information about the qualitative properties of coherent networked systems, such as mobile telecommunications networks (MTNs), the application studied in this thesis. MTNs enable wireless communication via mobile phones as speech or text messages, but today they also provide other services, e.g., mobile broadband Internet. An MTN consists of base station cells, or nodes, spread geographically to enable mobile communication in varying physical locations. Hence, like in many other physical systems, MTN cell nodes have a spatial configuration in a two- (or three) dimensional space, and the topological relations then depend on the physical distances between the nodes in that space. In fact, an MTN assumes topology information due to both the physical internode distances and a set of specified logical node relations, through which the cell nodes co-operate. The joint impact of the two pieces of topology information is then manifested in the coherent behaviour of the MTN.

As the phenomena in a networked system stem mostly from systemic node interactions, details of node structures and dynamics are less important. Indeed, many networks, which at first glance may appear very different, share similar qualitative properties; this is called universality in physics [166] and catastrophe theory in mathematics [112]. The universality of phenomena has been rigorously shown, e.g., in first- (discontinuous) and second- (continuous) order phase transitions of ferromagnets [166]. In particular, in technical networked systems, such as MTNs, transitions from one system (desired) state into another (undesired) have similarities to those occurring in ferromagnets. Indeed, a network collapse may show similarities to discontinuous transitions in ferromagnets. Also in many other technical networks similar behaviour have been reported; e.g., in many communications networks (see, e.g., [73], [110], [153]) and in power-grids as blackouts (see, e.g., [11], [40], [85], [104], [125]). Hence, methods developed to study the coherent behaviour of one system may prove to be well suited to other similar systems as well.

The Ising model, or Lentz-Ising model, [66], [89] (see also [22], [108]) is an example of a simple statistical network model, which was originally developed to analyse phase transitions in ferromagnets [166], but as some phenomena in the Ising model are universal, it has been widely applied, e.g., in image analysis [161]. As a statistical model, the Ising model assigns the joint probability of network node states, which are binary-valued, given the external forces affecting the node states. The Ising model belongs to a group of models called Markov random fields (MRFs) or Markov networks [70]. They are statistical models with their joint probability distribution factorising according to network topology or an undirected graph [18]. Thus, the graph depicting the link connections of a networked system defines the structure of an MRF model. Another well-known class of statistical graph models, the Bayesian networks [105], exploits a graph with directed links.

Identification of the Markov or the Bayesian network model consists of identification of graph structure and model parameters. Because a networked system may exhibit various types of qualitative behaviour, success in model identification depends on which type of network behaviour the identification data set represents. In the literature, parameter identification has been studied extensively for both MRF models (see, e.g., [16], [17]) and Bayesian networks (see, e.g., [105]). Graph identification studies have concentrated on the Bayesian networks (see, e.g., [31], [32], [36], [51], [94], [95], [105]), though some methods developed for Markov networks have been presented as well (see, e.g., [23], [82]). Once a statistical model has been identified for a networked system, the model can be studied under varying external forces with Markov Chain Monte Carlo (MCMC) simulations [55].

1.1 Research Problem

The dynamic behaviour of a networked system depends on the complex collective behaviour of the network nodes, where even the slightest change in external conditions, caused, e.g., by a random fluctuation, may have a drastic effect on the overall network behaviour. Because of system complexity, the exact dynamic behaviour of a network is extremely difficult, if not impossible, to predict. Instead of trying to estimate exactly a network state under certain conditions, statistical modelling of the network state assigns a probability to the node states, and thereby covers all the uncertainty information related to the network state. Because under certain critical conditions a networked system may manifest itself in two drastically different states, it is essential to understand the uncertainties related to the system state and to take action to reduce the risks, or consequences, of a possible undesirable state.

This thesis examines statistical modelling of networked systems with applications to MTNs and addresses the following research problems:

- How to model statistically technical networked systems, such as MTNs, under conditions of interest so that the model is able to reproduce the typical complex behaviour of networks, such as discontinuous phase transitions and hysteresis?
- How to identify a statistical model from a node state–load measurement?
  - How to identify model structure to capture topological node interconnections?
  - How to identify model parameters to capture qualitative system properties?
- How to evaluate with synthetic and real network data the performance of model identification methods and the quality of the models obtained?
- By what means to study qualitative model behaviour under various instances of network behaviour, in particular the effect of changes in external loadings on collective network state behaviour?

1.2 Hypothesis

In modern technical networked systems, such as MTNs, statistical models can be used to study network state behaviour to obtain statistical information about the uncertainty of the network state. The state probability information obtained may then be used in planning a network and to render it more robust against sudden node failures or unexpected excessive local node loadings, and thereby to improve its quality of service to its users. At best, in MTNs, information obtained through statistical modelling will help network operators in their decision-making. Taking action accordingly, operators can prevent or at least minimise the risk of a disastrous network collapse, a coherently behaving network from suddenly transiting from a desired into an undesired state. Serious financial and quality-of-service-related losses could thus be avoided.

1.3 Limitations

This thesis focuses on statistical modelling of the overall behaviour of networked systems, in particular of that of MTNs. Consequently, the following topics are not covered:

- Other approaches to modelling such as accurate modelling of network dynamics; yet an effort is made to study network dynamics by MCMC simulation of the statistical model
- Detailed statistical/dynamic modelling of individual network nodes; instead a simple binary description of each node is assumed with the aim to model their collective behaviour
- Detailed description of using statistical methodology in network planning and improving MTN performance; only methods to model such purposes are provided and their general use and performance described—their practical application is left to the network operator, who is better versed in network-specific concepts
- Detailed description of MTNs and their operation; MTNs serve to test the methods, but this study does not aim at improving their performance

1.4 Contribution

The main contributions of this thesis, already partly published in [115], [116], [117], [118], and [119], are as follows:

- Adaptation of statistical models developed and studied extensively in other disciplines, such as physics, for analysis of technical networked systems, particularly MTNs
    - Models are promising in describing the type of phenomena that exist in MTNs and in many other technical networked systems
    - Defining the joint probability of network node states as a function of external conditions affecting the network supports network operators in their decisions on actions to the network, e.g., to modify the network's response under certain heavy external loading situations
    - The joint probability of network node states provides all the uncertainty information of a network state and thereby helps in managing risks related to network operation
- Development of a topology identification method for networked systems, which can be applied to graph structure identification of MRF models
    - The method is suitable at least for systems such as MTNs, which assume an underlying spatial node configuration
- Extensive testing of the developed graph structure identification method and the parameter identification method with synthetic network data under varying qualitative network behaviour situations and with real MTN data
    - The identification works efficiently in practical cases, in which a networked system is neither minimally nor extensively coherent
- Study of the networked system's sensitivity to local and global changes in external node loading by simulating the statistical model with MCMC methods
    - Expected phase transitions do occur, such as network collapse under certain conditions
- MCMC-simulations-based study of both transient and adiabatic network dynamics under changing external node loading
    - MCMC simulations are well suited candidates for studying some dynamic properties of statistical models

In this study, the author's contribution is as follows. The author studied statistical models suitable for analysis of technical networked systems, especially of MTNs, and chose the Ising model for extensive study. The author participated in developing model identification methods, designed the necessary software, and carried out all the numerical method evaluations with synthetic and real network data. The author also contributed to the study of networked systems' sensitivity to changes in external node loading and to the MCMC-simulations-based study of network dynamics. The author performed all the sensitivity tests and simulations.

1.5 Structure

This thesis is divided into four parts. The first part consists of Chapters 2–3 and is an introduction to MTNs (Chapter 2) and in general to networked systems and their properties (Chapter 3). The second part contains Chapters 4–7, which provide a background to the methods applied in the thesis. This part introduces MRF models and their properties for modelling networked systems (Chapter 4), node dependency measures and other related methods for identifying MRF graph structures (Chapter 5), parameter identification methods for MRF models (Chapter 6), and MCMC methods for simulating MRF models (Chapter 7). The third part, through Chapters 8–10, develops and extensively tests the topology (Chapter 8) and parameter (Chapter 9) identification methods with simulated synthetic network data. Qualitative model behaviour is then studied under varying external loadings in Chapter 10. Finally, the fourth part, Chapters 11–14, introduces and preprocesses real MTN data (Chapter 11) and then applies the identification methods to this data (Chapters 12–13) and MCMC methods to the obtained models (Chapter 14).

2. Mobile Telecommunications Networks

Today's mobile telecommunications networks (MTNs) not only enable wireless communication via mobile phones as speech or text messages but also provide many other services, such as mobile broadband access to the Internet. MTNs have rapidly developed from first-generation analogue mobile networks, such as the Nordic Mobile Telephone (NMT) standard, into second-generation (2G) digital mobile networks, e.g., the Global System for Mobile communications (GSM) standard. Although 2G networks are widely used today, third-generation mobile networks (3G) have become popular in providing more data bandwidth, and even fourth-generation networks are already on their way (see, e.g., [171], [172]).

This thesis focuses mainly on GSM networks, simply because data was available from an anonymous GSM operator, and because GSM networks provide a good starting point for modelling telecommunications networks, having been in full-scale use for many years. Structurally, a GSM network is complex, for even its basic units, i.e., base transceiver stations (BTSs) and their cells, have two sources for topology information, geographical and logical. Extensive communication and control are necessary to manage the network and to deliver mobile cell phone traffic. All this makes detailed modelling of the network's dynamics a challenge, which is why simplifications and approximations are necessary in any applicable methods of statistical modelling.

The properties and functioning of single network nodes in MTNs have been analysed, e.g., in [78], [79], [145], whereas network planning and optimisation have been studied, e.g., by [80], [81]. The overall quality of service provided by an MTN has been studied in [156], [157]. Telecommunications networks have also been extensively modelled through networks of queues [60] with each node being referred to as a queuing process. Queue network models yield probability distributions and probability measures such as equilibrium distributions of the number of clients for each network node or for the whole network, hand-off rates, and blocking probabilities (see, e.g., [19], [83]).

Chapters 2–3 introduce the reader to networked systems and their applications. The present chapter is a brief introduction to mobile telecommunications networks with focus on GSM networks. Section 2.1 discusses the structure and functioning of GSM networks, and Section 2.2 outlines the topology information of GSM networks. Finally, Section 2.3 examines the effects of loading and other disturbances on GSM networks. Queuing networks or other previous methods for modelling telecommunications networks are not covered.

2.1 Structure and Functioning

The detailed structure of a GSM network (Figure 2.1) is complex consisting of three subsystems: a Network and Switching Subsystem (NSS), a Base Station Subsystem (BSS), and an Operations Subsystem (OSS). The BSS consists again of Base Transceiver Stations (BTSs) and Base Station Controllers (BSCs) and generally connects mobile stations, such as cell phones, via the NSS, which manages connections within the GSM network and bridges connections to an outside

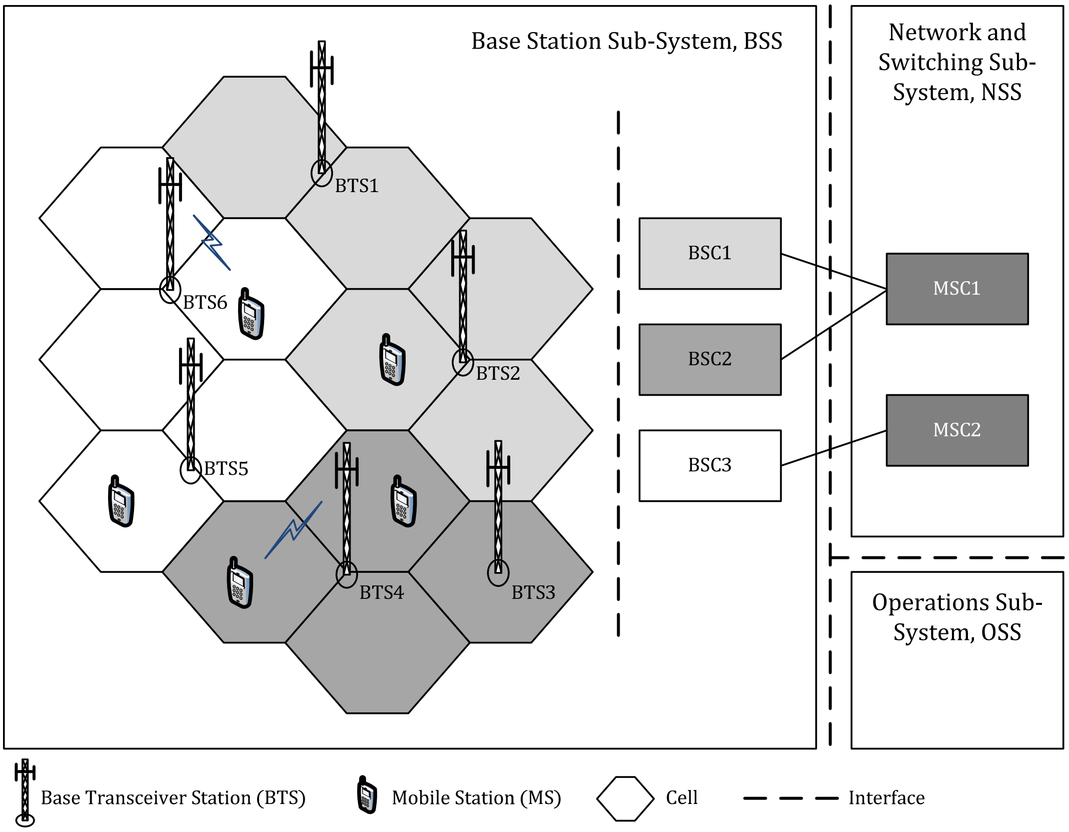
Base Station Sub-System, BSS
Network and Switching Sub-System, NSS
Operations Sub-System, OSS
BTS1
BTS2
BTS3
BTS4
BTS5
BTS6
BSC1
BSC2
BSC3
MSC1
MSC2
Base Transceiver Station (BTS)
Mobile Station (MS)
Cell
Interface

Figure 2.1. Structure of a GSM network. The figure is slightly modified from [156].

network, e.g., a public telephone network. The OSS operates and maintains the network and manages the information of mobile stations. [156]

Mobile stations connect to the GSM network via BTSs, which constitute the basic units of the GSM network. Each BTS has one or several transceivers (TRX) communicating with the mobile stations via BTS antennas in specified areas called (BTS) cells. Cells form an area covered by a single BTS handling communication with the mobile stations. BTSs must cover through the cells the entire geographical area where mobile stations may move to enable connections between the GSM network and the mobile stations [156]. BTS cells are the units considered in this thesis as network nodes. However, because many BTSs contain only one cell, often the two terms have quite the same meaning here.

One or several BTSs operate under a single BSC, and several BSCs are further connected to a Mobile Switching Centre (MSC). The BSC controls its BTSs and transfers incoming and outgoing mobile station traffic between the BTSs and the basic units of the NSS, the MSCs. The BSC also operates alarms, security, and reconfiguration. In general, for a GSM network to work properly, extensive communication and control are required. For details of GSM network structure and functioning, see, e.g., [150] and [156].

2.2 Topologies

Because BTSs manage connections to mobile stations, the BTS cells must cover all the geographical locations where mobile stations can be positioned to enable the latter to connect to a GSM network. Therefore, BTSs (and their cells) are scattered in the geographical area covered by the GSM network, and their physical locations thus form a physical, or geographical node location map of the GSM network. Because the mobile stations connecting to the GSM network cause traffic, or loading, on the GSM network, and because mobile station connections depend on their physical locations, the loadings of physically close cells obviously correlate.

Mobile stations can change their physical locations while connected to a GSM network, whereas BTS locations are fixed and cover a certain physical area where they maintain communications with the mobile stations. Therefore, when a mobile station exits an area covered by one cell and enters one covered better by another cell, the connection between the GSM network and the mobile station is retained by transferring it from the former BTS to the latter. In telecommunications networks, this operation is referred to as handover.

Handover is one example of exchange of information between BTS cells. However, though physically close cells usually co-operate via handovers, logical neighbour connections, in fact, determine the cells to which a given cell is connected. All logical connections in a network can be depicted as an undirected graph with the links between the nodes, cells, representing logical connections or neighbourhood relations. We have thus two pieces of topology information: the first defined by physical locations of cell nodes, in which internode distances describe continuously dissimilarities between the cell nodes, and the second deriving from logical cell relations, defined in logical topology as binary node relations.

2.3 Loading and Disturbances

In a GSM network, speech and data transferred between mobile stations and BTSs cause a loading on the BTS cells. Because the number of mobile stations and the amount of traffic caused by speech and data of the mobile stations vary from one location to another, the loading on cell nodes varies both over time and space. If a BTS cell becomes loaded close to its maximum capacity, attempts are made to hand the on-going connections between the BTS cell and the mobile stations over to some logical neighbouring BTS cell. If this is not possible, e.g., because the mobile station cannot reach other cells, or because the other cells are also operating at full capacity, connections to mobile stations must be dropped [80].

In addition to the loading caused by the traffic of mobile stations, various disturbances may affect steady network operation. Such disturbances include failures in devices maintaining network traffic, electric blackouts at BTSs or in parts of the network, and possibly even hostile attacks at the network and its BTSs. Disturbances causing failure at cells and links between cells may drastically affect the quality of service of the network to mobile stations, as calls may be dropped or even widespread traffic blackouts may ensue (see, e.g., [168], [169], and [170]). This thesis deals only with disturbances that affect network operation through network loading and changes in traffic loading.

3. Networked Systems and Properties

MTNs are an example of technical networked systems with other examples being, e.g., the Internet, power grids, supply chains, and water distribution systems. More generally, examples of networked systems include, e.g., neural networks in human brains, cellular networks in living organisms, social networks of human beings, and the ecological networks of food webs.

Networked system is thus a very wide concept, at its widest signifying any system consisting of nodes that interact. This is a loose definition because in general systems consist of subsystems that somehow exchange information or goods or interact by other means. In this thesis, networked system refers to any system with homogeneous nodes, each specified by a state and each interacting through network topology with some other nodes, thus causing correlations seen in the joint statistics of states of interacting nodes. In addition, each node is specified in having an external effect on the state of the node itself.

Today, by complex networks it is usually referred to networks with certain non-trivial topological properties, related, e.g., to the distribution of the number of neighbours or to the clustering of nodes [9]. On the other hand, complex systems also refers to systems with such behavioural qualitative properties that cannot be explained only by inspecting the properties of its subsystems [14], because new, unexpected properties emerge as a consequence of interaction between the subsystems. In practice, complex systems are networked systems with their behavioural properties affected by underlying topological network features.

Studying some specific networked systems, such as MTNs, one should review in general the properties networked systems usually have, because various systems indeed share similar structural and behavioural properties [9], [42], [44], [106]. Consequently, methods for analysing a particular network application may prove equally well suited for other, initially rather distant, applications. Therefore, the present analysis of MTNs and their properties make use of a general approach to networked systems.

This chapter introduces networked systems and considers both general topological and behavioural properties. Section 3.1 introduces the concepts and data used in this thesis, and Section 3.2 considers the various types of network topologies and their topological properties. Finally, Section 3.3 discusses the effect of node removals and node failures on the topological and behavioural properties of networks.

3.1 Data and Concepts

Each network node is associated with a state and a loading. The state and loading values are both assumed to be observed without measurement uncertainty. In general, node state data can be either discrete- or continuous-valued, but here only binary node states are considered in detail. Respectively, loading data may assume either discrete or continuous values. Here the state of node m as a random variable is denoted as S_m, whereas its value in a network observation l is denoted as $s_m^{(l)}$. Similarly, the loading of node m in a network observation l is denoted as $h_m^{(l)}$. The network observation of a set of node state–load pairs is denoted as $\{s_m^{(l)}, h_m^{(l)}\}_{m=1}^{M}$. The number of network nodes is denoted by M and the number of observations by L.

Each node may also have a specified physical location and always has a set of neighbours—MTNs assume both of these pieces of topology information. Here nodes are also assumed homogeneous in the sense that they are structurally identical. Both the physical locations of nodes and their neighbourhood relations are assumed constant through network observations. The physical location of node m, if given, is described by the node location map of the network and is denoted by a coordinate vector $\mathbf{x}_m$. The neighbours of node m are denoted by a set of nodes $n \in N(m)$, which for all nodes can be visually presented as a graph.

Specifying a network node with a state and loading in general description means that for some specific networks, such as GSM networks, the information available about the nodes for each node must be compressed into the two variables, the state and the loading. With GSM networks, the two variables have the following meaning: the node state variable describes the performance of a BTS cell, e.g., how well the cell node performs requests from mobile stations; respectively, the load variable describes the external load, the amount of traffic caused by speech and data flows to and from mobile stations and affecting the BTS cells. However, since this is not the only way to define node states and loads in a GSM network, one goal in seeking an appropriate network model is also to find an appropriate node description.

Structure, or topology, is essential in defining many properties of a networked system [9], [44], [143]. Topology refers here to the list of nodes in networks and the list of interconnections between the nodes. In addition, the term graph, or graph structure, is used here in parallel to topology. For a comprehensive specification of a networked system, the weights or strengths of interconnections are often specified in addition to a topological or graph description.

3.2 Topologies

Even diverse networks may share a similar basic structure of network topology. Only few such structures have been thoroughly studied in graph theory [38]. Regular networks are usually a simplification, or an idealisation, of a true system topology. Yet regular topology structures exist in nature as well; e.g., in ice crystals, where the atoms are organised in a three-dimensional regular square grid. In a two-dimensional lattice, regular networks may assume interconnections in squares, triangles, or hexagonals (a regular square lattice structure is shown in Figure 3.1). In regular networks, if the network is infinite, each node has the same number of neighbours. In finite regular networks, the edge-nodes have fewer neighbours than the other nodes. However, in some cases, finite networks are deemed to have periodic boundary conditions with all nodes having the same number of neighbours. Periodic boundary conditions mean that the edge nodes are neighbours to nodes at the edges on the opposite side of the network, the network being a torus.

Random graphs, or random networks, [47], [48], [133] are models for network structures where node links are drawn samples according to some specified random process [9]. Widely studied

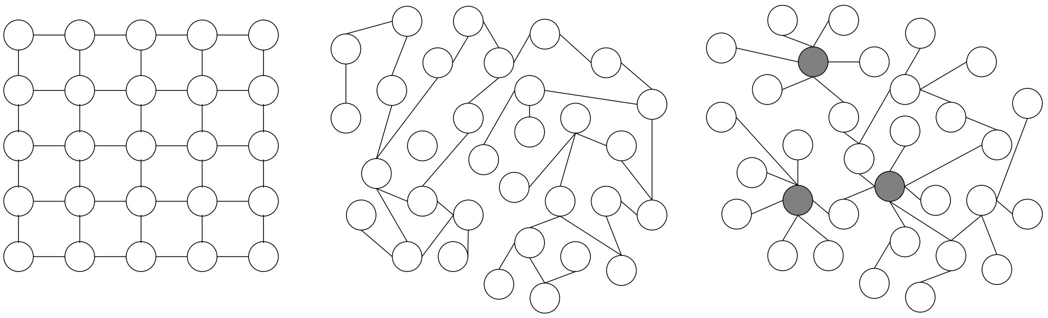

Figure 3.1. Examples of regular square lattice ($5x5$) (left), random network (middle), and scale-free network (right). The last two topologies have both 32 nodes and 32 links and are redrawn according to [26].

[47], [48], the Poisson process leads to the Poisson distribution of the number of neighbours, but random graphs with arbitrary distributions of the number of neighbours have been studied as well [101], [102], [107] (a random network topology is shown in Figure 3.1). Because the Poisson distribution is right-skewed with exponential tails, only few nodes have many more neighbours than the typical neighbourhood. Though extensively used to study the properties of real networks, random graphs with the Poisson distributed number of neighbours are seldom realistic since the real distribution of the number of neighbours is usually non-Poisson, such as the power law or exponential distribution [9], [25].

Scale-free networks [9] are more complex in structure than the above with properties similar to those of many real-world networks. Scale-free networks have their number of neighbours distributed according to the power law. Consequently, scale-free networks are scale invariant [9]—their topology appears similar at all length scales. Because the power law distribution has a heavier tail than the exponential distribution, many nodes have relatively many neighbours (a scale-free network is shown in Figure 3.1). An artificial generation of scale-free networks (see, e.g., [9]) mimics the evolution of growing networks; i.e., a new node is more likely to connect to a node with a high than a low number of neighbours. Growing networks often self-organise into scale-free structures [44].

Two further properties are generally used to classify network structures. One is the mean-shortest path length (MSPL), which is the average node graph distance over all node pairs. Graph distance means here the smallest number of steps between two nodes along neighbourhood relations. Such networks have short graph distances in comparison to the size of the network. The other is about the clustering of the network structure, characterised by the clustering coefficient [160], i.e., how close the nodes are on average to forming cliques with their neighbours. Clique means a group of nodes in which each node is directly connected to every other node.

A network is called a small-world network if it has a small MSPL value and a large clustering coefficient. Hence, small-world networks contain shortcut links that connect otherwise distant parts of the network, and the nodes form clusters of inter-connected node groups. Regular networks lack shortcuts and have thus large MSPL values, whereas both random graphs and scale-free networks assume shortcuts and thus small MSPL values [9]. In addition, scale-free networks and regular networks are typically highly clustered [9], whereas random networks are not because of their randomly drawn links [41], [9]. Among regular networks, only nodes in triangular networks form cliques with their neighbours and assume thus a non-zero clustering coefficient. Finally, true networks often have some special properties that go beyond generated idealised topologies, and finite networks may be hard to classify as pure members of any topology types.

3.3 Properties

Networked systems may experience disturbances, e.g., due to device failures or electrical blackouts, which affect their system topology and its connectivity by either disrupting links or removing nodes. Scale-free networks are robust against random node removal, because a randomly chosen node comes most likely from among the typical nodes with relatively few links. Therefore, because the central nodes are likely to be unaffected, the network is still likely to be connected via a path from every node to every other node, even if relatively many nodes are randomly removed. Consequently, random node removals hardly change the topology or connectivity of scale-free networks, unless some central nodes with many neighbours are removed. However, removal of the central nodes may drastically affect network connectivity with the network fragmenting into several unconnected subnetworks [10].

Because random graphs and regular networks lack central nodes, all the nodes assume nearly the same number of neighbours. Consequently, a randomly removed node in a random network contributes relatively more to the network's connectivity than it does in a scale-free network with the same average graph distance. Random networks are thus more vulnerable to random node removals. When the threshold number of removed nodes is exceeded, the network decomposes into unconnected subnetworks [9]. On the other hand, random networks are more robust against removal of the most connected nodes than scale-free networks, because they have only few nodes with a large number of neighbours in the first place, which are thus rather insignificant to network connectivity [10]. Yet, in general, network robustness against removals is also affected by general network connectivity. Furthermore, let us recall the above reminder that real network topology may not follow any idealised topology type considered here (for more on the effects of node removals on networks, see, e.g., [9], [10], [34], [44]).

This thesis does not consider link disruptions or node removals and their impact on topological properties. Instead, since the focus here is on MTNs, node failures are considered when a node operates poorly, or abnormally, and affects the states of its neighbouring nodes through its poor performance rather than by removing the node and its links from the network. In MTNs, such node failures may be caused by heavy node loadings, phenomenon whose impact on node states and thus the whole network operation are under scrutiny here. In particular, the average network node state is studied under varying node loadings. Though topological properties are not considered here, the connectivity of normally operating nodes can be analysed under a certain loading configuration to find out how the topology changes if abnormally operating nodes are ignored.

Therefore, though the connectivity properties of two types of idealised network topologies under node removals have been briefly discussed above, the emphasis here remains on behavioural rather than topological properties. Topological properties also affect a networked system's behavioural properties, but only partly; the strength of node interactions is significant, too. In fact, interaction strengths are highly essential, because they determine the statistical coherence of the network analysis in this thesis, i.e., the stronger the interactions, the more coherent the network. In addition, though topology affects coherence, in this thesis network topology is mostly fixed and coherence therefore determined by node interactions only.

Coherence affects a network's qualitative behaviour, determining how the network behaves under external node loadings. Depending on its coherence, a networked system under varying loading may thus exhibit critical phenomena with continuous phase transitions at a critical coherence level or discontinuous phase transitions at high network coherence. A coherent networked system may also exhibit hysteresis with the state of the network depending on its past loading values. All these properties are studied in depth in Chapter 4.

4. Markov Random Fields as Models of Networked Systems

Markov random field (MRF) models are a set of collaborative statistical models representing dependencies of variables through a joint probability. MRF models have their origin in statistical physics, where the Ising model [66] was first used to analyse ferromagnetism and its properties [70]. Since then the Ising model and other MRF models have been extensively used to model collective network behaviour and applied to spatial modelling, e.g., in image analysis [161], geostatistics [33], [59], and genetics [92].

In the MRF model, a graph presentation of the interconnections of variables considered specifies a set of conditional independence properties and defines the structure of the MRF model. To specify an MRF model for a networked system, both the graph structure of the variables and the parameters of the chosen MRF model type must be set. In general, both structure and parameters affect the qualitative properties of an MRF model, but after the model's graph structure has been fixed, only parameters determine qualitative model properties.

Chapters 4−7 provide the methodological background to this work. This chapter begins with an introduction to MRF models, which in this thesis are applied as models for networked systems. Section 4.1 introduces first the general structure and definition of MRF models, followed by a few specific MRF models in Section 4.2. Section 4.3 discusses the physics background of MRF models, and Section 4.4 reviews in general the qualitative properties of the Ising model applied in this thesis.

4.1 General Structure and Definition

An MRF model is a joint probability distribution (JPD) of a set of random variables. In it, the JPD factorises into a product of subset JPDs, the factorisation being determined by conditional statistical independence (hereafter conditional independence for short) properties of node variables. The maximum number of nodes that are all pairwise conditionally dependent, given the states of the rest of the nodes, form one factorisation term. These conditional dependence relations can be visually presented in a graph, in which an undirected link is drawn between all pairwise conditionally dependent nodes (see Figure 4.1). Connected nodes are called neighbours. The set of neighbours of a node i is called the node's Markov blanket and denoted here by $N(i)$. Having undirected links, all neighbourhood relations are symmetric: $j \in N(i) \Leftrightarrow i \in N(j)$.

The notations in Section 3.1 can be used to present conditional independence properties formally. Denoting by $\mathbf{s}_{-ij}$ the states of all network nodes, except for nodes i and j, the pairwise conditional independence between node states s_i and s_j, given $\mathbf{s}_{-ij}$, can be written as

$$p(s_i, s_j | \mathbf{s}_{-ij}) = p(s_i | \mathbf{s}_{-ij}) p(s_j | \mathbf{s}_{-ij}). \tag{4.1}$$

Consequently, the conditional joint probability of node states s_i and s_j factorises into a product of marginal conditional probabilities, when the nodes are conditionally independent. This property can also be expressed as $p(s_i | s_j, \mathbf{s}_{-ij}) = p(s_i | \mathbf{s}_{-i}) = p(s_i | \mathbf{s}_{-ij})$. Furthermore, since the

Markov blanket of a node i defines the complete set of nodes $N(i)$, on which the state of node i is pairwise conditionally dependent, the previous condition can also be formulated as

$$p(s_i|\mathbf{s}_{-i}) = p(s_i|\{s_n\}_{n\in N(i)}). \tag{4.2}$$

This conditional probability is sometimes called the full conditional, a very useful concept in estimating MRF model parameters in Chapter 6 and in simulating the model in Chapter 7.

Two nodes connected with a link and thus pairwise conditionally dependent are said to form a clique on the graph. In general, a clique is a subset of nodes such that all nodes pairwise conditionally depend on each other. Thus nodes belonging to a clique are all directly connected to one another in the graph. A maximal clique is a clique of nodes that is not a subset of any other clique. For example, in Figure 4.1 each node forms a (trivial) clique of size 1, and each node pair connected with a link (e.g., nodes 1 and 2) forms a clique of size 2. Nodes 2, 3, and 6 form a maximal clique of size 3; but also node pairs (1, 2), (1, 4), (4, 5), (5, 6), (6, 7), (7, 8), and (3, 8) all form a maximal clique of size 2, because they are not subsets of any other cliques.

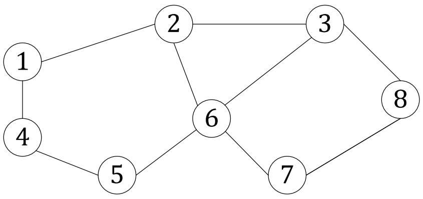

Figure 4.1. A graph with eight nodes.

Because the MRF JPD is defined in general based on pairwise conditional dependencies, the JPD can be formulated by using cliques. Let us first define the potential function (PF) of a clique as any positive definite function of the node states within the clique. In its general form, the MRF JPD is the product of the PFs of all cliques. In fact, it can be defined only through the PFs of maximal cliques, because any other cliques are subsets of the maximal cliques. However, this thesis considers only the PFs of maximal cliques that are products of the PFs of node pair cliques and the PFs of single node cliques within the maximal clique.

The following notations are used to define the MRF JPD. First, the set of all node pairs on a graph are denoted by V, and the PF of a node pair clique of nodes i and j is denoted by $\psi_V(s_i, s_j)$, where $(i,j) \in V$. A PF of a single node clique is denoted by $\psi(s_m)$, where the subscript $m = 1, \dots M$ indexes the set of all nodes. Now the probability associated with a joint state $\mathbf{s}$ (a vector of size M) of a random variable $\mathbf{S}$ of M nodes, the MRF JPD, can be written as

$$p(\mathbf{s}) = Z^{-1} \prod\nolimits_{(i,j)\in V} \psi_V(s_i, s_j) \prod\nolimits_{m=1}^{M} \psi(s_m). \tag{4.3}$$

Here Z is a partition function that normalises the probabilities and is defined as a summation over all combinations of node states $\mathbf{s}$: $Z = \sum_{\mathbf{s}} \prod_{(i,j)\in V} \psi_V\,(s_i, s_j) \prod_{m=1}^{M} \psi(s_m)$. In the definition of the partition function, discrete-valued variables are assumed; with continuous-valued variables, summations are replaced by integrals. Different choices of potential functions lead to different model types, which are considered in the following subsection. Graph structure defines the global structure of an MRF model, whereas specification of potential functions defines local properties.

4.2 Model Types

Because positive definite, exponential functions are commonly used as potential functions. The PFs of node pair cliques determine the node-to-node interactions in the network, whereas the PFs of single node cliques determine the single node effects that can be thought of being due to external forces or an external field, and affecting nodes locally. In this thesis only exponential PFs of node pair cliques and single node cliques are considered. This section introduces the following three MRF model types based on the structure of their node state: the binary-state Ising model, the discrete-state Potts model, and the continuous-state Gaussian model.

4.2.1 Ising Model

The Ising model [66], [89] originates in statistical physics, where it was first used to model ferromagnetism, i.e., the alignment of magnetic particles under the effect of an external magnetic field [166]. The properties of the Ising model have been studied in detail in statistical physics [166], but the model has also been applied to many other fields, such as image analysis [161], the distribution of galaxies in the universe [139], financial markets [134], [135], [154], spread of perturbations, e.g., diseases [140], and the elasticity theory of DNA [8].

In the Ising model, each node has only two possible states, here -1 and $+1$, and is hence classified as a binary node state MRF. Though at the node level it is then a very simple model, it has complex coherent properties and is phenomenologically rich for study of the collective behaviour of complex networks. In the previous notations, the JPD of the Ising model is here defined as

$$\begin{aligned} p(\mathbf{s}) &= Z^{-1}\prod_{(i,j)\in V}\exp(J_{ij}s_is_j)\prod_{m=1}^{M}\exp[Hs_m(h_m-h_0)] \\ &= Z^{-1}\exp\left[\sum_{(i,j)\in V}J_{ij}s_is_j + H\sum_{m=1}^{M}s_m(h_m-h_0)\right]. \end{aligned} \tag{4.4}$$

The first model term inside the exponent is the interaction term, whereas the second term is the external field (load) term. Here J_{ij}, H, and h_0 are the parameters of the model. J_{ij} defines the strength of interaction between neighbouring nodes i and j, H the magnitude of the external loads, and h_0 the threshold value of the external load h_m, i.e., node m prefers state $+1$ if $h_m > h_0$ and state -1 if $h_m < h_0$. This thesis assumes throughout a uniform J_{ij}, i.e., for all node pairs i and j $J_{ij} = J$; hence it describes the magnitude of the interaction term. Each parameter may assume any positive or negative scalar value.

Here the external loading term is over-parameterised in the sense that we may also write $Hs_m(h_m - h_0) = s_m h_m$, where the effects of H and h_0 are the same as re-scaling h_m and changing the zero of h_m, respectively. Nevertheless, over-parameterisation is applied to clearly interpret the parameters.

Because only node pair cliques and single node cliques are used here, the Ising model can also be written in terms of node neighbourhoods:

$$
\begin{aligned}
p(\mathbf{s}) &= Z^{-1}\exp\left[J\sum\nolimits_{m=1}^{M}\sum\nolimits_{n\in N(m)} s_m s_n + H\sum\nolimits_{m=1}^{M} s_m(h_m - h_0)\right] \\
&= Z^{-1}\exp\left[J\sum\nolimits_{m=1}^{M} s_m\left\{\sum\nolimits_{n\in N(m)} s_n + H(h_m - h_0)\right\}\right] \\
&= Z^{-1}\exp\left[\sum\nolimits_{m=1}^{M} s_m h_m^{\mathrm{eff}}\right].
\end{aligned}
\tag{4.5}
$$

Here the last form defines the effective load h_m^{eff} of node m, which is the total loading affecting the node state due to node interaction and external loading.

As node interactions (*J*) increase, the Ising model exhibits a behaviour typical of complex networked systems, such as spontaneous organisation at zero external field, resulting in coherence in the network node states and discontinuous phase transitions with hysteresis under external loading. These phenomena are discussed with the Ising model in detail in Section 4.4.

4.2.2 Potts Model

The Potts model [113] is a discrete-state MRF model, in which a node may have any positive integer, q, number of states. Like in the Ising model, the JPD of the Potts model factorises into the product of potential functions of single node cliques and node-pair cliques, and can be written in general form as

$$
p(\mathbf{s}) = Z^{-1}\prod\nolimits_{(i,j)\in V}\exp[J_{ij}\delta(s_i,s_j)]\prod\nolimits_{m=1}^{M}\exp[H(s_m)(h_m - h_0)]. \tag{4.6}
$$

Here δ is the Kronecker delta-function, which is zero everywhere except when $s_i = s_j$, where it assumes the value one. Therefore, two interacting nodes i and j contribute to the JPD only when they assume an equal state. Parameters J_{ij} and h_0 mean the same as in the Ising model. The parameterisation function H still defines the magnitude of the external loading but is now a function of the node state s_m, because the q possible states; respectively, direct multiplication by the node state s_m is omitted here.

The Potts model can be considered a q-state extension of the Ising model, because with $q = 2$ the Potts model reverts to the Ising model. The properties of the Potts model have been studied extensively with varying q values and dimensions (see, e.g., [163]). For example, at the critical point of spontaneous organisation, in two-dimensions phase transitions in the Potts model are discontinuous with $q > 4$ and continuous for $q \leq 4$ [163], [166].

4.2.3 Gaussian Model

The Gaussian model [122], or Gaussian MRF model (GMRF), is a continuous-state model, with its node states assuming continuous values. In its simplest form without an external load term, the GMRF model is the ordinary Gaussian JPD:

$$p(\mathbf{s}) = (2\pi)^{-M/2}|\mathbf{Q}|^{1/2} \exp\left(-\frac{1}{2}\mathbf{s}^{\mathrm{T}}\mathbf{Q}\mathbf{s}\right), \tag{4.7}$$

where $\mathbf{Q}$ is an MxM precision matrix (inverse of the covariance matrix) of $\mathbf{s}$ with the element Q_{ij} defining the strength of interaction between nodes i and j. Like in the Ising and Potts models, the external load term can be included in the Gaussian model by presenting it in the form of canonical parameterisation [122], after which the model JPD can be written as

$$\begin{aligned} p(\mathbf{s}) &= Z(\mathbf{Q},\mathbf{b})^{-1} \exp\left(-\frac{1}{2}\mathbf{s}^{\mathrm{T}}\mathbf{Q}\mathbf{s} + \mathbf{b}^{\mathrm{T}}\mathbf{s}\right) \\ &= (2\pi)^{-M/2}|\mathbf{Q}|^{1/2} \exp\left(-\frac{1}{2}(\mathbf{s} - \mathbf{Q}^{-1}\mathbf{b})^{\mathrm{T}}\mathbf{Q}(\mathbf{s} - \mathbf{Q}^{-1}\mathbf{b})\right), \end{aligned} \tag{4.8}$$

where in the first form the second term defines the external loading term, and $Mx1$ vector $\mathbf{b}$ contains the node loadings and loading parameters; e.g., $b_m = -H_m(h_m - h_0)$. $Z(\mathbf{Q}, \mathbf{b})$ is the partition function, which normalises the probabilities, and whose analytical form is known and easy to calculate. As the second form in Eq. (4.8) shows, the canonical form is just a reparameterisation of the Gaussian distribution; hence all related computational methods are available. From the viewpoint of MRF models, the interpretation of the precision matrix $\mathbf{Q}$ in the Gaussian MRF has an intriguing property: the element Q_{ij} is nonzero only if nodes i and j are pairwise conditionally dependent. Consequently, the nonzero elements of the precision matrix $\mathbf{Q}$ instantly determine the graph structure of the GMRF. Because of the sparseness of the precision matrix $\mathbf{Q}$, many efficient computational methods have been developed for the GMRFs (see, e.g., [122]).

From the standpoint of this thesis, GMRF models have the disadvantage that the phenomena they exhibit are rather simple in the sense that no phase transitions or hysteresis takes place at all. Instead the average network node state changes smoothly as a function of external loading, as shown by the second form of Eq. (4.8), where the expectation value of $\mathbf{s}$, $\mathbf{Q}^{-1}\mathbf{b}$ is a linear function of the loading parameter vector $\mathbf{b}$. It also follows that fluctuations around the mean values correlate according to the covariance matrix $\mathbf{Q}^{-1}$. In addition, on a lattice with finite neighbourhood relations according to $\mathbf{Q}$, correlations described by $\mathbf{Q}^{-1}$ decay exponentially in the distance between two nodes; hence no network-wide state coherence can occur.

4.3 Physics Background of MRF Models

MRF models originate in statistical mechanics, where the Ising model was first used to study phase transitions [66], [89]. Because the Ising model's many properties are explained in physical quantities such as temperature and magnetisation, its structure and properties can be better understood in terms of these physical concepts. This section discusses briefly how the Ising model can be derived from one fundamental concept of statistical mechanics, the Boltzmann distribution, and how temperature is related to the parameters of the model.

Statistical mechanics deals with macroscopic properties of systems consisting of large numbers of microscopic particles, each assuming a specific state (see, e.g., [86], [126]). Statistical mechanics assumes that for an isolated system, all accessible microstates are equally probable [126]. An isolated system consists of a studied system connected to a large reservoir with which the system exchanges energy according to the principle of equal probability of microstates. Such arrangement is called thermal equilibrium.

By using the laws of thermodynamics and the concept of equally probable microstates, a probability can be associated with the state of the studied system. However, rather than the probability depending directly on the state configuration of the microscopic particles, it now depends on the total energy exhibited jointly by those particles. Consequently, the total energy consists not only of energies related to individual particles, but also of energies arising from particle interactions. The probability for the system to assume a microstate configuration $\mathbf{s}$ with the related total energy $E(\mathbf{s})$ is determined by the so-called Boltzmann distribution (see, e.g., [126]):

$$P(\mathbf{s}) = Z^{-1}\exp[-\beta E(\mathbf{s})]. \tag{4.9}$$

Here $\beta = 1/kT$, where k is the so-called Boltzmann factor (constant) and T the temperature of the reservoir with which the system is assumed to be in equilibrium. The exponential factor is called the Boltzmann factor, and Z is the partition function (normalisation term), defined as the sum over all the possible Boltzmann factors or microscopic states $\mathbf{s}$:

$$Z = \sum_{\mathbf{s}} \exp(-\beta E(\mathbf{s})). \tag{4.10}$$

The partition function plays a central role in statistical mechanics, because many macroscopic quantities describing the system, such as free energy and entropy, can be calculated when the partition function is known. However, in practice the partition function is very difficult, if not impossible, to calculate for large systems, because the number of possible microstates grows exponentially with the size of the system; for example, consider the Ising model with M nodes with a total of 2^M microstates [166].

As an example of the Boltzmann distribution, let us consider the formulation of the Ising model in Subsection 4.2.1. The total energy of a networked system is now composed of energies due to node interactions and of external energies affecting the individual nodes. Using the same notations as in Subsection 4.2.1 and further denoting by $-J'$ the energy contributed by a single interacting node pair and by $-h_m$ the external energy contribution of a node m, we can write the total energy of a state configuration $\mathbf{s}$ as

$$E(\mathbf{s}) = -J' \sum_{(i,j)\in V} s_i s_j - \sum_{m=1}^{M} h_m s_m, \tag{4.11}$$

where the threshold parameter h_0 used in Subsection 4.2.1 is omitted for simplicity. When we now insert Eq. (4.11) in the Boltzmann distribution of Eq. (4.9), the JPD of $\mathbf{s}$ becomes

$$\begin{aligned} P(\mathbf{s}) &= Z^{-1}\exp(-\beta E(\mathbf{s})) \\ &= Z^{-1}\exp\left(\beta J' \sum_{(i,j)\in V} s_i s_j + \beta \sum_{m=1}^{M} h_m s_m\right). \end{aligned} \tag{4.12}$$

Comparing this definition to that of Eq. (4.5) and omitting h_0 from Eq. (4.5), we can see that the parameters in Eq. (4.5) are redefined as $J = \beta J'$ and $H = \beta$, and that both parameters J and H in Eq. (4.5) include the temperature parameter T via parameter β.

In most theoretical phase transition analyses (e.g., [166]) of the Ising model, external loadings are assumed uniform; i.e., for each node m, it is assumed that $h_m = h$, where h is the uniform external loading. However, this approach is often unrealistic for describing true phenomena in real networked systems, because for a regular grid and uniform loading, all nodes appear identical in the model. Yet, for example, in MTNs node loadings come from the traffic caused by mobile phones, and may obviously significantly vary locally and thus drastically affect the network behaviour.

The random-field Ising model is a version of the Ising model, developed accurately to describe the phenomena of some real networks by incorporating also the node-specific loading in the model as a random variable. When Eq. (4.12) is changed according to the random-field Ising model, both uniform and node-specific random-field components are included in the node loading; the loading of node m becomes then $h_m = h + h_m$. In theoretical analysis, random loading components are randomly drawn from some distribution, e.g., identically and independently distributed according to the Gaussian distribution: $f(h_m) = 1/\sqrt{2\pi}\exp[-h_m^2/2\sigma^2]$ [42]. The random-field Ising model describes true fluctuations and variations, or disorder, occurring in real systems. This model is used in Section 4.4 to demonstrate hysteresis phenomena, which the regular Ising model with uniform loadings and regular neighbourhoods is unable to describe.

4.4 Properties of the Ising Model

The Ising model exhibits various phenomena typical of complex networked systems. Though some typical networked system phenomena were briefly discussed in Chapter 3, this section is a technical review of these properties and concentrates on their appearance in the Ising model. In the literature, the qualitative properties of the Ising model are usually studied in the context of ferromagnetism, where the average magnetisation of spin nodes plays the role of the order parameter manifesting the model's qualitative properties, i.e., which type phase transition the model exhibits under which conditions. Phase transition is understood as a discontinuity in some overall system properties or in some of their derivatives with respect to model parameters [166].

Because the focus here is on technical networked systems, Ising model properties are not examined as they appear in magnetism; instead concepts relevant to the subject of this thesis are adopted by using the following substitutes: nodes instead of spins, node states instead of spin magnetisations, and external node loadings instead of external magnetic fields. Because in physics temperature is a quantity greatly affecting magnetisation properties, Ising model properties are studied as a function of temperature. However, here the temperature parameter is replaced with the parameter J of Eq. (4.5), which is inversely proportional to the temperature used in the context of magnetism (see Section 4.3). The term phase transition is still used here to describe changes taking place in a state of the model, i.e., the average of the node states.

In the literature, the properties of the Ising model are usually studied in connection of either infinite regular network topology, such as a square lattice with nearest-neighbour relations or a fully connected infinite graph with each node a neighbour of every other node. Recently, the Ising

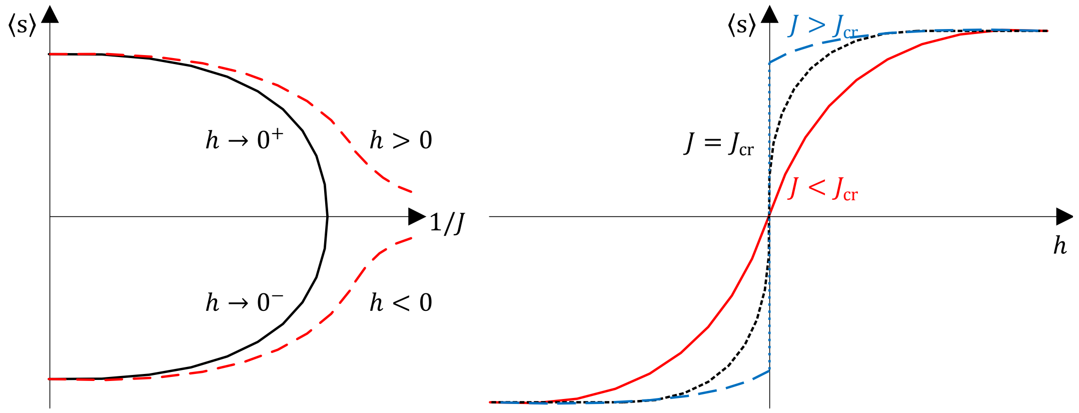

Figure 4.2. Qualitative properties of the Ising model. The left-hand plot shows spontaneous symmetry-breaking organisation in the Ising model under zero external field. The right-hand plot shows the average network node state $\langle s \rangle$ as a function of h with three levels of interaction parameter J. The average state $\langle s \rangle$ changes continuously, when $J < J_{\mathrm{cr}}$ (solid red). Phase transition is continuous with the diverging first derivative at $h = 0$ when $J = J_{\mathrm{cr}}$ (dotted black). Phase transition is discontinuous at $h = 0$ when $J > J_{\mathrm{cr}}$ (dashed blue).

model has also been analysed on more complex network structures [42], [43]. However, results of the idealised network structures are reported here as found in the literature to demonstrate the types of phenomena the Ising model may exhibit. Indeed, real networks are finite in size and usually widely irregular in topology [9]. Because of this finiteness, some consequences ensue from the boundaries of a finite network, where the nodes at the boundaries have generally fewer neighbours than the nodes at the centre of the network. However, because their network topology is anyway irregular with a varying number of node neighbours, boundary nodes are not expected to cause further peculiarities in model behaviour.

4.4.1 Qualitative Properties of the Ising Model

The Ising model's qualitative phenomena and types of phase transitions are schematically illustrated in Figure 4.2 for a regular network topology of M nodes and with nearest-neighbour interactions. On the left, the average state of the network nodes, denoted $\langle s \rangle$, is given as a function of $1/J$ in a zero external field ($h = 0$), and on the right $\langle s \rangle$ is given as a function of external uniform field, or loading, h with three J values. The uniform external loading is changed adiabatically, which means that the dynamics of the external loading are much slower than those in the model. Consequently, after the uniform external loading is changed infinitesimally, the system has enough time to relax into its metastable stationary state before another loading change is executed.

In general, the Ising model incorporates two types of phase transition; first-order, or discontinuous transitions and second-order, or continuous transitions with discontinuities in the first derivative of system properties. A thorough analysis of the model on a two-dimensional square lattice, as found in [166], is given next.

The Case with $J < J_{cr}$

Let us first study the state behaviour of the nodes in the Ising model using the right-hand plot in Figure 4.2, where the three curves give $\langle s \rangle$ with three J values: the solid red curve corresponds to $J < J_{\mathrm{cr}}$, the dotted black curve to $J = J_{\mathrm{cr}}$, and the dashed blue curve to $J > J_{\mathrm{cr}}$. With $J < J_{\mathrm{cr}}$, node loading dominates node interaction, and statistically the node states depend weakly on each other. Hence $\langle \mathrm{s} \rangle$ is a continuous function of loading between two extreme states, $\langle \mathrm{s} \rangle = -1$ and $\langle \mathrm{s} \rangle = +1$.

When the internode interaction gains strength and point $h = 0$ is approached under an external load change, the node states fluctuate less yet more coherently with significant correlations appearing in neighbouring node states. Interpreted physically, as the temperature drops, the interaction energy contributes increasingly, compared to thermal fluctuations in nodes due to temperature. That is, at low temperatures and at $h = 0$, the Ising model sets higher probabilities for organised network states, whereas at high temperatures, organised and disorganised states are almost equally probable.

In general, correlations in node states are described with a measure called correlation length, which measures the maximum size of node clusters that show correlations in their states. Hence the correlation length increases at the approach of the critical point J_{cr}; the system is then said to show a short-range order [166], because correlated node regions begin to appear.

The Case with $J = J_{cr}$

At $J = J_{\mathrm{cr}}$, when approaching the point $h = 0$, correlations appear at all length scales, and the correlation length becomes infinite. Hence a single node cluster can dominate the entire network, and the average state $\langle \mathrm{s} \rangle$ can be non-zero. The first derivative of $\langle \mathrm{s} \rangle$ with respect to h diverges at $h = 0$, indicating a second-order or continuous phase transition; on the right hand plot in Figure 4.2 at $h = 0$, the slope of $\langle \mathrm{s} \rangle$ approaches infinity.

The above phase transition can also be analysed using the left-hand plot in Figure 4.2, which shows the temperature, or $1/J$, dependence of $\langle \mathrm{s} \rangle$: at $J = J_{\mathrm{cr}}$ the symmetry of the system is said to be broken, and the system starts spontaneously organising itself into one of two possible extreme (coherent) states, the chosen state depending on random fluctuations. If the external loading assumes an infinitely small positive value, the path to state $+1$ is chosen, whereas if it assumes an infinitely small negative value, the path to state -1 is chosen. Finally at zero temperature (J approaches infinity), the system, respectively, ends up either in state $\langle \mathrm{s} \rangle = +1$ or in state $\langle \mathrm{s} \rangle = -1$. The network state is said to "freeze" with fewer and fewer fluctuations, when the internode interaction increases until all nodes finally assume the same state, either -1 or $+1$. Physically, the system now reaches its state of minimum energy.

Many intriguing critical properties appear at continuous phase transition. In particular, many quantities such as the correlation length, the average node state, and the sensitivity to external loading (first derivative of $\langle \mathrm{s} \rangle$ with respect to h) are all power law functions of $J - J_{\mathrm{cr}}$. The correlation length and the sensitivity to external loading diverge at $J = J_{\mathrm{cr}}$. The exponents in power laws are called critical exponents and are known to be universal, whereas J_{cr} depends on the details of the system's dynamics. Universality means that exactly the same exponents appear in many different physical systems, and that they depend only on a few fundamental parameters, such as the dimensionality and symmetry of the order parameter $\langle \mathrm{s} \rangle$ in models with short-term interactions, e.g., the Ising model on a regular lattice with nearest-neighbour interactions. [166]

The Case with $J > J_{cr}$

At $J > J_{\mathrm{cr}}$, nodes behave very coherently with their states strongly statistically dependent, but with fluctuations appearing at finite correlation lengths when approaching $h = 0$. The model is said to exhibit a long-range order [166]. A discontinuity now occurs in $\langle \mathrm{s} \rangle$ at $h = 0$, as shown on the right-hand plot in Figure 4.2: at the approach of $h = 0$, only a modest change occurs in $\langle \mathrm{s} \rangle$ with only a few nodes undergoing a state change until, at $h = 0$, an abrupt, discontinuous, jump takes place in $\langle \mathrm{s} \rangle$ with nearly all nodes changing their states simultaneously. Such a phase transition is called discontinuous or first-order, because of the discontinuity in $\langle \mathrm{s} \rangle$. As shown on the left-hand plot in Figure 4.2, discontinuous phase transitions take place in the region from $J = J_{\mathrm{cr}}$ to $J = \infty$.

The $1/J$ dependence of the Ising model was already analysed with continuous and discontinuous phase transitions. However, consider starting from $J < J_{\mathrm{cr}}$ and with either constant positive or negative external loading. When J increases beyond the critical threshold and all the way to $J = \infty$, $\langle \mathrm{s} \rangle$ changes continuously, and no phase transitions are encountered, as shown by the two curves above and below $\langle \mathrm{s} \rangle = 0$ on the left-hand plot in Figure 4.2. Depending on the sign of the external loading, $\langle \mathrm{s} \rangle$ moves continuously to either state -1 or state $+1$.

In the Ising model, the network state below J_{cr} is often referred to as the more symmetric phase in the sense that small fluctuations in node states do not change the network's overall appearance, whereas above J_{cr} state fluctuations at $h = 0$ determine the coherent state into which the network spontaneously organises itself. Consequently, the less organised state is more symmetrical than the more organised one. Because nodes at ($J = J_{\mathrm{cr}}$, $h = 0$) spontaneously organise themselves into the more ordered but less symmetrical state, the system's symmetry is said to be broken.

4.4.2 Qualitative Properties of the Random-Field Ising Model

With the Ising model on a regular lattice and under uniform external loading, all phase transitions take place at $h = 0$ [166]. However, when some disorder or irregularity is incorporated into the model by including random components in the external fields, drastic qualitative changes occur in model behaviour [42], [65]. This model, called the random-field Ising model [128], comprising both uniform and random loading components, can demonstrate hysteresis, a phenomenon that occurs in various real systems, e.g., in the magnetisation of materials [98], in plastic materials [72], and in economics [39].

Hysteresis, which has been extensively studied in the random-field Ising model (e.g., [37], [42], [111], [128], [129], [130]), means that the system's time-reversal symmetry is broken, and that the state of the network then depends on its history. Hysteresis exists even in the Monte Carlo dy-

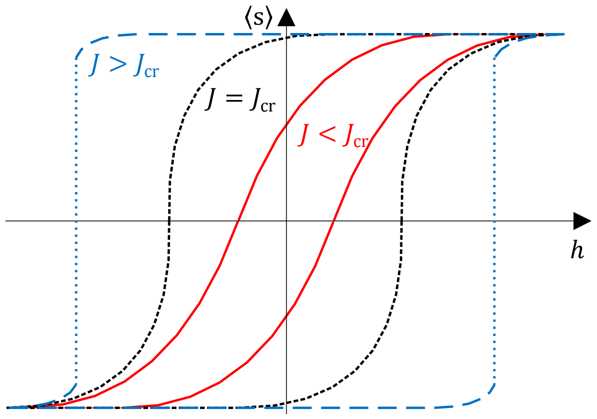

Figure 4.3. Qualitative properties of the random-field Ising model. The average network node state $\langle s \rangle$ is given as a function of h with three levels of interaction parameter J (disorder σ). The average state $\langle s \rangle$ changes continuously when $J < J_{cr}$ ($\sigma > \sigma_{cr}$) (solid red). Phase transition is continuous with the diverging first derivative when $J \cong J_{cr}$ ($\sigma \cong \sigma_{cr}$) (dotted black). Phase transition is discontinuous when $J > J_{cr}$ ($\sigma < \sigma_{cr}$) (dashed blue).

namics of the Ising model under an oscillating loading, because the time variation of the network average state lags behind that of the oscillating field. This occurs when the relaxation time of the Monte Carlo dynamics of the network average state is slower than the frequency of the oscillating field (see, e.g., [4], [5], [6], [27]). The random-field Ising model has shown hysteresis also on an irregular network topology, and the topological properties of an irregular network affect the hysteresis properties [42], [62], [93], [147]. On regular lattices, also network connectivity affects hysteresis properties [124].

Here hysteresis means that as h is adiabatically varied, two separate paths exist for $\langle s \rangle$: one is followed when initialising the model to state $\langle s \rangle \approx -1$ and as increasing h, and the other when initialising the model to state $\langle s \rangle \approx +1$ and decreasing h, respectively. A range of external field values exists now, where simultaneously exists two possible coherent metastable network states (-1 and $+1$) and an unstable state, from which the network always transfers to either of the two coherent states (Figure 4.3). Consequently, the JPD of the random-field Ising model becomes bimodal with the two coherent state expectations corresponding to the two modes of distribution and unstable solutions represented by the improbable states between the two modes.

In the random-field Ising model, hysteresis is stimulated by randomness in node loadings breaking the exact similarities of nodes. The model was studied in [128] at zero temperature for a regular network topology (cubic lattice) of M nodes and with nearest-neighbour interactions. The study of [128] is followed here. Based on Eq. (4.5), the random-field Ising model is obtained simply by replacing the term $h + h_0$ with a term $h + h_m - h_0$, where the random-field loading component h_m for each node m is now included.

Let us now assume that the random components of the external node loadings h_m are independent and identically distributed according to the Gaussian probability distribution with mean 0 and variance σ^2. The hysteresis properties of the random-field Ising model depend on the σ of the Gaussian distribution. This deviation can be interpreted as the size of the fluctuations, or disorder, caused by the random field. A critical value σ_{cr} exists for σ, which is directly proportional to the respective critical value of J, which is here denoted by J_{cr}; $\sigma_{\text{cr}} \sim J_{\text{cr}}$ [123]. In view of the qualitative properties of the random-field Ising model, having $\sigma > \sigma_{\text{cr}}$ is equal to choosing $J < J_{\text{cr}}$ at $\sigma = \sigma_{\text{cr}}$, because then $\sigma > \sigma_{\text{cr}}$; i.e., in this case, node interactions are smaller and transitions smoother. Respectively, having $\sigma < \sigma_{\text{cr}}$ is equal to choosing $J > J_{\text{cr}}$ at $\sigma = \sigma_{\text{cr}}$, which implies that $\sigma < \sigma_{\text{cr}}$; i.e., node interactions are now larger and transitions more abrupt.

The average node state $\langle \mathrm{s} \rangle$ under a varying loading h changes through a series of node avalanches, which in magnetism account for the phenomenon called the Barkhausen noise [35], [111], [123], [128], [129]. Accordingly, though changes in the average node state $\langle \mathrm{s} \rangle$ may appear continuous, they are, in fact, discontinuous with nodes changing states in various-size clusters or domains. In a node, a state change can be triggered by either the external loading or the effect of its neighbouring nodes flipping state. That is, if a node flips its state, e.g., if the external load affecting it becomes large enough, it may cause its neighbouring nodes also to flip their states. The latter nodes may further cause their neighbours to flip their states, resulting in an avalanche of state changes [129], [130] (also [158]).

At the limit of infinite internode interaction, the Ising model exhibits first-order, or discontinuous, phase transitions, i.e., abrupt jumps from one coherent state into another as the uniform external load changes between positive and negative values. Figure 4.3 shows hysteresis properties for the random-field Ising model with $h_0 = 0, J = 1$, and $H = 1$, and with $\langle \mathrm{s} \rangle$ again given as a function of adiabatically changing uniform external field. Figure 4.3 shows $\langle \mathrm{s} \rangle$ in each case first from the initial state $\langle \mathrm{s} \rangle = -1$ on to the final state $\langle \mathrm{s} \rangle = +1$ under an increasing h, and then from the initial state $\langle \mathrm{s} \rangle = +1$ to the final state $\langle \mathrm{s} \rangle = -1$ under a decreasing h.

The Case with $\sigma > \sigma_{cr}$ ($J < J_{cr}$)

With $\sigma > \sigma_{\text{cr}}$, the external loading dominates the interaction term in the random-field Ising model with the node loadings experiencing large fluctuations because of large random components. Because the node states are rather independent of each other, the nodes undergo state changes nearly independently. Therefore, in transitions all avalanches are quite small. Hysteresis occurs with $\langle \mathrm{s} \rangle$ changing smoothly as a function of h, as demonstrated schematically in Figure 4.3 by the two solid red curves; the two transition curves are symmetrical with respect to $h = 0$. [128]

The Case with $\sigma \cong \sigma_{cr}$ ($J \cong J_{cr}$)

At $\sigma \cong \sigma_{\text{cr}}$, hysteresis occurs with continuous phase transitions, and correlations appear at all length scales. Consequently, also avalanches occur at all length scales, their size following a power-law distribution [123]. At this critical point, the disorder σ is just large enough for the state change of each node, on average, to trigger a state change in one of its neighbours [129]. Because these critical phenomena are universal, similar behaviour is expected to be prevalent and independent of the details of individual node interactions [128]. The universal behaviour of continuous phase transitions can be analysed with mean-field theory by analytical approximate calculations, as demonstrated for the Ising model in Subsection 4.4.3 . The above case is shown in Figure 4.3 with dotted black curves. [128]

The Case with $\sigma < \sigma_{cr}$ ($J > J_{cr}$)

With $\sigma < \sigma_{\mathrm{cr}}$, the external loading is similar for all the nodes with only small fluctuations. The uniform loading must be greatly increased before considerable changes take place in $\langle \mathrm{s} \rangle$. When h is finally large enough, nearly all nodes change states at the same time, causing an infinite-size avalanche and potentially a set of smaller avalanches. Thus an abrupt change with a discontinuity in $\langle \mathrm{s} \rangle$ occurs, that is, a discontinuous phase transition. This case is shown in Figure 4.3 with dashed blue curves. The two paths corresponding to increasing and decreasing h are again separate at $h = 0$, and the system exhibits hysteresis with discontinuous phase transitions. [128]

In conclusion, the main difference in qualitative behaviour between the Ising model and the random-field Ising model is hysteresis. The qualitative behaviour of the random-field Ising model is more relevant than that of the regular Ising model to MTNs for two reasons. First, in MTNs external loadings vary between the network nodes, and their topology is irregular. Both facts cause disorder in the network, like random loadings in the random-field Ising model. Hence hysteresis is likely to appear in MTNs. Second, MTNs also tend to behave coherently; consequently, changes may occur rapidly in $\langle \mathrm{s} \rangle$, or even discontinuous phase transitions may take place. In an MTN under heavy external network loading, a discontinuous phase transition may mean a sudden collapse of a finely performing network. Because such an occurrence may be costly and detrimental to the quality of service, it is extremely important to study it. Furthermore, hysteresis together with discontinuous phase transitions is a particularly tricky phenomenon. In real networks, it means that after a network has collapsed, the desired coherent network state cannot be regained simply by returning external loads to their pre-phase-transition values.

4.4.3 Mean-Field Theory for the Ising Model

The Ising model is demanding to study analytically because of difficulties in evaluating the partition function (see, e.g., [164]). Hence some approximation techniques have been developed to analyse phase transitions by analytical calculations. One such widely used technique is the mean-field theory [166], which aims to simplify the interactions of the network nodes. Here the mean-field theory is used to demonstrate briefly how continuous and discontinuous phase transitions arise in the Ising model before these phenomena are analysed in real networked systems in later chapters by numerical simulations.

To give an example of an application of the mean-field theory, let us consider the system studied in Subsection 4.4.1. First, interactions are studied from the standpoint of a single node i. The full conditional of this node, as defined by Eq. (4.2) and written in its notation, is

$$p(s_i|\mathbf{s}_{-i}) = p(s_i|\{s_n\}_{n\in N(i)}) = Z_i^{-1}\exp(s_i h_i^{\mathrm{eff}}), \tag{4.13}$$

where the partition function Z_i for the full conditional of node i has the form

$$Z_i = \exp(-h_i^{\mathrm{eff}}) + \exp(h_i^{\mathrm{eff}}) = 2\cosh(h_i^{\mathrm{eff}}), \tag{4.14}$$

and is obtained simply by summing the two terms resulting from the evaluation of the full conditional in two possible node states, $s_i = -1$ and $s_i = +1$. Next, the states of the neighbouring nodes $N(i)$ of node i are temporarily fixed, or frozen, whereas node i is free to assume either state -1 or $+1$. A regular network structure, e.g., a square lattice, is assumed for all nodes to have the same number of neighbours, N. The state expectation value of node i is now

$$\begin{aligned}\langle \mathrm{s}\rangle_i &= 1 \cdot p(s_i = +1|\mathbf{s}_{-i}) - 1 \cdot p(s_i = -1|\mathbf{s}_{-i}) \\ &= Z_i^{-1}\exp(h_i^{\mathrm{eff}}) - Z_i^{-1}\exp(-h_i^{\mathrm{eff}}) \\ &= \frac{-\exp(-h_i^{\mathrm{eff}}) + \exp(h_i^{\mathrm{eff}})}{\exp(-h_i^{\mathrm{eff}}) + \exp(h_i^{\mathrm{eff}})} = \tanh(h_i^{\mathrm{eff}}),\end{aligned} \tag{4.15}$$

where the hyperbolic tangent function always assumes values between -1 and $+1$. For simplicity, a uniform loading h is assumed for all network nodes. When we also write the average state of the neighbours of node i as $\langle \mathrm{s}\rangle_{n\in N(i)}$ and approximate the states of these neighbouring nodes with this mean value, Eq. (4.15) becomes

$$\begin{aligned}\langle \mathrm{s}\rangle_i = \tanh(h_i^{\mathrm{eff}}) &= \tanh\left[J\sum\nolimits_{n\in N(i)} s_n + H(h - h_0)\right] \\ &\cong \tanh\left[JN\langle \mathrm{s}\rangle_{n\in N(i)} + H(h - h_0)\right].\end{aligned} \tag{4.16}$$

On the left, Eq. (4.16) has the expected state $\langle \mathrm{s}\rangle_i$ of node i and on the right the average state $\langle \mathrm{s}\rangle_{n\in N(i)}$ of its neighbours. In mean-field approximation, these two quantities are assumed equal, i.e., the effect induced by node interactions is the same for each node: $\langle \mathrm{s}\rangle_i = \langle \mathrm{s}\rangle_{n\in N(i)} = \langle \mathrm{s}\rangle$, where $\langle \mathrm{s}\rangle$ is the average state over the whole network. This approximation means that because all fluctuations in node states are omitted, Eq. (4.16) can finally be written as

$$\langle \mathrm{s}\rangle = \tanh[JN\langle \mathrm{s}\rangle_{n\in N(i)} + H(h - h_0)] = \tanh[JN\langle \mathrm{s}\rangle + H(h - h_0)]. \tag{4.17}$$

This equation is a self-consistent expression and cannot be solved analytically, because $\langle \mathrm{s}\rangle$ appears on both sides of the equation. However, it can be studied graphically and with varying J values, which leads to the curves studied on the right of Figure 4.2. Below the critical value $J < J_{\mathrm{cr}}$, the network state average changes smoothly with varying loading, whereas with $J > J_{\mathrm{cr}}$, a discontinuous phase transition and at $J \cong J_{\mathrm{cr}}$ a continuous phase transition appears.

Mean-field studies usually focus on continuous phase transitions and the critical behaviour of models, where many quantities exhibit power-law type behaviour. In such a case, exponents related to power-law decay can be derived from the mean-field approximation (see, e.g., [166]).

5. Methods to Estimate Topology

This chapter introduces dependency (similarity) measures of variables and other methods that can be used to describe network node dependencies, and later in Chapter 8 to help identify MRF graph structures. Mutual information (MI), a special case of the Kullback-Leibler divergence, is a dependency measure based on entropy, a highly endorsed concept in physics and information theory. Furthermore, χ^2-statistics (CSS) is an approximation of MI. All these concepts are based on probabilities; therefore, both MI and CSS are statistical dependency measures. Marginal and conditional versions of the MI and CSS dependency measures can be calculated, respectively, by using either marginal or conditional probabilities.

The concepts of entropy, joint entropy, and conditional entropy are introduced in Sections 5.1–5.2. Relative entropy, or the Kullback-Leibler divergence, is discussed in Section 5.3, its special case, mutual information, in Section 5.4, and the χ^2-statistics approximation in Section 5.5. Alternative dependency measures based on rank-correlation are briefly reviewed in Section 5.6. Finally, Sections 5.7, 5.8, and 5.9 discuss the methods of multidimensional scaling (MDS), Procrustes analysis, and the Frobenius matrix norm, respectively. MDS is a method to derive spatial representations from node dependency values, Procrustes analysis compares two such spatial representations, and the Frobenius matrix norm can be used to normalise two spatial representations into an equal scale.

5.1 Entropy

Entropy is a fundamental concept in physics, originally developed in thermodynamics but later developed into an essential measure in statistical mechanics. For a system consisting of a large number of objects, entropy, as used in statistical mechanics, is the amount of uncertainty in a detailed system state, given its observable macroscopic properties, such as temperature and volume [126]. Hence, in essence, it tells how probability is distributed among all possible system states under such macroscopic conditions: the more evenly probability is spread, the more uncertain the state of the system, and thus the larger the entropy [126].

In information theory, entropy [12], [131] is interpreted as the average information content included in an observation of a random variable, or as the amount of uncertainty about a random variable that is eliminated when an observation has been made [30], [91]. For example, in coding theory, entropy gives the lower bound of the average amount of bits needed to communicate the state of a random variable from source to destination (see, e.g., [30]). However, here entropy is interpreted simply as the uncertainty related to a random variable. For a discrete random variable S_i, it can be specified as

$$H(S_i) = -E_{p(s_i)}[\log p(s_i)] = -\sum\nolimits_{s_i} p(s_i) \log p(s_i), \tag{5.1}$$

where the sum runs over all the possible states of the random variable [30], [77], and $p(s_i)$ is the marginal probability of state s_i. The base of the logarithm is typically 2 or e, and the entropy is given, respectively, either in bits or nats. For continuous variables, entropy can be defined similarly by replacing the summation with integration over the value domain of the random variable. For a given set of possible states, entropy reaches its maximum value when each state is equally probable (uniform distribution) and its minimum value of zero when the probability of a state is one and zero for all other states [30].

5.2 Joint and Conditional Entropies

The joint entropy of two random variables S_i and S_j measures the amount of uncertainty the two-variable system contains. Joint entropy can be defined as

$$H(S_i, S_j) = -E_{p(s_i,s_j)}[\log p(s_i, s_j)] = -\sum_{s_i}\sum_{s_j} p(s_i, s_j) \log p(s_i, s_j), \tag{5.2}$$

where $p(s_i, s_j)$ is the joint probability of the two variables, and the sums run over all joint states of the two variables [30]. Joint entropy achieves its minimum value of zero if the probability of a single joint state is one. Respectively, the maximum value obtains if the probability is evenly, uniformly, distributed for all the joint states. Furthermore, the following inequalities hold: $H(S_i, S_j) \geq H(S_i)$, $H(S_i, S_j) \geq H(S_j)$, and $H(S_i, S_j) \leq H(S_i) + H(S_j)$. In the first two cases, the equalities hold if and only if S_i and S_j are deterministically related. In the third case, the equality holds if and only if S_i and S_j are statistically independent. [30]

Conditional entropy of a random variable S_i, given the value of another random variable S_j, is defined as

$$H(S_i|S_j) = -E_{p(s_i,s_j)}[\log p(s_i|s_j)] = -\sum_{s_i}\sum_{s_j} p(s_i, s_j) \log p(s_i|s_j), \tag{5.3}$$

where both sums run over all the possible values of the two variables [30]. Hence conditional entropy measures the residual uncertainty of a random variable S_i when the value of another random variable S_j is known; $H(S_i|S_j) = H(S_i, S_j) - H(S_j)$. The maximum value of the conditional entropy $H(S_i|S_j) = H(S_i)$ is obtained when the two random variables are statistically independent, i.e., knowledge of S_j does not decrease the uncertainty of S_i. Respectively, the minimum value $H(S_i|S_j) = 0$ is achieved when S_i and S_j are deterministically related.

5.3 Relative Entropy (Kullback-Leibler Divergence)

When two probability distributions, p and q, are associated for the same discrete random variable S_i, the entropies related to p and q, $H[p(s_i)]$ and $H[q(s_i)]$, are simply obtained via Eq. (5.1) by using the respective probabilities. However, when p and q are both involved with S_i, entropy can also be calculated in another way, as cross entropy:

$$H[p(s_i), q(s_i)] = -E_{p(s_i)}[\log q(s_i)] = -\sum_{s_i} p(s_i) \log q(s_i). \tag{5.4}$$

When the two probability distributions are the same, cross entropy equals the entropy given by both distributions: $H[p(s_i), q(s_i)] = H[p(s_i)] = H[q(s_i)]$. In coding theory, cross entropy is interpreted as the average number of bits (nats) needed to communicate the state of a random variable from source to destination if the coding scheme is based on probabilities defined by q instead of (true) probabilities defined by p.

Relative entropy, or the Kullback-Leibler divergence (KLD) [76], [77], measures the difference, or dissimilarity, between two probability distributions, and for a discrete random variable S_i with p and q associated with it, the KLD is defined as [76], [77], [30]

$$\begin{aligned} D_{KL}[p(s_i)||q(s_i)] &= H[p(s_i), q(s_i)] - H[p(s_i)] \\ &= -\sum\nolimits_{s_i} p(s_i) \log q(s_i) + \sum\nolimits_{s_i} p(s_i) \log p(s_i) \\ &= E_{p(s_i)}[\log \frac{p(s_i)}{q(s_i)}] = \sum\nolimits_{s_i} p(s_i) \log \frac{p(s_i)}{q(s_i)}. \end{aligned} \tag{5.5}$$

Hence the KLD is the difference between the cross entropy of p and q and the entropy associated with p. In coding theory, the KLD is thus the expected extra amount of bits (nats) needed to communicate the state of a random variable if the coding scheme is based on q instead of p.

The KLD is always non-negative, $D_{KL}[p(s_i)||q(s_i)] \geq 0$, with the equality holding if and only if the associated probabilities are the same, i.e., $p = q$. The KLD is closely related to conditional entropy. However, whereas conditional entropy is specified between two random variables with their associated probabilities, the KLD is specified for a single random variable between two probability distribution candidates associated with it.

From the definition of Eq. (5.5), it follows that the KLD is asymmetric: $D_{KL}[p(s_i)||q(s_i)] \neq D_{KL}[q(s_i)||p(s_i)]$. Hence though widely applied as a distance measure of two distributions, because of its asymmetry, the KLD is not a true distance measure. The Jensen-Shannon divergence (JSD) is a symmetric alternative to the KLD, and is defined as [87]

$$D_{JS}[p(s_i)||q(s_i)] = \tfrac{1}{2} D_{KL}[p(s_i)||r(s_i)] + \tfrac{1}{2} D_{KL}[q(s_i)||r(s_i)], \tag{5.6}$$

where $r(s_i) = \frac{1}{2}[p(s_i) + q(s_i)]$.

5.4 Mutual Information

Mutual information (MI), a special case of the KLD, measures the distance between the joint probability $p(s_i, s_j)$ of two random variables S_i and S_j and the product $p(s_i)p(s_j)$, which assumes the random variables statistically independent and described by their marginal probabilities. Hence MI can be defined as [30]

$$\begin{aligned} I(S_i; S_j) &= H(S_i) - H(S_i|S_j) = E_{p(s_i, s_j)}\left[\log \frac{p(s_i, s_j)}{p(s_i)p(s_j)}\right] \\ &= \sum\nolimits_{s_i} \sum\nolimits_{s_j} p(s_i, s_j) \log \frac{p(s_i, s_j)}{p(s_i)p(s_j)}. \end{aligned} \tag{5.7}$$

From Eq. (5.7), it follows that MI is symmetric: $I(S_i; S_j) = H(S_i) - H(S_i|S_j) = H(S_j) - H(S_j|S_i) = I(S_j; S_i)$. Hence MI can also be considered a reduction in the entropy of a variable given another variable. [30]

MI is a measure of statistical dependency between two random variables. According to Eq. (5.7), MI is always non-negative, i.e., $I(S_i; S_j) \geq 0$, and assumes the value zero if and only if the two random variables are statistically independent. The maximum value of MI equals the entropy of a random variable and is assumed if and only if the two random variables are deterministically related to each other, in which case $H(S_i|S_j)$ equals zero.

In the literature, MI has been widely applied as a similarity measure, e.g., in image registration [28], [29], [53], [120], [159], statistical language translation [21], and inferring relationships between genes [24], [84]. MI has also been used to select components for mixture models [165] and to study traffic similarities [167], interactions [45], and connectivity [138] in networked systems.

5.5 χ^2-Statistics Approximation

Between two probability distributions p and q the KLD is zero if and only if $p = q$ [30]. Around this point, the χ^2-statistics (CSS) is an approximation of the KLD, and more specifically, of MI [97], [114], [155]. Let us first consider the approximation of the KLD and assume that $p \cong q$. A first-order Taylor series expansion of the logarithm around $p = q$ leads to the approximation $\log[p/q] \approx p/q - 1$. By using this approximation and by adding a term $-p + q = 0$, the CSS approximation of the KLD can be written as [155]

$$\begin{aligned} D_{KL}[p(s_i)||q(s_i)] &\approx \sum\nolimits_{s_i} p(s_i)\left[\frac{p(s_i)}{q(s_i)} - 1\right] - p(s_i) + q(s_i) \\ &= \sum\nolimits_{s_i} \frac{p(s_i)^2 - 2p(s_i)q(s_i) + q(s_i)^2}{q(s_i)} = \sum\nolimits_{s_i} \frac{[p(s_i) - q(s_i)]^2}{q(s_i)}. \end{aligned} \tag{5.8}$$

Similar result is obtained also by making a second-order Taylor series expansion of the KLD around $p = q$. By the same reasoning, the CSS approximation of MI can be written as

$$I_{\text{app}}(S_i; S_j) \approx \sum\nolimits_{s_i}\sum\nolimits_{s_j} \frac{[p(s_i, s_j) - p(s_i)p(s_j)]^2}{p(s_i)p(s_j)}, \tag{5.9}$$

where essentially the same approximations are used as in Eq. (5.8).

5.6 Measures Based on Rank Correlation

Rank-correlation-based dependency measures are here given as an alternative to measures based on information theory. Rank correlation is similar to usual linear correlation, or the Pearson correlation coefficient, but exploits the rank values of samples among all the other samples instead of the sample values themselves [114]. Two well-known rank correlation coefficients are Spearman's rho (SR) [136] and Kendall's tau (KT) [69]. Let us first consider SR in some detail. We use the notation in Chapter 3 with l indexing the observations: $l = 1, \dots, L$. The ranks of the lth observed values of random variables S_i and S_j are denoted by $r_i^l$and r_j^l and the mean ranks of the observations of the variables by $\overline{r_i}$ and $\overline{r_j}$, respectively. SR is now defined as the linear correlation coefficient of the ranks as follows [114]:

$$C_{SR}(S_i, S_j) = \frac{\sum_{l=1}^{L}[(r_i^l - \overline{r_i})(r_j^l - \overline{r_j})]}{\sqrt{\sum_{l=1}^{L}(r_i^l - \overline{r_i})^2 \sum_{l=1}^{L}(r_j^l - \overline{r_j})^2}}. \tag{5.10}$$

KT is based on comparing the orders of the relative ranks of consecutive observations in S_i and S_j to each other. More specifically, first, the relative orders of consecutive observations in both variables are calculated, yielding two label vectors of size $L - 1$. These label vectors categorise each consecutive observation pair $(l + n, l + n + 1)$ such that the $l + n$ th observation is larger than the $l + n + 1$th, or vice versa, or a tie exists between the consecutive points. Then the rankings of the respective consecutive observations of S_i and S_j are compared, and if the relative ranks are the same for both variables, the observation pair is said to be concordant; if the ranks are opposite, the pair is called discordant. If there is a tie in variable S_i, the pair is called an extra S_i-pair, and if the tie is in S_j, the pair is called an extra S_j-pair. The pair is ignored if both variables have ties. Denoting the total numbers of events "concordant," "discordant," "extra S_i-pair," and "extra S_j -pair" by C, D, E_i, and E_j, we can define the KT as [114]

$$C_{KT}(S_i, S_j) = \frac{C - D}{\sqrt{C + D + E_i}\sqrt{C + D + E_j}}. \tag{5.11}$$

Compared to the Pearson linear correlation coefficient, SR and KT have the advantage of being more robust measures, because they use rank information instead of pure data observation values. On the other hand, exploiting only rank information, they may lose some information.

5.7 Multidimensional Scaling

Multidimensional scaling (MDS) [142], [152] is a method to spatially represent the dissimilarity data of variables in a q-dimensional space as a node location map, where the nodes represent the original variables and the internode distances their maximally preserved original dissimilarities. For M variables, all $M(M - 1)/2$ pairwise symmetric dissimilarities can be presented as an MxM proximity matrix. Dissimilarities are calculated from data for each node pair, e.g., by using some of the dependency measures introduced in the previous sections. For two nodes i and j such that $(i, j) \in V$, and where V denotes the set of all node state pairs, the dissimilarity is denoted by $\delta_{ij} = \delta_{ji}$. When the respective similarity σ_{ij} assumes a value between zero and one, the dissimilarity is obtained as $\delta_{ij} = 1 - \sigma_{ij}$ [50].

In essence, MDS is somewhat similar to principal components analysis in that it aims to reduce the dimensions of the original data. However, instead of projecting points from the original high dimensional state space directly into an orthogonal subspace of principal components, MDS searches for a new configuration in the q-dimensional space by trying maximally to preserve the original dissimilarities as internode distances in the q-dimensional space. Hence an original $M - 1$-dimensional presentation is turned into a q-dimensional one, where $q < M - 1$. The choice of q depends on the effective dimension of the data and the goodness of the MDS representations in each dimension. Yet as proximity data is usually presented visually, typically $q = 2$ or $q = 3$. Because of the nature of MTNs, it is here assumed that $q = 2$.

The resulting map of node locations at $q = 2$ is often circular with neighbouring nodes almost equidistant from each other. However, MDS may also produce more structured location map representations with nodes clustered tightly into one or several groups. The internode distances in the mapping are invariant to any rotation, reflection, scaling, or translation of the node location map; hence the map's coordinate values are not unique [50]. In practice, absolute coordinates are usually solved by placing mean coordinate values at the origin. For comparison, two node location maps must then be aligned with respect to each other, a matter considered later in Section 5.8.

Subsections 5.7.1–5.7.2 examine in detail two variants of MDS, metric and non-metric MDS. MDS is applied in diverse ways, e.g., to visualize genes [71], molecules [7], and databases [15], to identify brain areas involved in cognitive tasks [148], and to analyse connections between brain regions [49], [52]. In [67], MDS is applied to positioning sensors in wireless ad-hoc sensor networks and in [61] to estimating the position and velocity of mobile stations. In [13], MDS is used to visualise relations between countries through the properties of their telecommunications networks.

5.7.1 Metric Multidimensional Scaling

Metric MDS aims to search for optimal node coordinate values by minimising a fit criterion between original dissimilarities and internode distances in q-dimensions. Let us denote the coordinate values of node i on a q-dimensional node location map by a vector $\mathbf{x}_i$ and the distance between two nodes i and j by $d_{ij}(\mathbf{x}_i, \mathbf{x}_j)$. If the Euclidean distance measure in the q-space is chosen, the simplest way to formulate MDS is to find the coordinate values that minimise a squared distance between the distances and dissimilarities, $\sum_{i,j}[d_{ij}(\mathbf{x}_i, \mathbf{x}_j) - \delta_{ij}]^2$. However, this criterion assumes a very simplified relation between fitted distances and observed dissimilarities: $d_{ij} = \delta_{ij} + \varepsilon_{ij}$, where ε_{ij} gives errors due to measurements and distortions, because distances in q-dimensions may not correspond exactly with observed dissimilarities [50].

Let us allow a more complex relationship between q-distances and dissimilarities such as a simple linear relation: $d_{ij} = \alpha + \beta\delta_{ij} + \varepsilon_{ij}$. Furthermore, any other parametric functional ship, $f(\delta_{ij})$, can be exploited: $d_{ij} = f(\delta_{ij}) + \varepsilon_{ij}$. In general, the minimised criterion is

$$S_1 = \sqrt{\frac{\sum_{i,j}[d_{ij}(\mathbf{x}_i, \mathbf{x}_j) - f(\delta_{ij})]^2}{\sum_{i,j} d_{ij}(\mathbf{x}_i, \mathbf{x}_j)^2}}, \tag{5.12}$$

where the denominator is included for the criterion to be invariant not only to rotations, translations, and reflections, but also to uniform scaling [50]. This is known as Kruskal's stress-1 criterion [74], [75].

When a functional dependency f is assumed for distances and dissimilarities, the search for optimal coordinate values by minimising the goodness-of-fit criterion becomes a two-stage recursive process (see, e.g., [50]). After the initial coordinate values and their respective distances d_{ij} have been chosen, e.g., through some random process, the first stage is to estimate the parameters associated with the functional relation. For example, in the case of a linear relation, parameter estimates $\hat{\alpha}$ and $\hat{\beta}$ are obtained through linear regression between the distances and dissimilarities. This results in a set of estimated distances, $\hat{d}_{ij}$, called disparities, which are obtained as evaluating the fitted parametric function at the dissimilarity values, $\hat{d}_{ij} = f(\delta_{ij}) = \hat{\alpha} + \hat{\beta}\delta_{ij}$. In the second stage, revised coordinate values are searched for by minimising the criterion of Eq. (5.12) with some optimisation algorithm (e.g., the steepest-decent). If the fit is not adequate, the two stages are repeated.

5.7.2 Non-Metric Multidimensional Scaling

In metric MDS, numerical values of dissimilarities are exploited directly and disparities are obtained from dissimilarities through some parametric function. In non-metric MDS, dissimilarities are not used directly in the search for optimal coordinate values. Instead of absolute values, only the rank order information of dissimilarities is exploited. Hence the resulting coordinate values are invariant to monotonic transformations of the proximity matrix. Non-metric MDS is thus more robust than metric MDS and more practical for real observed dissimilarity values containing measurement uncertainties and other distortions.

In non-metric MDS, node positions are arranged on the map so that the order of distances between the nodes on the map matches best the order of dissimilarities between the respective variables. The aim is again to represent distances with respect to disparities, $d_{ij} = \hat{d}_{ij} + \varepsilon_{ij}$, where ε_{ij} again are fitting errors to be minimised. The map can be fitted, e.g., by minimising the stress-1 criterion of Eq. (5.12). Disparities $\hat{d}_{ij}$ are again related to dissimilarities δ_{ij} through some function, but now any monotonic function is acceptable so that between any two pairs of variables (i, j) and (k, l) the relationship $\hat{d}_{ij} < \hat{d}_{kl} \Leftrightarrow \delta_{ij} < \delta_{kl}$ holds. The method for calculating disparities is called the monotonic regression method [50], [74], [75].

The above entire procedure follows the iterative Shepard-Kruskal algorithm, which is here described only briefly (for details, see, e.g., [75]). First, node coordinates are initialised randomly and their respective distances calculated. The distances are then used to find, by the monotonic regression method, the monotonic relation between disparities and dissimilarities. Then node coordinates, and hence distances, are revised by minimising the goodness-of-fit criterion between distances and disparities. The last two steps are repeated until a satisfying value is achieved for the stress criterion. To avoid MDS get stuck on local minima, the algorithm can be run several times from varying node coordinate initialisations. The configuration yielding the best fit can then be chosen.

5.8 Procrustes Analysis

Because node location map estimates resulting from MDS are unique up to translation, rotation, reflection, and scaling, it is difficult to compare the estimates obtained, e.g., on the basis of different data sets to one another or to a known true node location map. Procrustes transformation is a combined translation, rotation, reflection, and scaling operation, and Procrustes analysis (see, e.g., [50], [57], [127]) is a method to find the best match between two node location maps by performing a Procrustes transformation on one map with respect to the other [50]. Hence Procrustes analysis is a method to evaluate through the final value of the Procrustes criterion the similarity between two node location maps and, further, to better visually compare the maps.

Let us consider two q-dimensional node location maps with the node coordinates presented with size Mxq matrices $\mathbf{V}$ and $\mathbf{Q}$. In $\mathbf{V}$ and $\mathbf{Q}$, the m^{th} rows $\mathbf{v}_m$ and $\mathbf{q}_m$ contain the coordinates of a node m. If the two maps have different dimensions, this can be managed by just adding columns of zeros to the map that corresponds to fewer dimensions [50]. To apply a Procrustes transformation, e.g., to location map $\mathbf{Q}$ with respect to location map $\mathbf{V}$, in Procrustes analysis, the following sum of the squared residuals (SSR) criterion will be minimised

$$R_1 = \min_{\mathbf{Y} \in R(\theta) x \Phi, \mathbf{b}, c} \sum_{m=1}^{M} [\mathbf{v}_m - c\mathbf{Y}(\theta, \varphi)^{\mathrm{T}} \mathbf{q}_m - \mathbf{b}]^{\mathrm{T}} [\mathbf{v}_m - c\mathbf{Y}(\theta, \varphi)^{\mathrm{T}} \mathbf{q}_m - \mathbf{b}]. \tag{5.13}$$

Here $\mathbf{Y}$ is a size qxq orthogonal matrix defined by a rotation $R(\theta)$ and a possible reflection $\varphi \in \Phi$, where Φ is the set of all possible reflections. The size $1xq$ vector $\mathbf{b}$ is defined by a translation and the scalar c by a scaling. The minimum value of the SSR criterion can be solved from the singular value decomposition of $\mathbf{Q}^{\mathrm{T}}\mathbf{V}$ [50], and it is a measure of dissimilarity between the two node location maps.

5.9 Frobenius Matrix Norm

The Frobenius matrix norm for matrices is similar to the Euclidean norm defined for vectors. Hence it is a sort of length assigned for a matrix. For an Mxq matrix $\mathbf{W}$ with elements w_{ij} ($i = 1, \dots, M$ and $j = 1, \dots, q$), the Frobenius matrix norm is defined as [99]

$$||\mathbf{W}|| = \left[\sum_{i=1}^{M} \sum_{j=1}^{q} |w_{ij}|^2\right]^{\frac{1}{2}}. \tag{5.14}$$

In practice, for two matrices each describing a set of point coordinate values in q-dimensions, normalising the values in each with their respective Frobenius matrix norm results in having the points in the q-dimensional space on a comparable scale. For example, let us consider the Frobenius norm for the two matrices $\mathbf{V}$ and $\mathbf{Q}$ defined in Section 5.8. With the mean coordinate values in the two matrices denoted by $\bar{\mathbf{v}}$ and $\bar{\mathbf{q}}$, and the coordinates with the mean coordinate values subtracted as $\mathbf{V}_0$ and $\mathbf{Q}_0$, the Frobenius-scaled coordinates $\mathbf{V}_{\mathrm{F}}$ and $\mathbf{Q}_{\mathrm{F}}$ for $\mathbf{V}_0$ and $\mathbf{Q}_0$, are obtained as

$$\{\mathbf{V}_{\mathrm{F}}, \mathbf{Q}_{\mathrm{F}}\} = \left\{ \mathbf{V}_0 \left[\sum_{m=1}^{M} (\mathbf{v}_m - \bar{\mathbf{v}})(\mathbf{v}_m - \bar{\mathbf{v}})^{\mathrm{T}}\right]^{-\frac{1}{2}}, \mathbf{Q}_0 \left[\sum_{m=1}^{M} (\mathbf{q}_m - \bar{\mathbf{q}})(\mathbf{q}_m - \bar{\mathbf{q}})^{\mathrm{T}}\right]^{-\frac{1}{2}} \right\}. \tag{5.15}$$

Because the mean values are removed from the two matrices, the points in the q-dimensional space are centred at the origin. Because of Frobenius scaling, the scale of the points in $\mathbf{V}_{\mathrm{F}}$ and $\mathbf{Q}_{\mathrm{F}}$ is similar.

6. Methods to Estimate Parameters

This chapter examines general model parameter estimation methods for later use in Chapter 9, which focuses on identifying MRF model parameters and in particular the Ising model. Bayes' theorem provides the foundation on which all the probabilistic inference methods are ultimately based. The theorem works a framework for updating probabilistic information about a system as new uncertain information through measurements arrives. In parameter estimation, Bayes' theorem is applied by combining existing uncertain (a priori) information about model parameters with (likelihood) information obtained by evaluating the parameterised model with a set of data observations. Combining the two pieces of information then leads to updated (a posteriori) information about the model parameters.

Even though Bayes' theorem is not, in fact, directly applied here, it forms the basis for understanding all parameter estimation methods. Consequently, Bayes' theorem is first discussed in Section 6.1. Section 6.2 introduces the maximum a posteriori (MAP) parameter estimation method, which exploits all the properties of Bayes' theorem. Maximum likelihood (ML) estimation, introduced in Section 6.3, can be viewed as a special case of the MAP method without a priori information. Maximum pseudolikelihood (MPL) estimation, examined in Section 6.4, is the most specialised method arising as an approximation of the ML method, and is especially suitable for identifying MRF model parameters.

6.1 Bayes' Theorem

Bayes' theorem (see, e.g., [18], [46]) combines uncertain information obtained through observations with uncertain prior system information to arrive at a posteriori system information. The approach carries all the uncertainty about system parameters and is thus formulated by using probabilities. Hence to estimate model parameters, Bayes' theorem combines the prior probability of parameters with the likelihood function of the parameters evaluated with data observation values. Therefore, the Bayesian approach provides not only point estimates as the most probable parameter values, but also an entire probability distribution as the uncertainty information about the model parameters.

Let us now consider a system model with a multivariable state described by a random variable $\mathbf{S}$ and with a set of model parameters $\boldsymbol{\theta}$. Let there be a set of observations $\{\mathbf{s}^{(l)}\}_{l=1}^{L}$ of size L. Essentially, the Bayesian approach says that our information about parameters is described with a probability distribution, and that hence parameters $\boldsymbol{\theta}$ are random variables. For the probability distribution of model parameters, Bayes' theorem reads [18]

$$\begin{aligned} p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L}) &= \frac{p(\{\mathbf{s}^{(l)}\}_{l=1}^{L}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\{\mathbf{s}^{(l)}\}_{l=1}^{L})} = \frac{p(\{\mathbf{s}^{(l)}\}_{l=1}^{L}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{\int_{\boldsymbol{\theta}} p(\{\mathbf{s}^{(l)}\}_{l=1}^{L}|\boldsymbol{\theta})p(\boldsymbol{\theta})} \\ &= N_C p(\{\mathbf{s}^{(l)}\}_{l=1}^{L}|\boldsymbol{\theta})p(\boldsymbol{\theta}). \end{aligned} \tag{6.1}$$

Here $p(\{\mathbf{s}^{(l)}\}_{l=1}^{L}|\boldsymbol{\theta})$ is the probability of the observation data set when parameter values are $\boldsymbol{\theta}$. This probability is also called the likelihood of parameters, because once given, the data set can be used to find parameter values that correspond to the largest probability of the observation set.

Assuming the observations statistically independent, we can write their likelihood as the product of the probabilities of individual observations: $p(\{\mathbf{s}^{(l)}\}_{l=1}^{L}|\boldsymbol{\theta}) = \prod_{l=1}^{L} p(\mathbf{s}^{(l)}|\boldsymbol{\theta})$. The prior probability of the parameters is defined by $p(\boldsymbol{\theta})$. The denominator in Eq. (6.1), $p(\{\mathbf{s}^{(l)}\}_{l=1}^{L})$, is the marginal probability of the observation set. However, in view of parameter estimation, it is just a constant N_C normalising the probabilities of parameter values, and hence does not affect the relative conditional probabilities of the parameters or their estimation. If necessary, N_C is obtained for continuous (discrete) $\boldsymbol{\theta}$ by integrating (summing) the numerator in Eq. (6.1) over a defined range of parameter values.

6.2 Maximum a Posteriori

With Bayes' theory, the probability distribution of model parameters can be obtained in parameter estimation. There is strong information-theoretical motivation, not discussed here, why best point estimates of parameters should be chosen as mode values of the posterior probability distribution. This distribution fully describes the uncertainty of the chosen point estimates. In MAP estimation [121], point estimates are obtained according to Bayes' theory by choosing the parameter values that correspond to the highest posterior probability. Assuming independent observations, we can formulate MAP parameter estimation as [121]

$$\widehat{\boldsymbol{\theta}}_{\mathrm{MAP}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L}) = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \prod_{l=1}^{L} p(\mathbf{s}^{(l)}|\boldsymbol{\theta})\, p(\boldsymbol{\theta}), \tag{6.2}$$

where the normalisation constant is omitted, because, as a constant, it has no effect on the position of the maximum of the probability distribution or on the relative probabilities of parameter values.

Logarithm is a monotonic function, and thus does not change the values at which its argument assumes maximum or minimum values. When the logarithm is taken from the product of the probabilities of individual observations in Eq. (6.2), the product of the terms transforms into a sum of the terms:

$$\widehat{\boldsymbol{\theta}}_{\mathrm{MAP}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{l=1}^{L} \log p(\mathbf{s}^{(l)}|\boldsymbol{\theta}) + L \log p(\boldsymbol{\theta}), \tag{6.3}$$

from which the parameter values corresponding to the maximum probability are usually easier to obtain.

6.3 Maximum Likelihood

Based on Bayes' theorem, finding MAP parameter estimates requires prior information about parameter values, and if this is not available, a guess should be made about parameter probabilities. However, more often than not prior information is lacking or a good guess is hard to make. Therefore, all information is obtained by observation, and hence only the likelihood of parameters can be used to estimate parameters. In Bayes' theorem, lack of prior information can be interpreted as probability being uniform for all parameter combinations. Because the prior probability $p(\boldsymbol{\theta})$ is a constant, the posterior probability can thus be rewritten as [46]

$$p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L}) = N_C \prod_{l=1}^{L} p(\mathbf{s}^{(l)}|\boldsymbol{\theta})\, N_{C2} = N_{C3} \prod_{l=1}^{L} p(\mathbf{s}^{(l)}|\boldsymbol{\theta}), \tag{6.4}$$

where the two constants are finally combined as $N_{C3} = N_C N_{C2}$. If parameters may assume only values, the prior distribution of constant is not well-defined. However, such improper priors can be used with Bayes' theorem, because only the posterior distribution is then normalised. Without N_{C3}, Eq. (6.4) is called the likelihood $L(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L})$ of parameters $\boldsymbol{\theta}$ [144]: $L(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L}) = \prod_{l=1}^{L} p(\mathbf{s}^{(l)}|\boldsymbol{\theta})$. Because the constants again do not affect the parameter values at which maximum probability occurs, in the ML method parameter estimates are obtained by simply maximising the likelihood or its logarithm [144]:

$$\widehat{\boldsymbol{\theta}}_{\mathrm{ML}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \prod_{l=1}^{L} p(\mathbf{s}^{(l)}|\boldsymbol{\theta}) = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{l=1}^{L} \log p(\mathbf{s}^{(l)}|\boldsymbol{\theta}). \tag{6.5}$$

As a result, best point estimates are again obtained for the parameters, whereas the uncertainties related to these estimates can be studied through Eq. (6.4).

6.4 Maximum Pseudolikelihood

The ML method is a general approach to estimating model parameters, if no prior information is available. However, it is difficult to apply to estimating MRF model parameters, because the conditional probability distribution $p(\mathbf{s}|\boldsymbol{\theta})$ in Eq. (6.5) includes the partition function $Z(\boldsymbol{\theta})$, which, as discussed in Chapter 4, is practically impossible to calculate in general. Therefore, Eq. (6.5) cannot be evaluated, and parameters cannot be estimated. Yet again, we have exceptions, such as the Gaussian MRF model, discussed in Subsection 4.2.3, in which the normalisation constant is easy to calculate, and thus the ML method can be applied.

Maximum pseudolikelihood (MPL) [16], [17] is similar to ML, but the partition function need not be calculated for the whole joint probability distribution; instead the joint probability is approximated as a product of the full conditionals of variables (see Eq. (4.2)), i.e., the conditional probability of a variable given the remaining variables. Though all these conditional probability distributions include the normalisation term, the number of states over which the summation (or integration) runs is equal to the number of possible variable states rather than possible system states. For example, in the Ising model, the normalisation term of a full conditional consists only of the sum of two terms, as shown by Eq. (4.14).

When indexing the variables with subscript m and subscript $-m$ referring to the remaining variables except m, the approximation of the conditional probability in Eq. (6.5) at observation l can be written as [16], [17]

$$p(\mathbf{s}^{(l)}|\boldsymbol{\theta}) \approx \prod_{m=1}^{M} p(s_m^{(l)}|\mathbf{s}_{-m}^{(l)}, \boldsymbol{\theta}). \tag{6.6}$$

The Bayesian posterior probability of parameters in approximate form now reads

$$p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L}) \approx N_{C3} \prod_{l=1}^{L} \prod_{m=1}^{M} p(s_m^{(l)}|\mathbf{s}_{-m}^{(l)}, \boldsymbol{\theta}). \tag{6.7}$$

Without the constant N_{C3}, Eq. (6.7) is called the pseudolikelihood $L_p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L})$ of parameters $\boldsymbol{\theta}$: $L_p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L}) = \prod_{l=1}^{L} \prod_{m=1}^{M} p(s_m^{(l)}|\mathbf{s}_{-m}^{(l)}, \boldsymbol{\theta})$. As a constant, N_{C3} can again be omitted from the parameter estimation, and when taking the logarithm from the pseudolikelihood, MPL parameter estimates are obtained as

$$\begin{aligned} \widehat{\boldsymbol{\theta}}_{\mathrm{MPL}} &= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \prod_{l=1}^{L} \prod_{m=1}^{M} p(s_m^{(l)}|\mathbf{s}_{-m}^{(l)}, \boldsymbol{\theta}) \\ &= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{l=1}^{L} \sum_{m=1}^{M} \log p(s_m^{(l)}|\mathbf{s}_{-m}^{(l)}, \boldsymbol{\theta}), \end{aligned} \tag{6.8}$$

and the pseudouncertainty of the parameters is given by Eq. (6.7).

7. MCMC for Analysis and Evaluation

Observations must be sampled or simulated according to the MRF joint state probability to both verify the methods developed with synthetic data and to predict the behaviour of the identified system. Two essential difficulties are involved here: first, the distribution may be very large in its dimension, and, second, owing to the large system dimension, it becomes nearly impossible to calculate the partition function, because the number of possible states grows exponentially with respect to the number of variables.

Because of high dimensionality, the common sampling methods, such as importance sampling and rejection sampling (see, e.g., [91]) are inappropriate, because they are based on static proposal distributions, i.e., distributions from which the samples are drawn. A static proposal distribution is effective only if it is similar to a true distribution. However, with high dimensionality, such a proposal distribution is very difficult, in fact, impossible to construct, because the number of accepted samples falls exponentially with an increasing dimension [18]. In addition, importance sampling is suitable only for estimating expectation values with respect to a distribution, not for generating actual samples.

Markov Chain Monte Carlo (MCMC) sampling methods are particularly suited for systems with high-dimensional sample spaces. With MCMC, the partition function need not be calculated [18], [91], which makes MCMC suitable for sampling MRF models. In fact, though they require considerable computation time, MCMC methods are ideal for sampling joint probability distributions of high-dimensional MRF models. In the literature, MCMC has been applied extensively as part of many methods, such as Bayesian inference or hypothesis testing, where complex distributions often must be sampled or approximated by sampling from the distribution (see, e.g., [55]). MCMC methods have also been extensively used to simulate MRF models, such as the Ising model (see, e.g., [90], [146]).

This chapter concentrates on sampling by MCMC methods from an identified MRF joint probability distribution. Model validation and system simulation with MCMC are studied in Chapters 8−10 and in Chapter 14. The present chapter focuses on two MCMC techniques: the general Metropolis-Hastings method, discussed in Section 7.1, and its special case, the Gibbs method, introduced in Section 7.2. Convergence and other properties of MCMC methods are examined in Section 7.3.

7.1 Metropolis-Hastings Algorithm

MCMC sampling methods are based on proposal distributions; i.e., a sample is drawn from a proposal distribution and then either accepted or rejected according to some probability. Unlike in rejection sampling and importance sampling, proposal distributions in MCMC methods are not static. In form, the proposal distribution is usually constant, but, e.g., its mean value typically depends on the previous, accepted, generated state (observation). Because adjusting the proposal distribution according to the previous point secures a high acceptance rate even in high-dimensional cases, MCMC methods do not suffer from the dimensionality curse to the extent that, e.g., rejection sampling does [18]. Typically, the proposal distribution is some simple distribution, such as the Gaussian distribution, which is easy to sample.

Let us now examine the Metropolis-Hastings (MH) method [91] and assume that the algorithm starts from some initial state configuration $\mathbf{s}^{(0)}$ of a set of variables to be considered. Now the sample $\mathbf{s}^{(l+1)}$ at iteration $(l+1)$ is drawn from the proposal distribution, denoted by $q(\mathbf{s}^{(l+1)}|\mathbf{s}^{(l)})$, which depends only on the current, accepted state $\mathbf{s}^{(l)}$. Consequently, the sequence of generated samples $\mathbf{s}^{(1)}, \mathbf{s}^{(2)}, \dots, \mathbf{s}^{(L)}$ forms a Markov chain [58], which means that each generated value depends only on the value of the previous, generated and accepted value.

Assume that the goal is to draw samples from the target distribution $p(\mathbf{s}) = \tilde{p}(\mathbf{s})/Z$, where Z is the partition function. In MRF models, $\tilde{p}(\mathbf{s})$ is the product of potential functions, or more generally, it is the distribution except for the partition function term. It is now assumed that $\tilde{p}(\mathbf{s})$ is easy to evaluate at any state $\mathbf{s}$, and that some appropriate proposal distribution $q(\mathbf{s}^{(l+1)}|\mathbf{s}^{(l)})$ has been chosen. In the MH algorithm, a new candidate for the $(l+1)$th sample $\mathbf{s}^{(l+1)}$ is generated from $q(\mathbf{s}^{(l+1)}|\mathbf{s}^{(l)})$, and accepted with the probability

$$A(\mathbf{s}^{(l+1)}, \mathbf{s}^{(l)}) = \min\left[1, \frac{\tilde{p}(\mathbf{s}^{(l+1)})q(\mathbf{s}^{(l)}|\mathbf{s}^{(l+1)})}{\tilde{p}(\mathbf{s}^{(l)})q(\mathbf{s}^{(l+1)}|\mathbf{s}^{(l)})}\right]. \tag{7.1}$$

If the latter term in Eq. (7.1) is larger than one, the generated point is accepted into the sample with probability one; otherwise, the point is accepted according to the probability of the latter term. This probability can be evaluated by drawing a random number u from a uniform distribution over the unit interval $[0, 1]$ and by accepting it if $A(\mathbf{s}^{(l+1)}, \mathbf{s}^{(l)}) > u$ [18].

Because the criterion of Eq. (7.1) does not depend on Z, Z need not be known to be able to draw samples from the target distribution $p(\mathbf{s})$ [18], [91]. Being able to omit Z when sampling a high-dimensional distribution is a significant property, considering that for an MRF model with M nodes of q states, the total number of possible states, and thus the number of terms to be summed over at Z, is equal to q^M.

When we choose a symmetric proposal distribution, i.e., $q(\mathbf{s}^{(l+1)}|\mathbf{s}^{(l)}) = q(\mathbf{s}^{(l)}|\mathbf{s}^{(l+1)})$ for all l, we obtain a special case of the MH algorithm, called the Metropolis algorithm, with the criterion of Eq. (7.1) reduced to [18], [91]

$$A(\mathbf{s}^{(l+1)}, \mathbf{s}^{(l)}) = \min\left[1, \frac{\tilde{p}(\mathbf{s}^{(l+1)})}{\tilde{p}(\mathbf{s}^{(l)})}\right]. \tag{7.2}$$

Consequently, in Eq. (7.2) acceptance of the generated value does not depend on the proposal distribution.

7.2 Gibbs Sampling

Gibbs sampling [18], [91] is a special case of the MH method such that each update is always accepted. In addition, in the Gibbs method, the states of the variables are updated one at a time, Algorithm 7.1. Simple implementation of the Gibbs sampling algorithm.

```
Choose an initial state $\mathbf{s}^{(0)} = [s_1^{(0)}, s_2^{(0)}, \dots, s_M^{(0)}]^{\mathrm{T}}$
For $l = 1, \dots, L$
    For $i = 1, \dots, M$
        Sample $s_i^{(l)} \sim p(s_i | s_1^{(l)}, s_2^{(l)}, \dots, s_{i-1}^{(l)}, s_{i+1}^{(l-1)}, \dots, s_M^{(l-1)})$
    End
    Return sample $\mathbf{s}^{(l)}$
End
```

either in some regular order or randomly. In particular, for the proposal distribution of each node, the node's full conditional (see Eq. (4.2)) is used. In a way, the Gibbs sampling algorithm thus assumes the same as the pseudolikelihood method in Section 6.4, where the joint probability of variables was approximated as the product of the variables' full conditionals. When conditional distributions are used as proposal distributions, the former are assumed simple in form for easy sampling. Indeed, this is often the case, at least when univariate conditional distributions are compared to their respective multivariate joint (target) distributions.

For a precise description, assume an initial state $\mathbf{s}^{(0)}$ for the variables, which are being updated in some specific, regular order. Assume also that each variable has already been updated l times, yielding a state configuration $\mathbf{s}^{(l)}$, which also serves as the initial state configuration for the iteration $l + 1$. Let us now assume that the state s_i of variable i is chosen to be updated next. In Gibbs sampling, the proposal distribution is now selected as $q(s_i^{(l+1)} | \mathbf{s}^{(l)}) = p(s_i^{(l+1)} | \mathbf{s}_{-i}^{(l+1)})$, with $p(\mathbf{s})$ being the target distribution from which samples are to be drawn, and where $-i$ refers to all variables except variable i, and where $p(s_i^{(l+1)} | \mathbf{s}_{-i}^{(l+1)})$ is thus the full conditional of i. In Gibbs sampling, when the state of a variable is being updated through its conditional distribution, the remaining variables are fixed at their most recently updated state values. A sketch of the Gibbs algorithm is shown in Algorithm 7.1 [18].

7.3 Convergence Properties and Other Issues

Usually, two issues must be addressed in generating samples with the MCMC methods. The first is to generate samples so that the sample chain produced converges to the desired target distribution. By using the properties of Markov chains, we can show that the sequence of samples $\{\mathbf{s}^{(l)}\}_{l=1}^{L}$ converges to the desired target distribution $p(\mathbf{s})$ as $l \to \infty$ [91] (see, e.g., [18]). This result holds for both the MH and the Gibbs methods but does not determine the number of MCMC steps required in practice to attain a suitable convergence to a stationary state. The second is the degree of correlation in the generated set of samples and how to deal with it. Correlations arise because consecutive samples in the Markov chain depend on each other through a probability distribution.

The number of steps the Markov chain requires to produce samples from a desired stationary distribution $p(\mathbf{s})$ and the independence of the samples can be solved roughly by using the so-called burn-in period, a period of MCMC steps at the beginning of each simulation during which no samples are recorded. During this period, the Markov chain is expected to converge to the stationary distribution $p(\mathbf{s})$, and consequently, the samples are also rather independent of the initial state. However, the appropriate length of the burn-in period is usually somewhat difficult to choose; fortunately, too long a burn-in period has no other drawbacks than lost computation time.

When appropriate burn-in periods are used before recording any samples, MCMC can be approached in two ways to generate samples from the target distribution $p(\mathbf{s})$. The first is to perform only a single long MCMC simulation, starting from some initial state $\mathbf{s}^{(0)}$, and, after a burn-in period, run until the set of samples $\{\mathbf{s}^{(l)}\}_{l=1}^{L}$ has been obtained. If all consecutive samples are recorded in the sample set, adjacent samples correlate strongly. To avoid this, a certain number of generated samples can be ignored between two recorded samples. This approach has the advantage that the samples have the best chance of reaching a stationary distribution, and that only a single burn-in period is necessary. [91]

A second approach is to record each sample from a separate MCMC simulation with each sample preceded by a burn-in period. Further, to avoid dependence on the initial state, each simulation could be started from a different initial state and thereby improve chances that recorded observations cover samples from the entire state space. This is important especially if the underlying Markov process is not ergodic; i.e., if every state in the state space cannot be reached from every other state. For example, above the critical node interaction J_{cr}, the Ising model is non-ergodic. For non-ergodic processes, when all samples are recorded from a single MCMC simulation, some region in the state space is not reached from the chosen initial state; consequently, this region remains uncovered in the generated sample set. This problem can be avoided if several MCMC simulations are used with each starting from a different initial state. This approach also renders the generated samples rather independent of each other. The drawback is that nearly all computation time is wasted on simulating burn-in periods; consequently, considerable computation time is required to record the same number of samples as in a single long MCMC simulation. [91]

A third approach to MCMC to generate samples is to combine short and long MCMC simulations. For example, a few MCMC simulations could be run, each starting from a different initial state with several samples then recorded during each simulation. Because this compromise circumvents the worst drawbacks, though also the best advantages of the two extreme approaches, it is usually a safe and relatively efficient choice. [91]

One more issue related to MCMC sample generation is worth mentioning here. MCMC methods are somewhat inefficient for exploring the state space when variables to be sampled correlate highly. This inefficiency is caused by the methods' random walk type behaviour, because the width of the proposal distribution is selected according to the smallest variance direction among the variables, and because the proposal distribution is always centred on the previous sample state [18]. Therefore, countless iterations may be required to cover the whole state space in directions of high correlation. However, this problem pertains mostly to continuous variables and less so to models such as the binary-state Ising model.

8. Topology Identification

This chapter deals with the identification of the MRF model graph structure with a data set. Identification is based on statistical dependencies between the network node states. However, since a link on a graph should exist only between conditionally dependent nodes, given all the other nodes, the properties of the conditional dependency must also be taken into account. Besides, other issues must be emphasised. First, some dependency measures are affected by individual node properties; e.g., mutual information is affected by the nodes' marginal entropies. Second, with finite data sets, these measures are random variables, whose uncertainties depend on the number of observations, and, are thus not necessarily directly comparable for different node pairs, e.g., because of different node state distributions or missing data values. The focus here is on MTNs, whose topology is essentially based on a two-dimensional spatial node configuration with their graph structure accordingly projected in two dimensions. However, the topology identification method considered here is not restricted only to such systems.

The MRF model identification methods developed here and their evaluation are discussed in Chapters 8–10. This chapter first applies the methods and tools introduced in Chapters 4–7 to topology identification and then evaluates the methods extensively in various qualitative network behaviour situations with Ising model-based synthetic data. Sections 8.1 and 8.2 introduce the topology estimation method developed in this thesis. Section 8.3 discusses other similar methods reported in the literature. Section 8.4 introduces the methods to evaluate the resulting topology estimates. In Section 8.5, the topology identification method developed here is evaluated with synthetic data representing various qualitative network behaviour situations. Finally, Section 8.6 compares the new topology identification method with those in the literature.

8.1 Network Node Dependencies

Dependency measures of variables were discussed in Chapter 5. For topology identification, mutual information (MI), χ^2-statistics approximation (CSS) of MI, and rank-correlation-based similarity measures can all be used to model the dependencies or similarities of network nodes. However, the above measures, when calculated from data, are uncertain, their uncertainty depending on the number of observations available from the network. Therefore, more robust dependency measures are needed to compare node dependencies estimated with different-size observation sets and node pairs with different properties. The method developed here—instead of applying the above dependency measures directly—exploits the statistical significance that the measure implies nonzero dependency.

The statistical significance of some estimated quantity is calculated by comparing the estimated value to the distribution of values of that quantity, when the values of the quantity are calculated from the same-size data set under a related null hypothesis. With the dependency measures considered here, the dependency value estimated with a data set for two node variables is thus compared to the distribution of values of the respective measure estimated from the same-size data set under the assumption that the two nodes are statistically independent. In topology identification, statistical significances of dependency measures are estimated for each node pair, and the values are then used as similarity values between nodes to derive the topology of the network. In the following subsections, statistical significances are derived for MI, CSS, and rank-correlation measures.

8.1.1 Statistical Significance of Mutual Information

MI was defined in Eq. (5.7) as a statistical dependency measure of two random variables S_i and S_j as $I(S_i; S_j) = \sum_{s_i} \sum_{s_j} p(s_i, s_j) \log \frac{p(s_i,s_j)}{p(s_i)p(s_j)}$. We now derive a more robust dependency measure, the statistical significance of MI (SSMI). The MI value estimated from a set of L observations, $I(S_i; S_j|L)$, must be compared to the distribution of MI values estimated from a same-size, generated data set under the null hypothesis that the two variables are statistically independent. However, we face the difficulty that we do not know the analytical form of the MI distribution, and must, therefore, calculate numerically the probabilities under the null hypothesis.

Let us assume that the two random variables under consideration are statistically independent, i.e., $q(s_i, s_j) = p(s_i)p(s_j)$. Here $p(s_i)$ and $p(s_j)$ are the marginal probability distributions of the variables, derived from the true joint probability distribution $p(s_i, s_j)$, and $q(s_i, s_j)$ is the joint distribution under the assumed null hypothesis. In theory, and when estimated from an infinite data set, MI is obviously zero under the assumed statistical independence: $I^{(0)}(S_i; S_j) = \sum_{s_i} \sum_{s_j} q(s_i, s_j) \log \frac{q(s_i,s_j)}{p(s_i)p(s_j)} = 0$. However in practice, when two variables are statistically independent, and when the marginal probabilities $p^{(\mathrm{obs})}(s_i)$ and $p^{(\mathrm{obs})}(s_j)$ are estimated from a finite data set of L observations, the product of the marginals is not equal to the joint probability $p^{(\mathrm{obs})}(s_i, s_j)$ estimated from the same set of observations: $p^{(\mathrm{obs})}(s_i, s_j) \neq p^{(\mathrm{obs})}(s_i)p^{(\mathrm{obs})}(s_j)$. Hence MI assumes a non-zero positive value.

Under the null hypothesis and L observations, the distribution of MI values is estimated as follows. First, N data sets are generated for S_i and S_j under the null hypothesis assumption, each data set consisting of L observations. These observations are easily generated for discrete variables from the multinomial distribution (binomial distribution for binary variables) by using the original data-estimated state probabilities $p(s_i)$ and $p(s_j)$. Then MI values are estimated from each N observation set, leading to a histogram estimate of the MI distribution $f^{(0)}(I|L)$; alternatively, a more sophisticated method can be used to estimate the distribution. Finally, the SSMI, $\sigma_{\mathrm{MI}}(S_i, S_j)$, is obtained for variables S_i and S_j as

$$\sigma_{\mathrm{MI}}(S_i, S_j) = \int_0^{I(S_i;S_j|L)} f^{(0)}(I|L) dI. \qquad (8.1)$$

The probability that the null hypothesis is now erroneously discarded is $1 - \sigma_{\mathrm{MI}}(S_i, S_j)$. Because estimated from a simulated set of observations, SSMI is a random variable with its uncertainty depending on both N and L. Because the uncertainty of the generated MI values is determined by the fixed L, the uncertainty of the SSMI estimate can be reduced only by increasing N. Throughout the studies in this thesis $N = 2000$. Since SSMI is a probability value, it is clearly interpreted and always lies between zero and one: $0 \leq \sigma_{\mathrm{MI}}(S_i, S_j) \leq 1$. These are desired properties for a similarity measure.

In the literature, SSMI has been applied, e.g., to modelling interactions of perturbed genes in [45], where it is first calculated for all gene pairs, and then a network based on SSMI values is constructed to represent genetic interactions. An estimation of mutual information and SSMI values from finite data sets is proposed also in [141].

8.1.2 Statistical Significance of χ2-Statistics

In Chapter 5, CSS was introduced as an approximation of MI and defined for two variables S_i and S_j by Eq. (5.9) as $I_{\mathrm{app}}(S_i; S_j) \approx \sum_{s_i}\sum_{s_j}[p(s_i, s_j) - p(s_i)p(s_j)]^2[p(s_i)p(s_j)]^{-1}$. The probability distribution of CSS is the incomplete gamma function. For S_i and S_j, the statistical significance of CSS (SSCSS), $\sigma_{\mathrm{CSS}}(S_i, S_j)$, when a CSS value $I_{\mathrm{app}}(S_i; S_j|L)$ is estimated from a data set of L observations, and with D degrees of freedom, is defined as [114]

$$\sigma_{\mathrm{CSS}}(S_i, S_j) = \frac{1}{\Gamma(\frac{D}{2})}\int_0^{I_{\mathrm{app}}(S_i;S_j|L)/2} \exp(-I)\, I^{\frac{D}{2}-1} dI. \tag{8.2}$$

The degrees of freedom parameter D is calculated as follows: $D = q(S_i)q(S_j) - q(S_i) - q(S_j) + 1$, where the function q indicates the number of accessible states of its argument variable. The gamma function, Γ, is defined as $\Gamma(D/2) = \int_0^\infty \exp{(-I)} I^{D/2-1} dI$. Because the only random variable here is the estimate $I_{\mathrm{app}}(S_i; S_j|L)$, unlike to the SSMI estimate, no other uncertainties are related to the SSCSS estimate. On the other hand, the value $I_{\mathrm{app}}(S_i; S_j|L)$ itself is an approximation of the respective MI value $I(S_i; S_j|L)$, hence causing inaccuracies in the similarity estimate.

The advantage of CSS over MI is that, in the case of the null hypothesis, the analytical form of its distribution is always known. Thus SSCSS is much less demanding to estimate computationally than the SSMI. Recently some other approximations to MI and its distribution have also been introduced, e.g., in [56], [63], and [64]. Particularly in [56], the approximation of MI is similar to that of the respective CSS approximation. In [56] MI is approximated as a second-order Taylor series at $p = q$, whereas CSS is obtained by approximating the logarithm function at $p = q$ as a first-order Taylor series (see Section 5.5). Hence the respective dependency test based on the MI distribution approximation, also shown in [56], is similar to the χ^2-dependency test demonstrated here.

8.1.3 Statistical Significances of Rank-Correlation Measures

Approximations of the distributions of rank-correlation measures in the null hypothesis case are also known. For Spearman's rho, a test measure $t = C_{SR}(S_i, S_j)\sqrt{(L-2)/[1 - C_{SR}(S_i, S_j)^2]}$ with the data-estimated correlation value $C_{SR}(S_i, S_j)$ (see Eq. (5.10)) has been constructed to test the hypothesis. In the null hypothesis case of statistical independence, this test measure is approximately distributed according to Student's distribution with $L - 2$ degrees of freedom [114].

With Γ again denoting the gamma function, the statistical significance of an estimated value $C_{SR}(S_i, S_j)$ with test measure t is now obtained as

$$\begin{aligned}\sigma_{\mathrm{SR}}[C_{SR}(S_i, S_j), L] &= \sigma_{\mathrm{SR}}(t, L)\\ &= \frac{\Gamma[(L-1)/2]}{\sqrt{(L-2)\pi}\Gamma[(L-1)/2]} \int_{-\infty}^{|t|} \left(1 + \frac{x^2}{D}\right)^{-\frac{L-1}{2}} dx.\end{aligned} \tag{8.3}$$

Under the null hypothesis of zero statistical dependency, Kendall's tau correlation coefficient $C_{KT}(S_i, S_j)$ (see Eq. (5.11)) is known to be approximately distributed according to the Gaussian distribution, with a zero expectation value and with a variance of $\sigma^2 = (4L + 10)/[9L(L-1)]$ [114]. Consequently, for an estimated value $C_{KT}(S_i, S_j)$, statistical significance is obtained as

$$\sigma_{\mathrm{KT}}[C_{KT}(S_i, S_j), L] = (2\pi\sigma^2)^{-\frac{1}{2}} \int_{-\infty}^{|C_{KT}(S_i, S_j)|} \exp\left(-\frac{1}{2}\frac{x^2}{\sigma^2}\right) dx, \tag{8.4}$$

where the measure $C_{KT}(S_i, S_j)$ now defines the upper limit for the integration.

8.2 Spatial and Graph Representations of Network Nodes

The statistical significance value of a dependency measure, generally denoted here for two node variables S_i and S_j by $\sigma(S_i, S_j)$, is interpreted as a similarity value and transformed into a dissimilarity value $\delta(S_i, S_j)$ as $\delta(S_i, S_j) = 1 - \sigma(S_i, S_j)$. Dissimilarity values for all $M(M-1)/2$ node pairs are given as a symmetric, size MxM, dissimilarity matrix. In principle, these dissimilarities can be perfectly "visually" represented in an $M - 1$ dimensional space as a set of nodes, where the Euclidean distances between the nodes correspond to their dissimilarities. However, by applying the multidimensional scaling methods in Chapter 5 to the dissimilarity matrix, we can reduce this "visual" representation to a q-dimensional approximation ($q < M - 1$), where the true dissimilarity $\delta(S_i, S_j)$ is represented as closely as the dimensionality allows by a continuous-valued distance $d_{ij}(\mathbf{x}_i, \mathbf{x}_j)$ between the pairs of nodes.

To obtain a topology, or a graph presentation, of the nodes, the node dissimilarities represented by a spatial node location map must be turned into binary relations, a set of neighbourhood sets. Here we adopt a uniform thresholding procedure with the distance threshold d_{thr} defined for

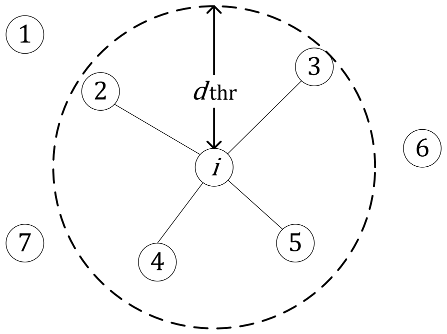

Figure 8.1. Thresholding of a node location map into a graph structure. The construction of a neighbourhood for node i is demonstrated when the threshold distance is defined by d_{thr}. Nodes within the threshold distance d_{thr} from i are neighbours to i.

Algorithm 8.1. Sketch of the MGMN method.

```
1. Initialise $U$ as the set of all node variables: $\forall S_i; i = 1, \ldots, M: S_i \in U$
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 an $MxM$ dissimilarity matrix $D$ so that $D(i, j) = \delta(S_i, S_j)$
6. Apply MDS to $D$ to obtain a low-dimensional spatial representation with node locations $\mathbf{x}_i$
   for all nodes $S_i; i = 1, \ldots, M: S_i \in U$, 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 $d_{\mathrm{thr}}$
8. For all node pairs $(S_i, S_j)$; $S_i \in U, S_j \in U$
9.   If $d_{ij}(\mathbf{x}_i, \mathbf{x}_j) < d_{\mathrm{thr}}$
10.     Set $G(S_i, S_j) = 1$ and $G(S_j, S_i) = 1$
11.  End
12. End
```

pairs of nodes on the node location map. The nodes within the threshold distance from one another are considered neighbours; i.e., if $d_{ij}(\mathbf{x}_i, \mathbf{x}_j) < d_{\mathrm{thr}}$ for nodes i and j, then $i \in N(j)$ and $j \in N(i)$. An example of this procedure is shown in Figure 8.1, where the neighbours of node i are those within the distance d_{thr} from i, and denoted here by undirected links. Although with a given threshold distance the graph structure is unique for a location map, an infinite number of location maps may correspond to a single graph structure, because small changes in node distances do not necessarily affect the thresholding and thus the neighbourhood relations. For convenience, this graph construction is abbreviated below as the MGMN (MDS-based Graph estimation for Markov Networks) method (sketch of it shown as Algorithm 8.1).

8.2.1 Uncertainty and the Effect of the Threshold Distance

The uncertainty of an estimated node location map and its corresponding graph structure is related mainly to the size of the data set, from which node similarities are estimated. Because the node location map is a q-dimensional approximation of the original dissimilarities, this approximation also introduces some further uncertainties into graph estimation. Furthermore, if nodes i and j are neighbours on a graph, the uncertainty related to this neighbour relation can be seen as depending on the gap between the threshold distance d_{thr} and the distance $d_{ij}(\mathbf{x}_i, \mathbf{x}_j)$; i.e., the smaller the value $d_{\mathrm{thr}} - d_{ij}(\mathbf{x}_i, \mathbf{x}_j)$ on the node location map, the more uncertain the corresponding graph link. In addition, the larger the d_{thr} being used, the more uncertain the neighbour relations included. With a small d_{thr}, only the most certain neighbour relations are included, though many true neighbours may be excluded.

Because d_{thr} is important in graph estimation, it may, when chosen incorrectly, later in the modelling phase drastically affect the qualitative properties of an MRF model. If d_{thr} is chosen far too large, the graph is too strongly connected, resulting in false coherent behaviour in the model. If chosen far too small, the graph is too loosely connected, and truly coherent behaviour may be lost in the model. However, choosing d_{thr} slightly incorrectly is not expected to incur these drastic effects; on the contrary, to affect qualitative model behaviour in such a fashion, d_{thr} must be chosen quite poorly. Some approximate information, e.g., the number of neighbours nodes should typically have, is usually available for choosing an appropriate d_{thr}.

If no information is available for choosing d_{thr}, we may try various threshold values and compare the model identification results with the varying threshold values to each other. Yet again, usually no reference is available, to which the resulting graph could directly be compared. However, predictions with the MRF model with the obtained graph structure and parameters can be evaluated by comparing the model predictions to data. Uncertainties in estimated model parameters can also be studied, and such a threshold distance chosen as to minimise these uncertainties. Model predictions and uncertainties are discussed further in Chapter 9 in connection with MRF model parameter identification.

8.2.2 Advantages and Limitations of Graph Construction

A data-based graph estimation scheme is valuable when the true system topology is not known on the basis of domain knowledge, when several pieces of inconsistent topology information exist, or when some other uncertainty about the system topology exists. When no information at all is available about the system topology, the data-estimated graph structure is the best estimate available for such topology. In case of some prior information about a mostly unknown system topology exist, a graph estimate can be applied by complementing it with the prior knowledge by simply adding or removing links. If the system topology is known but partly uncertain, a graph estimate can be used to evaluate the prior topology information and to complement the uncertain parts.

Some networked systems, such as MTNs, may contain several pieces of topology information. MTNs embrace both physical and logical topologies, as discussed in Chapter 2. BTSs and their cells (nodes) interact mostly through logical connections, but also their physical topology is important, because, e.g., the handover between two cells depends on their physical locations and thus affects the states of the respective nodes. Because information from several topologies may be partly inconsistent and partly overlapping, it may be difficult to obtain a single consistent topology to MRF modelling. We could simply choose one of many topologies, but important topology information would then be wasted. However, as the effect of all topologies should be seen in node state data, at least on the scale they matter, the data-estimated graph structure should manifest the joint impact of all these topologies. In addition, the weightings of the topologies in the estimated graph structure should reflect the impact of each topology on the operation of the system. Consequently, we propose that the graph estimate be seen as the best single topology estimate for the MRF modelling purposes.

In two obvious and extreme situations the MGMN method does not work. The first arises when the nodes are completely independent and all dissimilarities equal one. All nodes should be equidistant apart from each other, corresponding to the distance of total dissimilarity. This corresponds to a graph with no links. The second, an opposite situation, obtains when all nodes are highly coherent and dissimilarities equal zero, a situation corresponding to a fully connected graph with links between every node.

8.3 Other Methods to Identify Topology

In the literature, mutual information is often applied as a similarity measure for constructing a spatial presentation of variables with multidimensional scaling. For example, in [1] and [2] MI and MDS are used to search for a spatial configuration to model dependencies in speech and music data, and in [96] and [103] to analyse word relations. In these publications, mostly spatial configurations are studied, but in [1] and [2] also the topology inference from spatial configuration is mentioned. Several other topology construction methods are reported in the literature as well. Perhaps the most straightforward approach to constructing a graph is simply to threshold the node dependencies, modelled, e.g., with mutual information, into neighbourhood relations. Such a graph structure estimation method has been previously applied, e.g., to construct relevance networks of associated genes [24] and will be compared to the MGMN method in Subsection 8.3.1.

In general, two types of methods are used to estimate directed (Bayesian networks) and undirected graphs (Markov random fields). Score-based methods are used mainly to estimate directed graphs (e.g., [36], [51], [149]), but recently a method has also been proposed in [88] for MRF models. Constrained-based methods, generally used for both directed and undirected graphs [82], are based on performing a set of conditional independence tests for nodes. SGS (named after Spirtes, Glymour, and Scheines) and PC (named after Peter Spirtes, and Clark Glymour) algorithms constitute the basic approaches of the constrained-based methods [137]. These algorithms and constrained-based methods in general are discussed in detail in Subsection 8.3.2. The literature contains also some other network topology estimation methods, mainly for specific applications (see, e.g., [20], [109], [132], [151]). The methods discussed in the following subsections are later experimentally compared to the MGMN method in Section 8.6.

8.3.1 Comparison to a Straightforward Approach

The following two questions may well be asked about the MGMN method used in this thesis. First, why is the SSMI or some other statistical significance measure adopted as a similarity measure, instead of applying MI directly? And, second, why is the MDS phase used to construct a spatial presentation and then threshold the topology, instead of thresholding the topology directly from similarity values? This subsection addresses these questions by comparing the MGMN method to a more straightforward approach, which is directly to threshold similarity values, e.g., MI estimates, into a graph structure.

When the statistical significance of a dependency measure, such as SSMI, is used, the resulting probability value lying between zero and one has a clear interpretation, and the same threshold value can be used with the same meaning for networks with varying node state distributions or data set sizes. SSMI values are thus, in principle, independent of the number of observations and on the number of possible states and state probability distributions. Hence even though the data contains missing values for some nodes, all the available joint observations for two nodes can still be used. If MI is used directly, observation of all nodes should in principle be abandoned, because L should be kept the same to have MI estimates that are comparable between node pairs.

Even if L were the same, the varying marginal node state distributions would yet render MI estimates incomparable.

If a networked system is known to have a spatial representation, e.g., in two-dimensions, such as MTNs do, and if the internode distances on this spatial representation are related to node dependencies, then using MDS to reconstruct a spatial representation sounds like a natural phase in estimating graph structure. In MTNs, estimating first the spatial node configuration adds information about two-dimensionality to the node relations. If, in contrast, the underlying networked system does not assume such a spatial configuration, it seems more reasonable directly to threshold the $M - 1$-dimensional similarity values to define the graph structure. However, the problem now is that the decision whether two nodes are neighbours or not, is based only on a single unconditional dependency value estimated for the two nodes. When the information about all other unconditional node dependencies is ignored, it is impossible to decide whether the two nodes, in fact, are neighbours or only connected via other nodes. In other words, all conditional dependency information, information defining graph links in the first place, is omitted (see Chapter 4).

Seemingly, also the MGMN method first ignores unconditional dependency information, because it applies only unconditional dependency values. However, when MDS is applied, conditional dependency information is, in fact, taken into account by using all the dependency information at once to construct a spatial topology, in which internode distances describe which nodes are the closest neighbours to which nodes. As an example, consider a simple one-dimensional system with three nodes A, B, and C. The central node B is a neighbour of both A and C, while A and C are not neighbours. Neighbour relations appear in the data as high dependencies between the nodes, and the goal is to estimate these neighbour relations from the data. If only unconditional dependency values are applied to each node pair directly, a high threshold value may result in a graph where all nodes are interconnected. But when MDS is first applied to dependency values in one-dimensions, B becomes located between A and C on the resulting node location map, and with an appropriate threshold distance, correct neighbourhood relations are recovered.

In some cases, construction of a spatial presentation can add to the robustness of neighbour relations, because all dissimilarity data is then exploited to define neighbourhood relations, instead of a single dissimilarity value being applied to each node pair separately. With small amounts of data, dissimilarities are uncertain, thereby making neighbourhood relations uncertain and sensitive to data observations. But when all dependency values are used at once with the MGMN method, the topology may become less sensitive to random fluctuations in some dissimilarity values.

8.3.2 Constrained-Based Methods to Estimate Graphs

Most constrained-based graph estimation methods have originally been developed to estimate directed graphs for Bayesian networks. For Bayesian networks, the Markov blanket of a node consists of its parents, its children, and its children's other parents. If an estimation method first estimates the Markov blanket rather than removes links between a child node's parents or directs

Algorithm 8.2. Sketch of the SGS algorithm for undirected MRFs ('%' indicates a header of a comment).

```
Initialise $U$ as the set of all node variables: $\forall S_i; i = 1, \ldots, M: S_i \in U$
Initialise the adjacency matrix $G$ as a fully connected graph: $\forall S_i, S_j: G(S_i, S_j) = 1$
Initialise for each $S_i$ its set of neighbours $N(i)$ according to $G$
Specify a significance level parameter $\alpha$ for the dependency test
For all nodes $S_i \in U$
  For all nodes $S_j \in N(i)$
    For all subsets $\mathbf{S}; \mathbf{S} \subseteq N(i) - S_j$
      If $\sigma_{\text{CSS}}(S_i, S_j|\mathbf{S}) < 1 - \alpha$
        Set $G(S_i, S_j) = 0$ % The nodes are conditionally independent
        Update $N(i)$ and $N(j)$ according to $G$
        Break % Exiting the inner-most loop
      End
    End
  End
End
```

the remaining links, it can also be applied to estimating undirected graphs for Markov networks. This is because a Markov blanket obtained as an intermediate step in Bayesian network identification corresponds to the graph structure of a Markov network.

Constrained-based graph estimation approaches conduct a set of conditional independence tests of node variables. These conditional independencies have also been taken into account in the MGMN method, but in an indirect way by constructing a spatial configuration from all node-to-node dependency values at once. In contrast, constrained-based methods conduct a series of conditional independence tests directly for each node by conditioning on subsets of the rest of the nodes. Node pairs found conditionally dependent are considered neighbours.

The most straightforward approach in constrained-based methods is the SGS algorithm introduced in [137], originally proposed for estimating directed graphs for Bayesian networks. However, the algorithm can also be applied to estimating undirected graphs by simply terminating it when it has found a Markov blanket. In this algorithm, a number of conditional independence tests are conducted for each node pair by conditioning on each subset of the rest of the nodes at a time, excluding the two nodes under study. The algorithm starts with a fully connected graph and removes the link between two nodes if they are found conditionally independent according to any of the tests conducted. In other words, if the two nodes are conditionally dependent according to all tests, only then are they considered neighbours (see, e.g., [68]). A sketch of the part of the algorithm seeking the Markov blanket is given as Algorithm 8.2. The algorithm uses Pearson's chi-square test with the test statistics specified by the quantity $\sigma_{\text{CSS}}(S_i, S_j)$ in Eq. (8.2), and denoted here, when conditioning on a set of nodes $\mathbf{S}$, as $\sigma_{\text{CSS}}(S_i; S_j|\mathbf{S})$. When conditioning on a subset of nodes, the respective conditional frequencies are considered in χ^2-statistics.

The problem with the SGS algorithm is that its computation time grows exponentially as a function of the number of nodes, and thus does not scale for large systems. The PC algorithm (see, e.g., [68]), a better scaling version of the SGS algorithm, conducts conditional independence tests

Algorithm 8.3. Sketch of the PC algorithm for undirected MRFs ('%' indicates a header of a comment).

```
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 a fully connected graph: $\forall S_i, S_j: G(S_i, S_j) = 1$
3. Initialise for each $S_i$ its set of neighbours $N(i)$ according to $G$
4. Specify a significance level parameter $\alpha$ for the dependency test
5. Specify a maximum size max_size for the subset size tested
6. For all nodes $S_i \in U$
7.   For all nodes $S_j \in N(i)$
8.     For all subsets $\mathbf{S}; \mathbf{S} \subseteq N(i) - S_j \ \wedge \ \text{size}(\mathbf{S}) \leq$ max_size
9.       If $\sigma_{\text{CSS}}(S_i, S_j | \mathbf{S}) < 1 - \alpha$
10.        Set $G(S_i, S_j) = 0$ % The nodes are conditionally independent
11.        Update $N(i)$ and $N(j)$ according to $G$
12.        Break % Exiting the inner-most loop
13.      End
14.    End
15.  End
16. End
```

only for subsets of nodes, which have fewer nodes than some predefined threshold [137]. Though less accurate than the SGS algorithm, the PC algorithm scales as the network size to the power of maximum subset size and thus suits also for large systems containing some hundreds of nodes. In the sketch given as Algorithm 8.3, the PC algorithm differs from the SGS algorithm only on rows 5 and 8. Some modifications have been suggested to the PC algorithm [3]; besides, some other constrained-based algorithms are available for estimating graph structures (see, e.g., [23], [31], [32], [68], [82], [95]; for details of the PC algorithm, see, e.g., [68]).

Here the Grow-Shrink (GS) algorithm [95] is examined as an alternative graph construction method to the MGMN method. Also the GS algorithm was originally developed for directed graphs, but a version called the Grow-Shrink Markov Network (GSMN) has been proposed in [23] for estimating undirected graphs of Markov networks. Two improvements, mostly of computational efficiency, have been introduced to the GSMN algorithm in [23] and [54], called the GSIMN and DGSIMN algorithms. Though the two also boast other improvements on accuracy, only the GSMN algorithm is discussed here in detail because of its simplicity, and because computational efficiency is not crucial here. This algorithm has been applied in [100] to structure learning of Markov logic networks.

The GSMN algorithm (sketch shown as Algorithm 8.4) is also based on conditional independence tests. In [23], Pearson's chi-square test is applied. In the GSMN algorithm, nodes are examined through in a loop in a specific order, called the visit-order, determined by unconditional dependency tests conducted for each node pair. In particular, unconditional dependency is calculated for each node with respect to all other nodes (lines 3–6 in Algorithm 8.4), and then their average is taken (line 8). In visit-order, nodes are considered in ascending order of average dependency values (line 8). Inside the visit-order loop (lines 15–39), conditional independence tests are conducted between a visit-order node and the rest of the nodes. Inside the visit-order loop, another order, called the grow-order, bears on how the conditional independence tests are exe-

Algorithm 8.4. Sketch of the GSMN algorithm for undirected MRFs [23] ('%' indicates a header of a comment).

```
1.  Specify a significance level parameter $\alpha$ for the dependency test
2.  Initialise $U$ as the set of all node variables: $\forall S_i; i = 1, \ldots, M: S_i \in U$
3.  For all node pairs $(S_i, S_j)$; $S_i \in U, S_j \in U$
4.    % Calculate unconditional dependencies
5.    $p_{S_i,S_j} = \sigma_{\text{CSS}}(S_i, S_j)$
6.  End
7.  % Visit order – sort nodes in ascending order according to unconditional dependencies
8.  Initialise all nodes in $\pi$ so that $i < i'$ if and only if $\text{Avg}_{S_j}(p_{S_i,S_j}) < \text{Avg}_{S_j}(p_{S_{i'},S_j})$
9.  For all nodes $S_i$; $S_i \in U$
10.   $\mathbf{B}^{S_i} = \emptyset$ % Initialisation of the neighbourhood of $S_i$
11.   % Grow order – sort the neighbour candidates of $S_i$ in ascending order according to
12.   % unconditional dependencies
13.   Initialise all nodes except $S_i$ in $\lambda_{S_i}$ so that $j < j'$ if and only if $p_{S_i,S_j} < p_{S_i,S_{j'}}$
14. End
15. While $\pi$ is not empty
16.   $S_i = \text{Next}(\pi)$ % Take the next node from the top of the queue
17.   $\mathbf{S} = \emptyset$ % Initialisation of the subset of nodes
18.   % Grow phase
19.   While $\lambda_{S_i}$ is not empty
20.     $S_j = \text{Next}(\lambda_{S_i})$ % Take the next neighbour candidate from the top of the queue
21.     If $\sigma_{\text{CSS}}(S_i, S_j | \mathbf{S}) \geq 1 - \alpha$ % The nodes are conditionally dependent
22.       $\text{Add}(S_j, \mathbf{S})$ % Add $S_j$ to $\mathbf{S}$
23.     Else
24.       $\text{Remove}(S_i, \lambda_{S_j})$ % Remove $S_i$ from $\lambda_{S_j}$
25.     End
26.   End
27.   % Shrink phase
28.   For all nodes $S_j$; $S_j \in \mathbf{S}$
29.     If $\sigma_{\text{CSS}}(S_i, S_j | \mathbf{S} - S_j) < 1 - \alpha$ % The nodes are conditionally independent
30.       $\text{Remove}(S_j, \mathbf{S})$ % Remove $S_j$ from $\mathbf{S}$
31.       $\text{Remove}(S_i, \lambda_{S_j})$ % Remove $S_i$ from $\lambda_{S_j}$
32.     End
33.   End
34.   $\mathbf{B}^{S_i} = \mathbf{S}$ % The set of neighbours of node $S_i$
35.   % Collaboration phase – make the neighbourhood relations symmetric
36.   For all nodes $S_j$; $S_j \in \mathbf{B}^{S_i}$
37.     $\text{Add}(S_i, \mathbf{B}^{S_j})$ % Add $S_i$ to $\mathbf{B}^{S_j}$
38.   End
39. End
```

cuted. In grow-order, all nodes, except the current visit-order node, are sorted in ascending order of their unconditional dependency values with the visit-order node (lines 9–14).

Inside the visit order, the GSMN algorithm consists essentially of two parts, the grow phase (lines 19–26) and the shrink phase (lines 28–33). In the former, nodes are taken according to their grow-order (line 20), and the conditional dependency test is then conducted between the visit-order node and the grow-order node conditioned with a set of nodes earlier found dependent on the visit-order node (line 21). If the grow-order node is found conditionally dependent on the visit-order node, it is added to the set of nodes conditionally dependent on the visit-order node (line 22). In the shrink phase, all nodes added in the grow phase to the conditional-dependent set of the visit-order node are surveyed and removed from the set if found conditionally independent of the visit-order node, given that the rest of the nodes remain in the conditional-dependent set (lines 28–33). The shrink phase exists for that there may have been some false nodes added to the conditional-dependent set during the grow phase. Finally, the collaboration phase (lines 36–38) ensures the symmetry of the neighbourhood relations (more details of this algorithm in [23]).

In general, constrained-based graph estimation algorithms are suitable only for estimating relatively sparse graphs. The result of a conditional dependency test depends strongly on the chosen value of the significance level (specified by parameter α in Algorithms 3 and 4). The larger the value of the parameter α chosen, the looser the test and eventually the larger the number of the node pairs considered conditionally dependent. Consequently, a large α generally leads to large neighbourhoods; thus densely connected graphs can be constructed with a large α. However, the problem with a large α value is that as more nodes are considered conditionally dependent, the subsets of nodes on which conditional dependency tests are conditioned become larger. In Pearson's chi-square dependency test to test the unconditional dependency of two nodes, a frequency table of size q^2 must be constructed. When conditioning is done on a subset of W nodes, node state occurrence frequencies must be calculated for a total of q^W instances [23]. Thus the number of conditioning instances studied grows exponentially as a function of the number of conditioning nodes. In practice, calculation becomes very difficult for even as small subsets as those consisting of a few tens of nodes. In addition to computational inefficiency, accurate results require a vast amount of data. In general, with conditioning done on W nodes, for an average of one observation for each table cell, q^W observations are required [23]. Although the GSMN algorithm somewhat alleviates these problems by reducing the number of tests conducted, it is still inefficient for obtaining dense graphs with large α values. Hence in practice, the GSMN can produce only sparse graphs. However, the two, more developed versions of this algorithm introduced in [23] and [54], the GSIMN and DGSIMN algorithms, ease these problems slightly by reducing the number of tests to run.

8.4 Evaluation Methods of Topology Estimation

The estimated network topology must be evaluated to confirm that it appropriately represents a true system topology. However, this is difficult because the true topology is unknown. Because this thesis evaluates also a topology estimation method with synthetic data from a known true topology, methods are now studied to help compare an estimated topology with a true topology. These methods can also be exploited with real MTN network data, as examined in Chapters 11–14, where logical and physical topologies are known. When the true topology for defining an MRF graph structure is unknown, which is usually the case and holds also for the real data case here, evaluations must be made through MRF model predictions, as discussed in Chapter 9.

8.4.1 Frobenius Scaling and Procrustes Analysis

Location maps that are identical up to translation, rotation, and reflection lead to identical MRF structures. Applying uniform stretching or shrinking to a map and scaling correspondingly the neighbourhood threshold also yield identical graphs. Consequently, maps identical up to a Procrustes transformation (see Section 5.8) lead to identical graph structures. Therefore, Procrustes analysis [127] (see Section 5.8) is suitable for comparing two maps, and thus two graph structures, by means of searching for a global match between two location maps. Rather than comparing two estimated maps, however, the goal here is to compare one estimated location map to a true map by which synthetic data is generated; yet the methodology remains the same.

When a network consists of tightly bound node groups, the node map may be divided into subnetworks, where nodes within a subnetwork heavily depend on each other, yet are nearly independent of other subnetworks. If Procrustes analysis were applied to two such networks, the results would be poor, because the two maps consisting of nearly independent subnetworks may not show a good global match, even if the subnetworks themselves were identical. In particular, the scaling component of the Procrustes transformation tends to zero, because the translation, rotation, and reflection operations cannot provide a satisfying solution. Therefore, the Procrustes scaling component must be omitted and the node coordinates of both location maps first scaled explicitly with their Frobenius matrix norms according to Eq. (5.15). When we use the notations in Sections 5.8 and 5.9 and, further, denote the coordinate values of node m in $\mathbf{V}_\mathrm{F}$ and $\mathbf{Q}_\mathrm{F}$ by $\mathbf{v}_{\mathrm{F},m}$ and $\mathbf{q}_{\mathrm{F},m}$, we can rewrite the SSR criterion of Eq. (5.13) for $\mathbf{V}_\mathrm{F}$ and $\mathbf{Q}_\mathrm{F}$ as

$$\begin{aligned} R_1 = \min_{\mathbf{Y} \in R(\theta) \mathrm{x} \Phi, \mathbf{b}, c} \sum_{m=1}^{M} & [\mathbf{v}_{\mathrm{F},m} - \mathbf{Y}(\theta, \varphi)^\mathrm{T} \mathbf{q}_{\mathrm{F},m} - \mathbf{b}]^\mathrm{T} \\ & [\mathbf{v}_{\mathrm{F},m} - \mathbf{Y}(\theta, \varphi)^\mathrm{T} \mathbf{q}_{\mathrm{F},m} - \mathbf{b}], \end{aligned} \tag{8.5}$$

where the Procrustes scaling component c used in Eq. (5.13) is now omitted.

8.4.2 Node and Graph Distances

Even though Procrustes analysis may be applied even to networks consisting of nearly independent subnetworks by scaling the location maps with the Frobenius matrix norms and by omitting the Procrustes scaling component, the final SSR criterion may not be a good measure to compare the similarities of partitioned networks that have local rather than global similarities. Because weak dependencies are difficult to estimate with certainty, a good global match between two location maps based on different data sets is highly improbable, even if the two networks were really the same. Therefore, it is natural to look for local similarity measures.

Local similarities can be assessed by comparing internode distances between matching nodes on two location maps. If internode distances $d_{ij}^A(\mathbf{x}_i, \mathbf{x}_j)$ and $d_{ij}^B(\mathbf{x}_i, \mathbf{x}_j)$, with mean values $\bar{d}^A$ and $\bar{d}^B$, are calculated on two maps, abbreviated here as A and B, for all $M(M-1)/2$ node pairs $(i, j) \in V$ (V is the set of all node pairs), the linear (or Pearson) correlation coefficient of the distances can be calculated as

$$C_\mathrm{d}(A, B) = \frac{\sum_{(i,j) \in V} [d_{ij}^A(\mathbf{x}_i, \mathbf{x}_j) - \bar{d}^A][d_{ij}^B(\mathbf{x}_i, \mathbf{x}_j) - \bar{d}^B]}{\sqrt{\sum_{(i,j) \in V} [d_{ij}^A(\mathbf{x}_i, \mathbf{x}_j) \; - \bar{d}^A]^2 \sum_{(i,j) \in V} [d_{ij}^B(\mathbf{x}_i, \mathbf{x}_j) - \bar{d}^B]^2}}. \tag{8.6}$$

To calculate $C_{\mathrm{d}}(A, B)$, the internode distances need not be Frobenius-scaled. This similarity measure is called here the node distance correlation (NDC), and it measures local similarities between two node location maps. Another local similarity measure, the graph distance correlation (GC), is defined by replacing the continuous-valued node distances d_{ij}^A and d_{ij}^B with their respective node graph distances $d_{g,ij}^A$ and $d_{g,ij}^B$.

To reveal similarities between two node location maps or two graph structures, A and B, we may also study the distributions of their internode distances, $p(d_{ij}^A)$, $p(d_{ij}^B)$, $p(d_{g,ij}^A)$, and $p(d_{g,ij}^B)$. Also the joint distributions $p(d_{ij}^A, d_{ij}^B)$, $p(d_{g,ij}^A, d_{g,ij}^B)$, $p(d_{ij}^A, d_{g,ij}^B)$, and $p(d_{g,ij}^A, d_{ij}^B)$ can be studied. Furthermore, conditional distributions, e.g., $p(d_{g,ij}^A | d_{g,ij}^B)$ and $p(d_{ij}^A | d_{g,ij}^B)$, can be used to compare two graph structures. Later in this chapter when the MGMN method is evaluated, $p(d_{ij}^A)$ is compared to $p(d_{ij}^A | d_{g,ij}^B = 1)$, and also $p(d_{ij}^A | d_{g,ij}^B)$ is studied.

8.5 Results with MCMC-Generated Synthetic Data

In this section, location map and graph structure estimation methods are extensively evaluated with synthetic data generated by MCMC methods from the Ising model with a given network topology. In particular, the accuracy and limitations of the MGMN topology estimation method are studied while network behaviour is varied qualitatively. Different qualitative behaviours are generated in simulations by varying the parameter values of the Ising model, and defined network coherence measures are used to describe this qualitative network behaviour. Topology estimation results are studied as functions of coherence measures. SSMI is used as a similarity measure within the MGMN method, and non-metric MDS is applied to construct node location map estimates.

In the following, synthetic data generated with the Ising model with different parameter values is discussed in Subsection 8.5.1. The same data is again discussed in Section 9.4 when the performance of parameter estimation is evaluated. Network coherence is characterised with coherence measures in Subsection 8.5.2, whereas Subsection 8.5.3 reviews the results of topology estimation. Subsection 8.5.4 studies the effect of data characteristics on topology estimation, analysing particularly the effects of the number of network observations, network size, node neighbourhood size, and node loading distribution.

8.5.1 MCMC-Generated Synthetic Data

To generate synthetic data with the Ising model, graph structure and model parameters must first be fixed. Network topology is obtained here by first generating a node location map on a two-dimensional plane with x- and y-coordinates for each node randomly drawn from a uniform distribution with interval $[0, 1]$. Varying graph structures are then derived from the obtained node location map by applying a uniform threshold for the internode distances. In most studies here, a network of 30 nodes is used with the threshold distance chosen so that each node has an average of 8.8 neighbours (node location map shown in Figure 8.2).

As discussed in Chapter 3, topology affects the behaviour of a networked system and hence the qualitative properties of MRF models. The synthetic topology chosen here is generated to re-

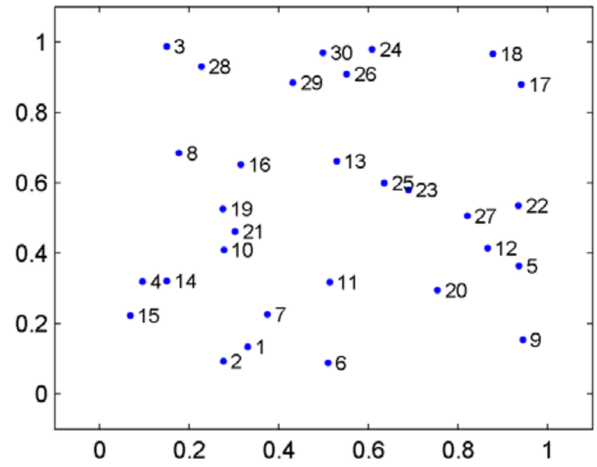

Figure 8.2. Node location map of a synthetic network of 30 nodes.

semble an MTN logical topology. However, comparison with an MTN logical topology is difficult in that both this topology and that of physical node locations are expected to affect the behaviour of an MTN, and such combined topology is unknown. The method of generating first a node location map and then thresholding the respective graph structure is somewhat similar to the MGMN method, in which a node location map is first estimated and the respective graph structure is then obtained by thresholding.

Throughout Chapters 8–9, the performance of the MRF model identification method with synthetic data is evaluated using only one sample of node location map, the one given in Figure 8.2, but several threshold values are tested to define the respective graph structures. Once an MRF model graph structure has been fixed, the three parameters of the Ising model, especially the interaction parameter J, determine its qualitative behaviour, as discussed in Chapter 4. Therefore, MRF model identification is studied by varying the values of J between 0 and 0.2 at 0.01 intervals, resulting in analysis of a total of 21 model parameterisations. The other two Ising model parameters, $H = 0.6$ and $h_0 = 0.7$, are kept constant.

Because the focus here is on node state probability distribution, and because node loadings affect state behaviour, we must, before any MCMC-generated synthetic state data, choose node loadings $\{h_m^{(l)}\}_{m=1}^M$ at each observation $l = 1, \dots, L$. Because node loads are here generated randomly, the model applied is the random-field Ising model, but with a zero uniform loading component. In particular, each $h_m^{(l)}$ is drawn identically and independently according to a chosen distribution. The synthetic node state data $\{s_m^{(l)}\}_{m=1}^M$ for each l is then generated with the Gibbs sampling method, described in Chapter 7. The ensemble scheme is adopted: only a single observation is generated in each MCMC simulation, which is initialised to a random state configuration and run through a constant-length burn-in period. The ensemble scheme ensures better chances for a non-ergodic system's entire state space to be sampled effectively with the generated observation set. Although the infinite-size Ising model is non-ergodic, for finite-size networks it is ergodic. However, because of the coherence of simulated networks, it may take an extremely long time to sample the entire state space in a single simulation, even for a finite system of 30 nodes. Thus, in practice, the model may appear non-ergodic and require ensemble sampling.

Throughout Chapters 8–9, a reference network is studied with a network of 30 nodes, shown in Figure 8.2, and with a graph structure corresponding to an average of 8.8 neighbouring nodes per node, denoted here by $A = 8.8$. With each J, a set of 270 network observations is generated, $L = 270$. Here A and L are chosen such that they are similar to the corresponding values of a real MTN considered later. Each ensemble observation is generated with a burn-in period of 500 complete rounds of node state updates (each node updated once). The set of node loadings is the same for each set of parameters, where $h_m^{(l)}$ is i.i.d. according to $\mathrm{Uni}[0, 1]$. This configuration forms the reference case studied in Subsections 8.5.2 and 8.5.3 with tools given in Sections 8.1–8.4. In Subsection 8.5.4, the data characteristics are altered to study their effects on topology estimation by comparing the results with the reference case.

Throughout this chapter, to reduce variation in the results due to the sample set and the stochastic aspects of SSMI estimation, three generated data sets are used in Chapters 8–9 with each combination of data and model parameters. Results are thus given in medians of the three sets, unless stated otherwise. In addition, to avoid local minima, MDS is always run 20 times from varying initial node coordinate values, and the node location map giving the smallest stress-1 value according to Eq. (5.12) is chosen.

8.5.2 Coherence in Network Data

Within a generated data set, coherence of network node states is determined by model parameters. Because with the Ising model, coherence is largely determined by the interaction parameter, J itself can be considered a measure of overall network coherence. However, since also various network connectivities are studied here by altering the graph distance threshold, J alone does not determine the level of coherence, and other coherence measures must be considered as well. Because the statistical significance-based similarity measures, introduced in Section 8.1, measure the dependency of node variables, another overall coherence measure is gained by taking the average of the dependency measure values over all node pairs. For example, SSMI values averaged over all node pairs within a given data set constitute one such coherence measure, abbreviated here to ASSMI.

Another coherence measure, defined specifically for the Ising model, though similar measures can also be defined for other MRF models, is based simply on comparing the ratio of the interaction and the external load terms of the Ising model within a given data set, and is defined as

$$R = \frac{\sum_{k=1}^{K} \left| \sum_{n \in N(k)} s_n \right|}{\sum_{k=1}^{K} \left| h_k - h_0 \right|}, \tag{8.7}$$

where $k = 1, \ldots, K$ ($K = LxM$) goes through all M nodes in all L observations, and where h_k is the loading of node k. Neighbourhoods $N(k)$ are the estimated ones. With synthetic data, h_0 may assume either a true or an estimated parameter value. Because with synthetic data it is an exception to know the true parameter value, only an estimated value is used in the following. When we choose an estimated value, such that $h_0 = h_0'$, we denote the coherence measure by R'. When we multiply R' with the absolute value of the ratio of the estimated interaction and exter-

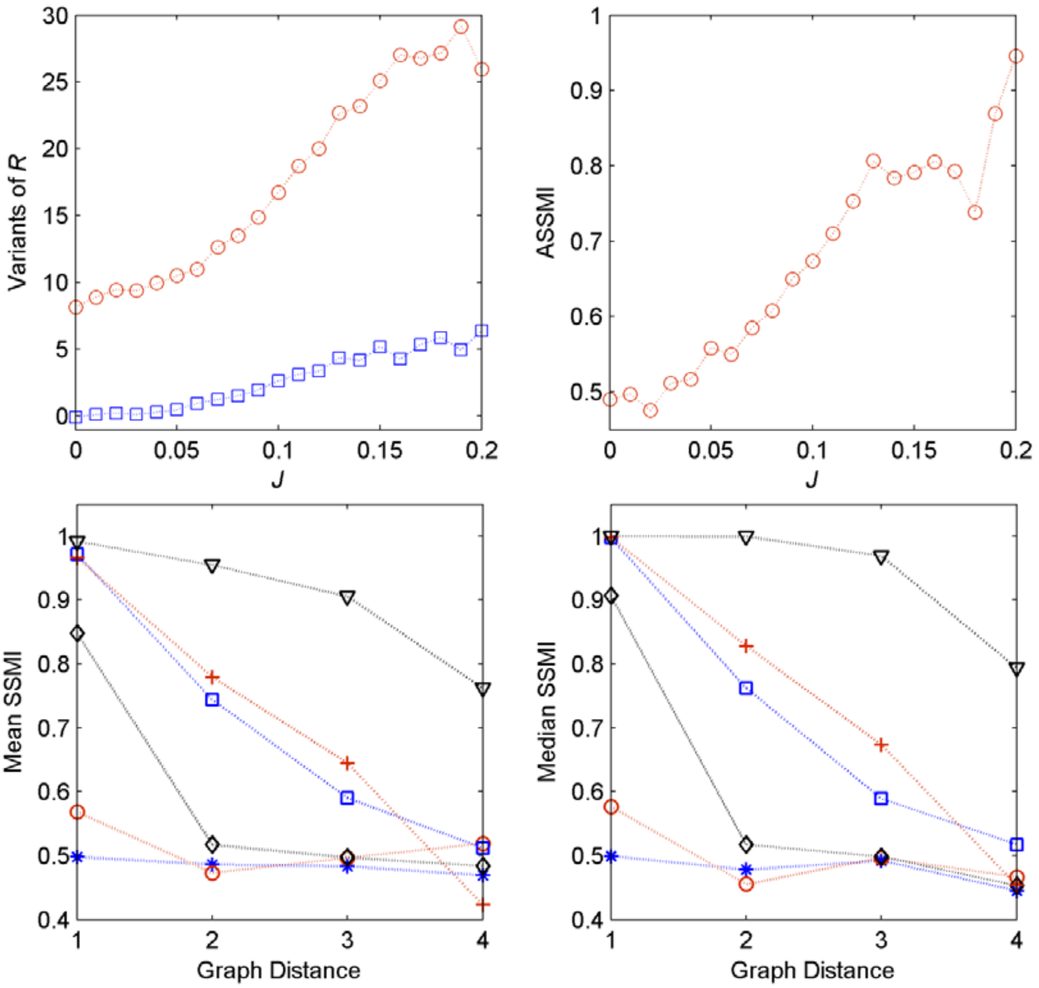

Figure 8.3. Network coherence in the reference case of a synthetic network of 30 nodes. Top-left plot shows R' (circles) and R_N (squares) as functions of J, while the top-right plot shows ASSMI as a function of J. Mean (bottom-left) and median (bottom-right) values of SSMI are given as functions of 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). Calculation: R' and R_N are median values over respective measures calculated for the three ensembles. ASSMI is the median value over the respective measure calculated for the three ensembles as averages over pairwise SSMI values. With each graph distance, 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.

nal load parameters, $|J'/H'|$, we obtain a normalised version R_N of the coherence measure. R_N is useful for studying conditional node state probabilities, because in the Ising model, it gives the ratio of contributions of interaction and external load terms to conditional probabilities.

The functional interrelatedness of the above overall coherence measures is demonstrated in Figure 8.3 with the reference data set by showing R', R_N, and ASSMI as functions of the true value of J. At least with this data set, coherence measures appear to have similar non-linear, nearly monotonic, functional relationships to J. Furthermore, Figure 8.3 gives the mean and median values of the SSMI measure as functions of the true graph distance of nodes in six out of the 21 parameterisations, corresponding to the J values of 0, 0.04, 0.08, 0.12, 0.16, and 0.20 (more on these six parameterisations in Chapters 8–9). SSMI values show how the coherence between two nodes depends on their graph distance when the model exhibits varying qualitative behaviour.

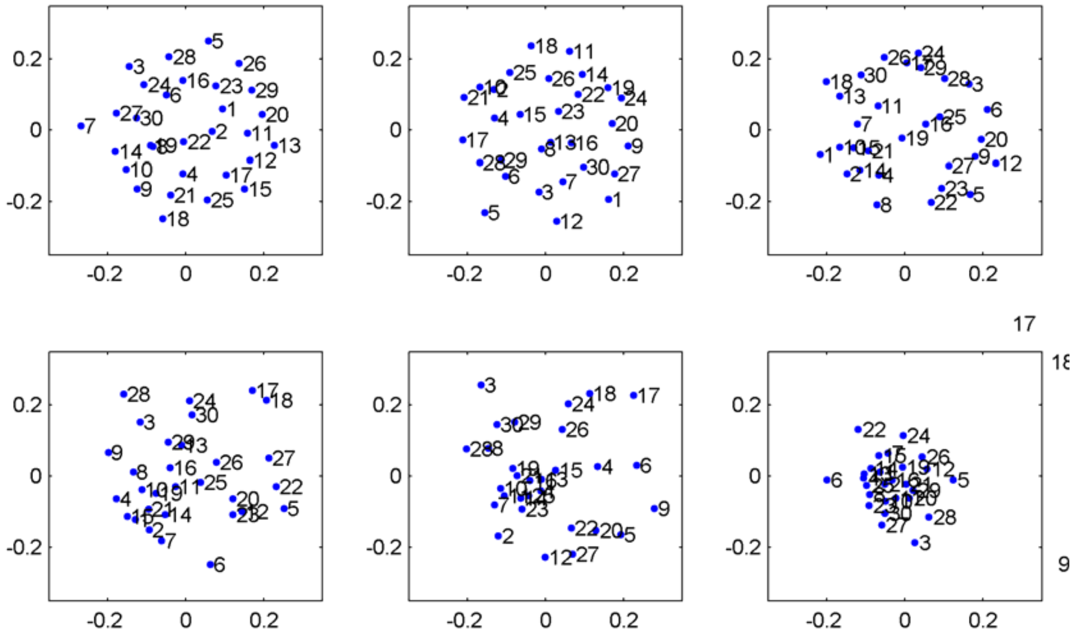

Figure 8.4. Estimated node location maps for a synthetic network of 30 nodes. From top-left to bottom-right, the maps correspond to the following J values: 0.02, 0.04, 0.08, 0.12, 0.16, and 0.20. Calculation: each map presents a single randomly picked ensemble.

The mean and median SSMI values are similar, and coherence according to SSMI is always greatest between neighbouring nodes, as expected. Also, the larger the J, the larger the overall level of coherence, and the slower the decrease in coherence as the graph distance increases.

8.5.3 Location Map and Graph Structure Estimates

This subsection discusses the reference case topology identification results with the MGMN method with all the 21 model parameterisations. In each case, the node location map obtained is Procrustes-transformed with respect to the true location map shown in Figure 8.2 so that networks can be visually compared and quantitatively analysed with the SSR criterion. The location maps are first scaled with their Frobenius norms, and then a Procrustes transformation is applied without the scaling component. Location map estimates are shown in Figure 8.4 with six selected parameterisations.

Obviously, visual comparison is difficult and subjective. The final Procrustes SSR criterion, given in Figure 8.5 as a function of R' and J, is one measure for comparing the similarities of estimated and true node location maps. The similarity is highest when coherence is high, though not excessively so. With the smallest R' values and with the corresponding J values, the nodes are nearly independent, and the data is not informative about node locations. With increasing coherence, the data becomes more and more informative about node dependencies, and node location map estimates begin to approximate the true node location map.

In the reference network, node location map estimates are thresholded into respective graph estimates by following the node distance thresholding scheme in Section 8.2 so that for each graph

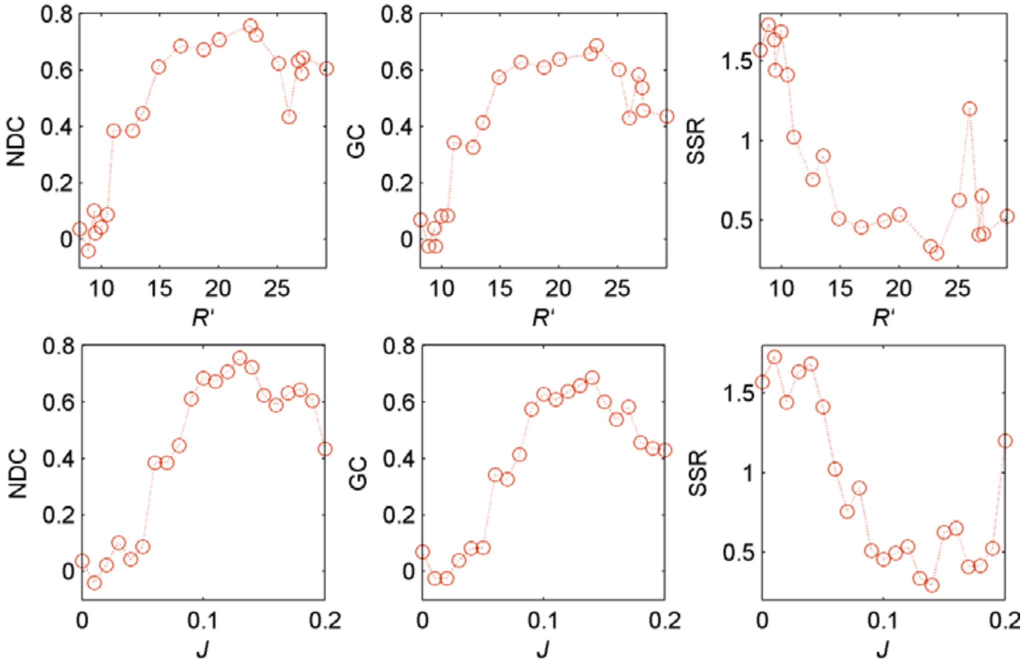

Figure 8.5. Similarity measures between estimated and true node location maps and their respective graph structures with a synthetic network of 30 nodes. NDC, GC, and SSR are shown as functions of R' (top row) and J (bottom row). Calculation: with each J, and respective R', measure values are median values over respective measure values calculated with each ensemble.

estimate $A = 8.8$. Node distance (NDC) and graph correlation (GC) measures with each J are also given in Figure 8.5. As expected, these measures give information very similar to the SSR criterion. The conclusion is that for the MGMN method to perform well, network nodes can neither act as a group of (nearly) independent nodes, as in the case of very low R', nor as a single network entity with very high coherence R'. However, the motivation to analyse networks is usually that the network is indeed considerably coherent yet does not behave as a single unit; i.e., the system's coherence is such that the MGMN is at its best. According to Figure 8.3, topology estimation seems most successful when the SSMI is the steepest function of the graph distance; i.e., when the coherence of true graph neighbours differs most distinctly from coherence between non-neighbours.

Though the previous studies already largely clarified the performance and limitations of the MGMN method in the reference case, for completeness, the rest of this subsection briefly examines the node distance distributions. Figure 8.6 compares the distribution of distances between the nodes on the estimated location map to true graph neighbours and all network nodes in the six selected parameter cases. Obviously, the more different the two distributions, the less randomly neighbourhoods are selected, and the better the graph structures. In addition, according to Figure 8.6, the best neighbourhood estimates are obtained with parameterisations corresponding to intermediate coherence levels, where the two distributions are considerably disparate.

To quantify the similarities of the distributions in Figure 8.6, the Figure 8.6 further shows the Kullback-Leibler divergence (KLD), its symmetric version, the Jensen-Shannon divergence (JSD), and the CSS approximation of the KLD with all the 21 model parameterisations (see

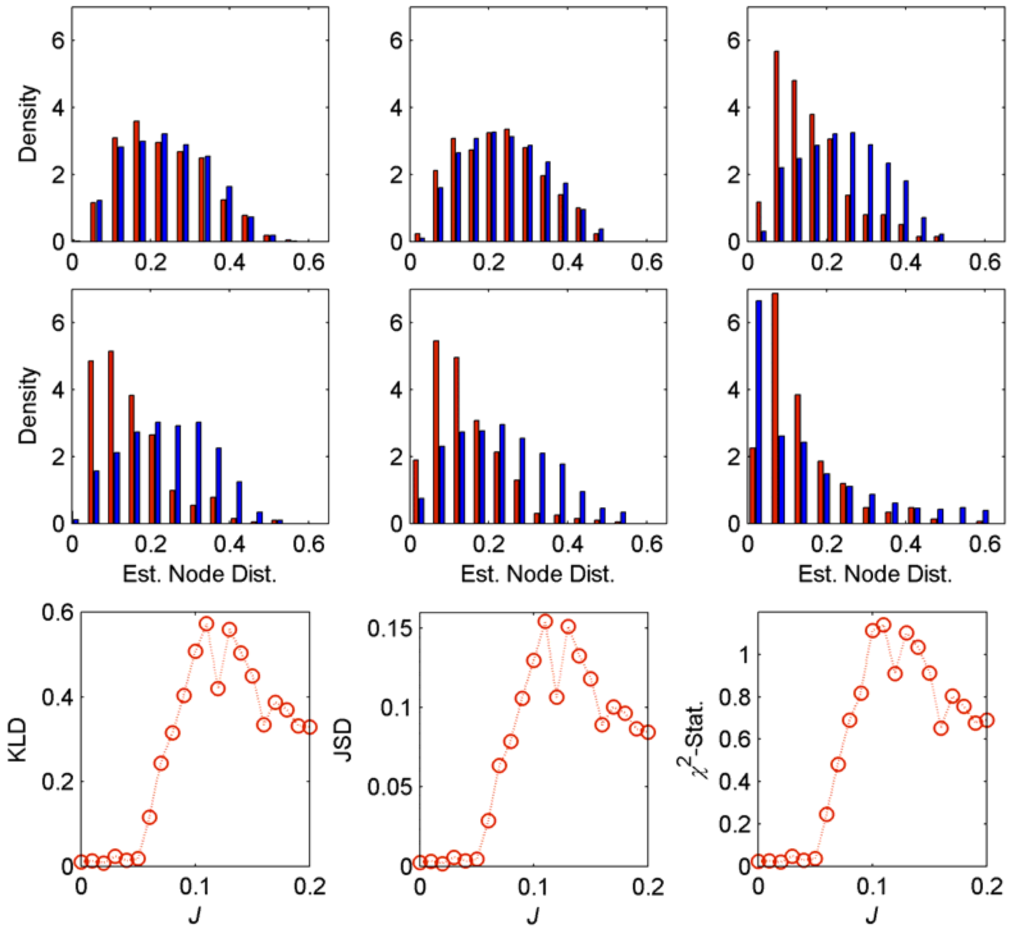

Figure 8.6. Distributions of internode distances in estimated node location maps of true graph neighbours (left-hand-side bars) and of all nodes (right-hand-side bars), and corresponding distribution similarity measures (bottom). The distribution plots from top-left to middle-right are shown for the following J values: 0.02, 0.04, 0.08, 0.12, 0.16, and 0.20. KLD (bottom-left), JSD (bottom-centre), and CSS approximation of KLD (bottom-right) are shown for all the 21 parameterisations as functions of J. Calculation: with each J, histograms are calculated by using all the data of the three ensembles with bars defined at equal intervals. The histograms are then represented in the form of probability densities. The number of bars is the same for each J, but the range changes according to the distance values. Similarity measures are obtained from the distributions.

Chapter 5). Evidently, the three measures provide very similar results, which clearly accord with all previous results. Further study of the histogram estimates of the distribution of true graph distances of estimated graph neighbours in Figure 8.7 reveals that the results are still similar to all previous results; in particular, bar heights at graph distance one somewhat follow the form of the plots in Figure 8.6.

8.5.4 Effect of Data Characteristics

This subsection examines how data characteristics, i.e., the type of node load distribution, node neighbourhood size, data set size, and network size affect topology identification. Furthermore, the quality of synthetic data is ascertained by studying the impact of the length of the burn-in period in MCMC data generation. The functional relationship between the overall coherence measures, R', ASSMI, and J, and the graph correlations as functions of R' are of main focus here.

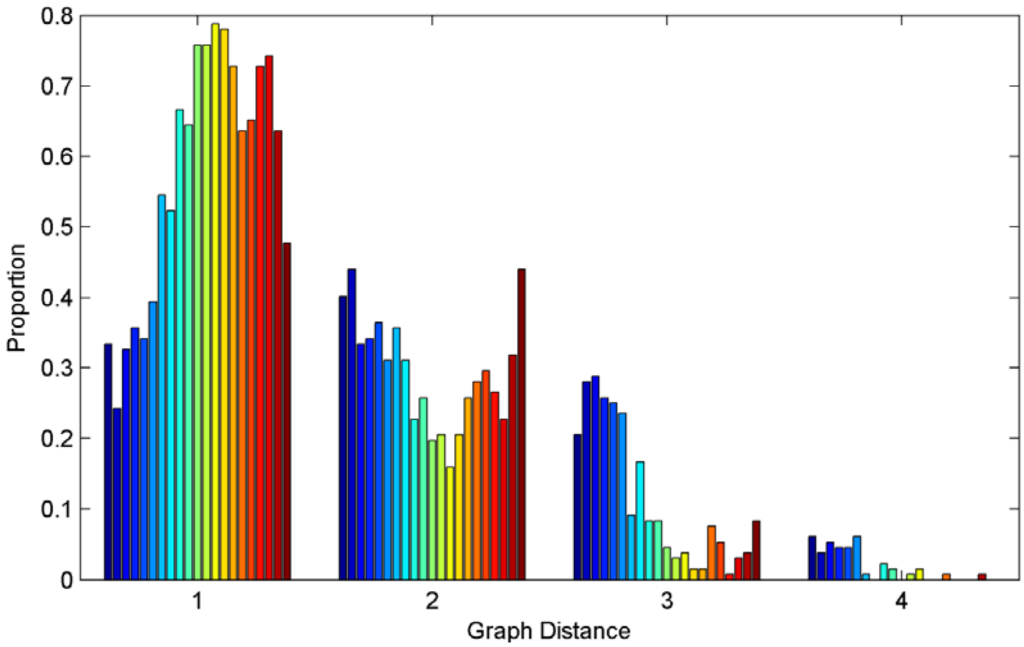

Figure 8.7. Histograms of true graph distances of estimated graph neighbours. Bars with each graph distance from left to right correspond to increasing J values from 0 to 0.20 with even intervals of 0.01. Calculation: with each J, histograms are first calculated for all three ensembles, then the hits at each bar are summed over the hits in the three ensembles, and finally the number of hits at each bar is divided by the total number of hits in the three ensembles together.

First, the distribution from which node loadings are randomly drawn is varied. Besides the uniform distribution $\mathrm{Uni}[0,1]$, of the reference case, normally distributed node loadings with $\mathcal{N}(0.5, 0.25^2)$ (mean 0.5, variance 0.25^2) and exponentially distributed node loadings with $\mathrm{Exp}(0.58)$ (mean 0.58) are studied. The results with these three loading distributions are shown in Figure 8.8. With exponentially distributed loadings, ASSMI assumes a slightly different functional form, and R' assumes clearly smaller values than with the other two distribution types. The differences may result from the smaller median value, which is 0.4 with the $\mathrm{Exp}(0.58)$ distribution. The graph correlation shows rather high and similar values with all three load distributions, implying that the MGMN method works well regardless of the type of node load distribution.

Next the neighbourhood size is varied as $A = 6.8$, $A = 8.8$ (reference), and $A = 10.8$. The results are shown in Figure 8.8, where the range of R' differs slightly from the rest of the figures. Obviously, the larger the J value and the network connectivity, the larger the coherence with the values of R' and ASSMI being large. In fact, with $A = 10.8$ and a few largest J values, the coherence is so great that nearly all nodes appear constantly in equal states, entailing problems with topology estimation. Among the cases with $J > 0.16$, only one ensemble at $J = 0.19$ yielded topology estimates, this value corresponding to the GC value, which is being clearly distinct from the remaining values at $R' \approx 33$. Otherwise, topology is successfully identified in each case, though small A values tend to yield somewhat better graph structures.

The effect of data size is tested with $L = 270$ (reference) $L = 540$, and $L = 1080$. Figure 8.8 shows that both ASSMI and GC values are affected. However, R' remains practically unchanged, because changing L does not change average node states or loadings. The ASSMI increases with

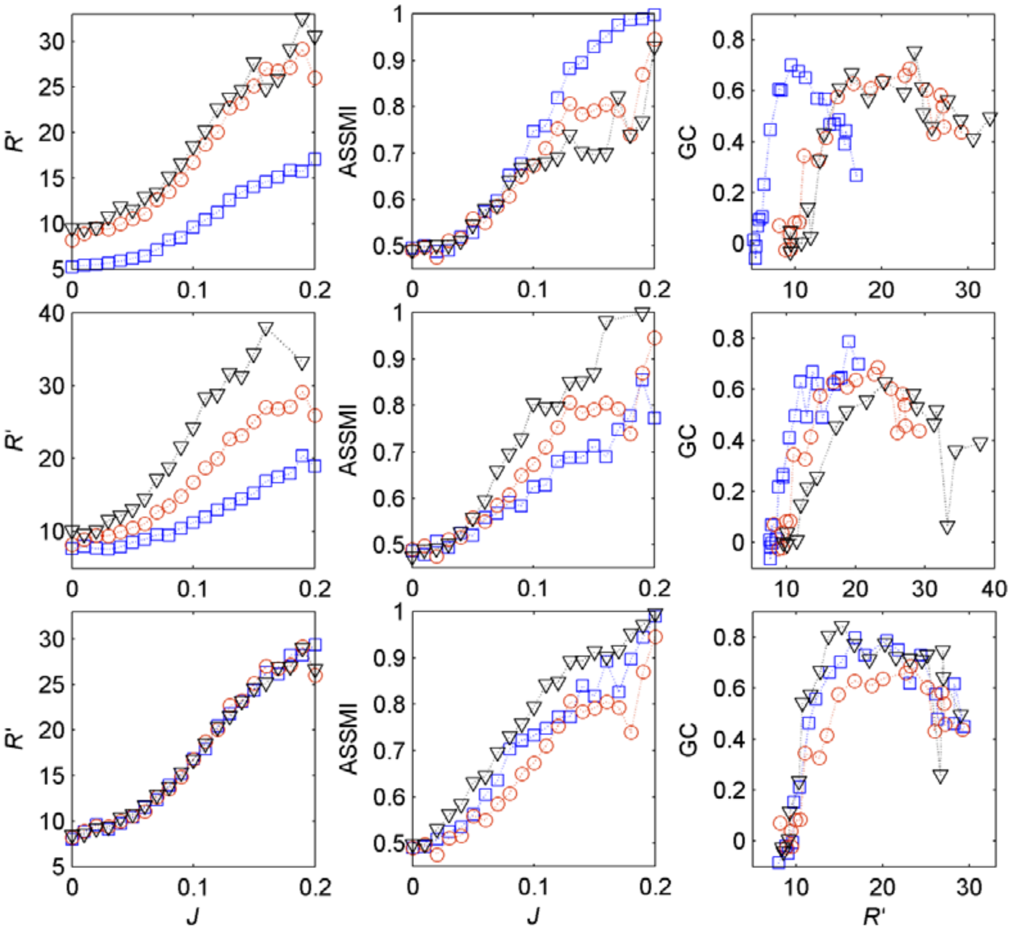

Figure 8.8. Effect of load distribution type (top row), node neighbourhood size (middle row), and data set size (bottom row) on topology identification. R' (left column) and ASSMI (centre column) are shown as functions of J, and GC (right column) as a function of R'. Top row: exponential (squares), uniform (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). Calculation: measures are medians over the respective values with the three ensembles.

L, because large data sets are more informative about node dependencies. As node dependencies are better estimated with larger data sets, also the graph correlation increases; the change in GC from $L = 270$ to $L = 540$ is particularly large, whereas the difference between $L = 540$ and $L = 1080$ is small. Consequently, the smallest data set seems still too small to estimate SSMI values accurately and thereby to obtain an accurate topology estimate, whereas the second largest data size gives already results similar to the largest one and is thus large enough.

Testing the effect of network size is more complicated, because increasing the number of nodes in a network should be coupled with a simultaneous increase in data set size. The following network sizes are studied here: $M = 30$ (reference), $M = 60$, and $M = 120$. Tests are run in three ways: L is first kept constant for each M, then increased linearly, and finally quadratically in M. The rationale for the last case is that the number of node pairs in a network grows quadratically in M. For the same quality data for each network size, steps in the MCMC burn-in period must be increased linearly in M as $500xM$. The neighborhood size is constant at $A = 8.8$ for all M.

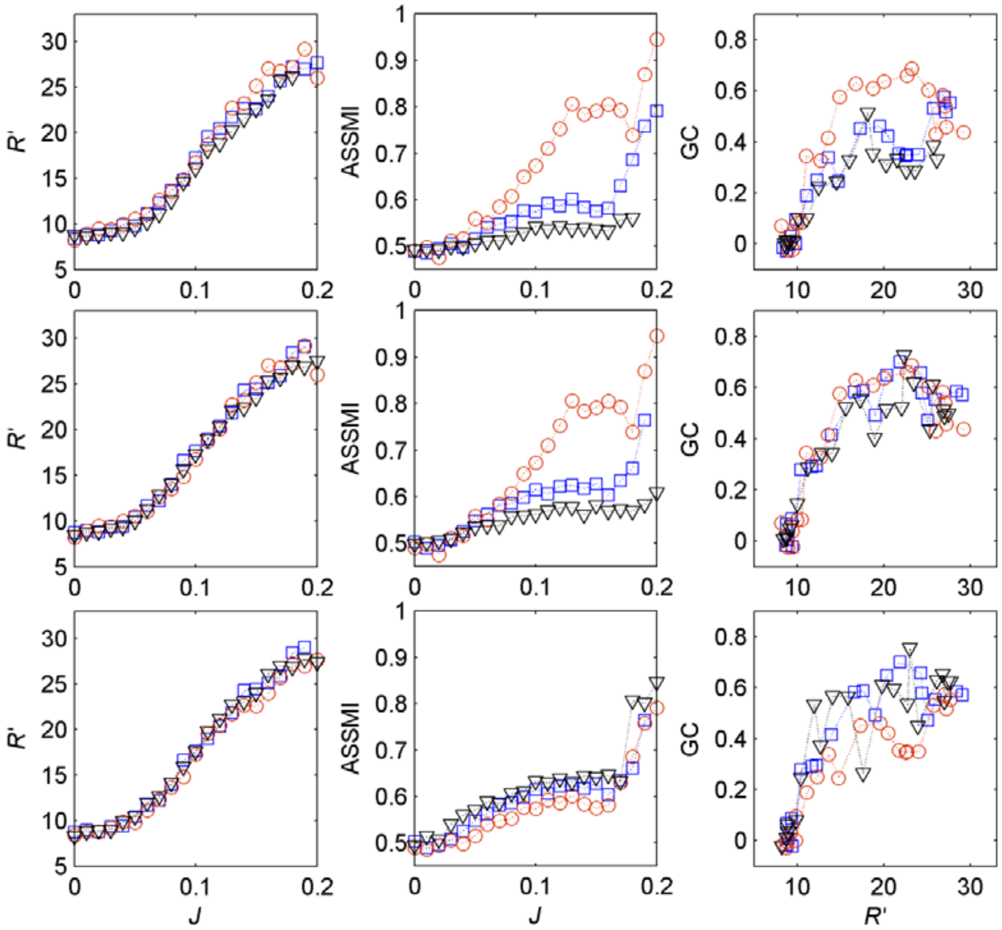

Figure 8.9. Effect of network size on topology identification. R' (left column) and ASSMI (centre column) are shown as functions of J, and GC (right column) as a function of R'. Top row: $M = 30$ (circles), $M = 60$ (squares), and $M = 120$ (triangles), each with $L = 270$. Middle row: $M = 30$, $L = 270$ (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 $M = 60$. Calculation: for $M = 30$, measures are medians over the respective values with the three ensembles. For the other network sizes, measures are calculated from single ensembles.

Figure 8.9 shows the results for constant data size, $L = 270$. With the smallest M, both the ASSMI and GC assume clearly larger values than with the two larger M values. Consequently, with the two larger networks, L is all too small for the data to uncover node dependencies and to obtain adequate topology estimates. R' remains nearly unchanged when M varies, which implies that the data assumes similar values with each network size. However, the ASSMI varies and is particularly different with $M = 30$, assuming clearly greater values than with the two larger networks. The conclusion is that L must be increased to gain reasonable topology estimates also with large networks.

Next, L is increased linearly in M (results in Figure 8.9). Because the computation time increases greatly with M, as the number of node pairs grows quadratically in M, only single ensembles are studied with $M = 60$ and $M = 120$. Consequently, the results may contain some extra uncertainty. Though coherence measures behave like in the previous case, GC values are similar for all the network sizes, suggesting that linear increase indeed is suitable. Even more computation time is required to test the quadratic increase of L in M. Therefore, again only a single ensemble is

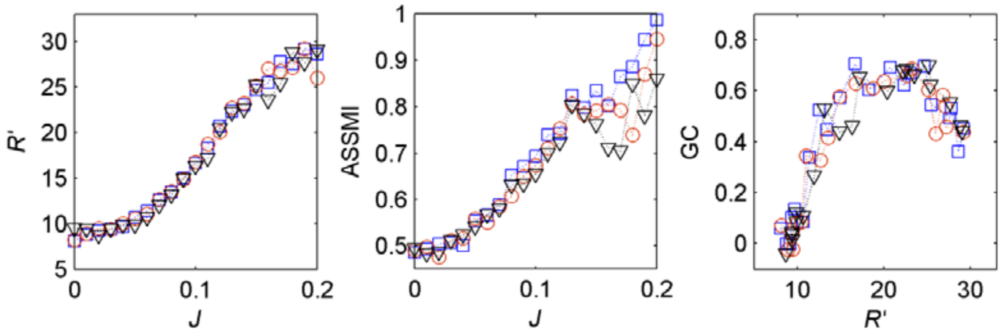

Figure 8.10. Effect of the number of MCMC burn-in steps on topology identification. R' (left) and ASSMI (centre) are shown as functions of J, and GC (right) as a function of R'. Results are shown with the following number of MCMC burn-in steps: $250x30$ (squares), $500x30$ (circles), and $1000x30$ (triangles). Calculation: measures are medians over the respective values with the three ensembles.

studied, and only the network with $M = 60$ is tested with $L = 270$, $L = 540$, and $L = 1080$ (results in Figure 8.9). The results for $L = 270$ are the worst, whereas despite rather heavy fluctuations, GC assumes values similar to those of the two larger data sets and almost those of the reference case. ASSMI values are similar to those of the two larger data sizes, whereas with $L = 270$ they are somewhat smaller. R' again remains nearly constant, indicating that the data is similar, as it should be, because in each case it is generated with the same Ising model parameters and with the same graph structure.

In conclusion, the question remains why the ASSMI seems to depend on network size, or at least differs greatly with $M = 30$ when compared to the two larger networks. The reason may simply be the particular properties of that randomly generated network, and because the network is so small, even a few highly connected nodes may drastically affect its properties, contributing to high coherence and a high ASSMI.

Finally, the quality of the data used in the above analyses is checked by changing the number of steps in the burn-in period of the MCMC data generation procedure. A well chosen burn-in period is especially important here, because the ensemble scheme is used to generate data observations. If too short a burn-in period is chosen, the Markov chain of the MCMC do not converge to the stationary distribution of the respective Ising model, and the samples are false. If too long a burn-in period is chosen, only some unnecessary time is wasted on sample generation, but the quality of the generated data set remains unaffected. With the 30-node network the number of MCMC steps is varied here from the reference case's $500x30$ to $250x30$ and to $1000x30$. As Figure 8.10 shows, all three cases yield similar results, confirming that the length of the burn-in period in the reference case is long enough to generate reasonably good network observations.

8.6 Comparison to Other Methods

In this section, the MGMN method is first compared to the straightforward topology identification approach discussed in Subsection 8.3.1 and then to one of the constrained-based graph learning methods discussed in Subsection 8.3.2, the GSMN algorithm. For a complete view of the performance of graph estimation methods, the methods should be tested with data generated

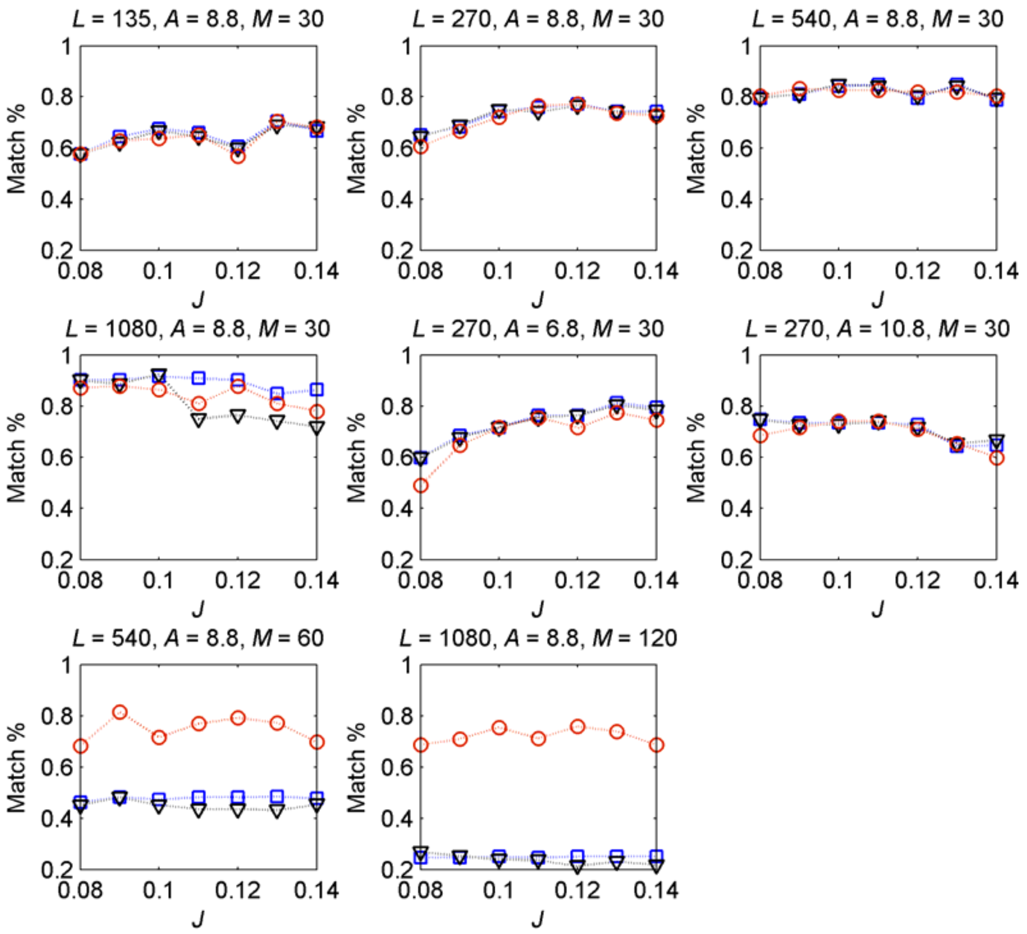

Figure 8.11. Comparison of the MGMN method (circles) to direct thresholding of MI (squares) and SSMI values (triangles). Performance of the methods is tested with several data set sizes (L), neighbourhood sizes (A), and network sizes (M). In each case, results are given in percentages of estimated graph 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. Calculation: measures are medians over the respective values with the three ensembles, except for the two largest networks where only single ensembles are used.

from various type graphs. For example, data could originate from a thresholded spatial network, which was already comprehensively discussed in the context of the MGMN method, but also from a graph where the links are not derived from a low-dimensional spatial configuration. The latter type graphs could be, e.g., random graphs and scale-free networks. However, because the methods are here applied to MTNs, where graphs are based on two-dimensional spatial configurations, the methods are compared only with graphs derived from the randomly generated 30-node spatial configuration shown in Figure 8.2.

Above, the NDC and GC measures have been applied to compare similarities of spatial configurations and graph structures. However, here another measure, the percentage of properly recovered links in an estimated graph, abbreviated to PRL measure, is introduced to fully compare graph structures. Because independent of A, when A is the same for the compared graphs, it is suitable to be used here. The PRL is a more exact measure than the GC to compare graphs in that the latter may yield relatively large values even if graph distances, but not necessarily exact neighbourhoods, are similar. The GC is thus well suited for measuring the overall fitness of an estimated graph. However, the PRL is used here for a more exact, or strict, measure of graph similarity. In some cases, it seems to categorise more clearly than the GC the performance of graph estimation methods.

In Figure 8.11, the MGMN method is compared to direct thresholding of MI and SSMI values in several cases of varying data size (L), neighbourhood size (A), and network size (M). When M increases here, L increases linearly in M, as in Subsection 8.5.3. Overall, the measures give similar results for $M = 30$, but with larger networks, i.e., $M = 60$ and $M = 120$, the MGMN method clearly outperforms the other two methods. Hence direct thresholding of MI/SSMI values seems to give good results with small size networks, though their performance drops drastically as M increases. Evidently, the difference in performance stems from conditional dependency relations, which only the MGMN method takes into account. Somewhat surprisingly, the thresholding of MI values gives results quite similar to the thresholding of SSMI values, because whereas the latter are absolute values and hence instantly comparable among different node pairs, the former are not, because they depend on node probability distributions, and hence on entropies. However, in this particular case with binary-state nodes, node probability distributions are apparently so similar that the MI values between different node pairs are nearly comparable and thus translate into good results.

As discussed in Section 8.3, the parameter α determines the neighbourhood size of a graph in the GSMN algorithm, and the GSMN algorithm, like many other constrained-based algorithms, estimates dense graphs rather poorly. The reference case with $A = 8.8$ (mimicking a true MTN network) is already a considerably dense graph and thus causes difficulties for the GSMN algorithm. Because of the above, the GSMN algorithm is studied in the following with various α values and with two sparser networks with $A = 2.8$ and $A = 5.8$, both obtained by thresholding the same spatial node configuration as in the reference case. Here $L = 540$ is used in most studies, because it gave clearly better results than $L = 270$, used previously in the reference case.

Figure 8.12 shows the results with the GSMN algorithm 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. Because the computation time increases with α, large α intervals are larger. The PRL measure can be applied in two ways: as done before, to take the set of true neighbours and to calculate the proportion of links the estimated graph covers (centre column in Figure 8.12), or to do the above reversely, i.e., to take the links of the estimated graph and to calculate the proportion of links covered by the true graph (right column in Figure 8.12). When A is the same for both the true and estimated graph, as it is in Figure 8.11, the two measures are the same. However, as in Figure 8.12, where graph estimation is tested by varying α, and hence the average estimated number of neighbours A', the two measures are obviously different. The latter way of calculating the PRL measure favours the GSMN algorithm, because when $A' < A$, obviously chances are better for a matching true neighbour relation for each estimated one. For comparison, results with the MGMN method are also shown in Figure 8.12 (straight lines) when $A' = A$.

Estimation of the denser graphs (large α values) with the GSMN method posed some difficulties with the GSMN method, which is why Figure 8.12 shows several missing data points. None-

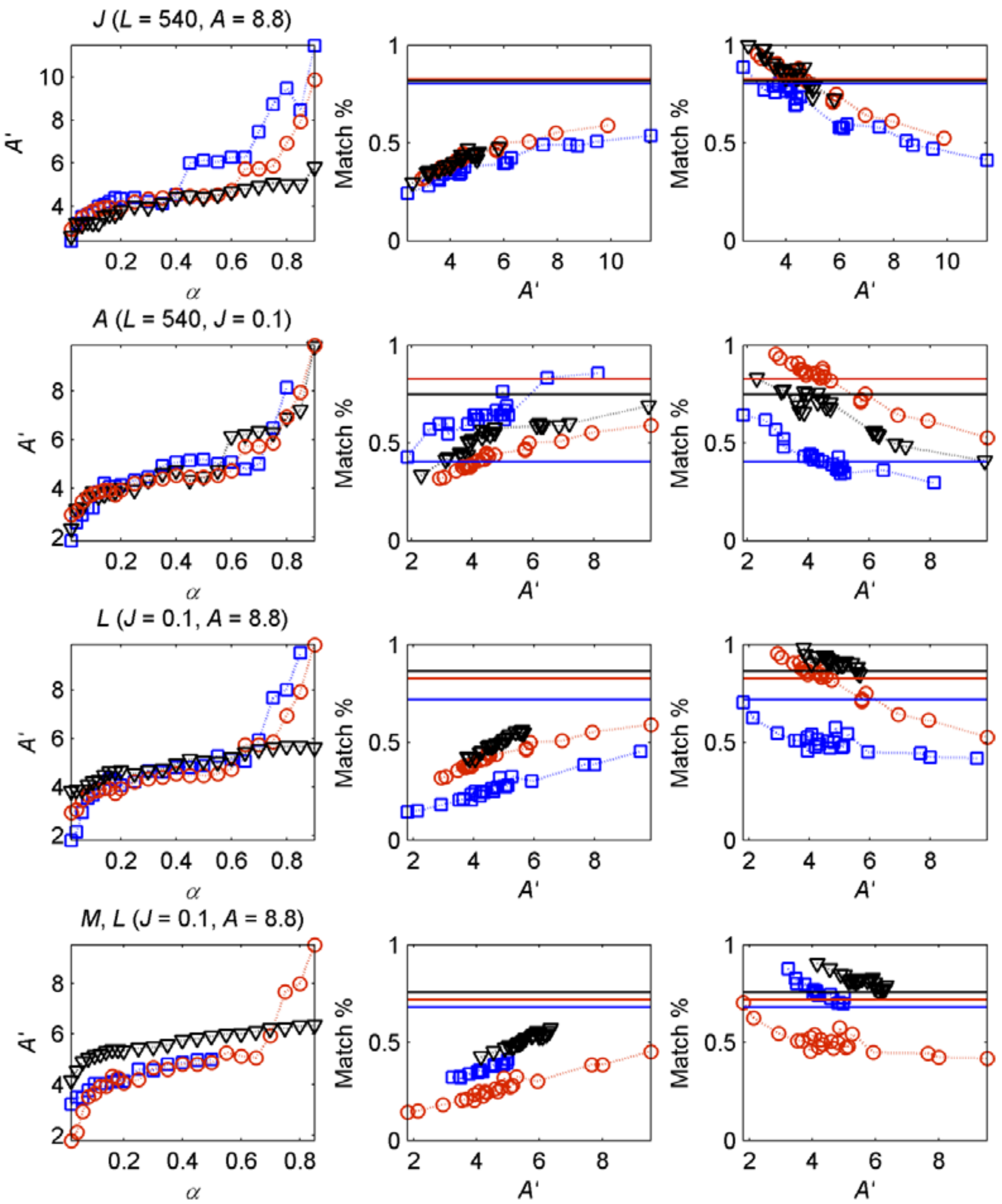

Figure 8.12. Performance of the GSMN algorithm. The figure shows A' as a function of α (left column), the proportion of correctly estimated links among all true links (centre column), and the proportion of true links matching estimated links among all estimated links (right column). From the top to bottom 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 even intervals of 0.02, and from 0.25 to 0.9 at even intervals of 0.05. The two rightmost columns also show result with the MGMN method, when $A' = A$, as vertical lines, with the colours matching the respective cases. In the first three rows, a reference case (circles) is used ($J = 0.1$, $A = 8.8$, $L = 540$, $M = 30$). Calculation: with the GSMN algorithm, results are obtained from a single ensemble, whereas with the MGMN method, results are medians over the three ensembles; however, when $M = 60$ and $M = 120$, results are from a single ensemble.

theless, first, graph estimation is studied with three J values (0.08, 0.1, 0.12) in the top row of Figure 8.12. Results are clearly worse with $J = 0.08$ than in the two cases of greater coherence. When A' is small, nearly all estimated links are correct, but as more links are estimated, the rate of correctness drops, and at about $A' = 8.8$, results are clearly worse than with the MGMN method.

In the second case of varying A (2.8, 8.8, 5.8), the two ways of calculating the PRL measure seem to yield somewhat inconsistent information in that according to the first way, the graphs are better when $A = 2.8$, whereas according to the second, the graphs are better when $A = 8.8$. This is because with most α values $A' > 2.8$ when $A = 2.8$, whereas with most α values $A' < 8.8$ when $A = 8.8$. In general, $A' > A$ translates into a smaller probability of a true link corresponding to an estimated link than an estimated link corresponding to a true link. Hence the two ways of calculating the PRL behave oppositely, and when A is changed, one way may give better results with a large A, whereas the other produces exactly opposite results. Upon further scrutiny of the A' values with each A in Figure 8.12, where $A' \approx A$, both PRL calculations give similar but poorer results than those by the MGMN method.

In the third case of varying L (270, 540, 1080), the two largest data sets give clearly the best results. With $L = 1080$, graph estimation is successful only in a narrow range of A' values. With each L, graph estimates at $A' \approx A$ are again clearly worse than those by the MGMN method. Finally, in the last case, M varies and L increases linearly in M, yielding cases (M, L): (60, 540; 30, 270; 120, 1080). Graph estimation with the two larger M is again successful only in a narrow range of A' values. Somewhat surprisingly, graph estimates are the better, the larger the network. This holds also for the results of the MGMN method, though differences there are small and based only on single data sets. Again, results obtained with the MGMN method seem at least as good as those by the GSMN method, even if the latter had yielded results also at $A' \approx A$. In conclusion, at least where relevant to the application studied here—a graph based on a two-dimensional spatial configuration—the MGMN method seem better suited for estimating graphs than the GSMN algorithm.

9. Parameter Identification

Identifying the MRF graph structure makes it possible to identify MRF model parameters. The MRF parameter estimation is undertaken in this chapter via the maximum pseudolikelihood (MPL) method [17]. MRF model parameters can be estimated also with some other methods, e.g., the coding method [16], usually considered less efficient than the MPL method and thus not widely applied [162]. Another method estimates parameters by maximum likelihood and evaluates the intractable partition function via MCMC simulations. This method, of course, comes with the obvious disadvantage of excessive computation time. These three methods have been compared in [162] with the MPL method deemed the most practical among them.

Traditionally, parameter estimation has focused on seeking the best point estimate values. However, when parameters are estimated with a finite data set and with all the randomness associated with the data, uncertainties related to parameter estimates must also be considered. Uncertainty information is naturally included in the Bayesian approach through probability distributions. This approach extends also to cases where prior information is not available. Uncertainty information can then be used with only the likelihood part and thus to search for MPL parameter estimates. In general, the form of the joint probability distribution of an MRF model is unknown, which makes it difficult to assess the uncertainty of estimated parameters. However, if the true distribution can be approximated with the Gaussian distribution, the covariance matrix provides straightforward uncertainty measures of parameter estimates.

In this chapter the MRF model parameter estimation with the methods introduced in Chapter 6 will be studied. Section 9.1 deals with the MPL method in a general MRF model case and derives MPL estimates for the Ising model. Section 9.2 discusses parameter uncertainties through the covariance matrix of the Gaussian distribution approximation with uncertainties derived for Ising model parameter estimates. Section 9.3 introduces methods to evaluate the estimated parameters. Section 9.4 evaluates the performance of MPL parameter estimation with all synthetic data cases and by using the graph structures identified in Chapter 8. Finally, it will be emphasised that quality of model parameter estimates provides information about the entire MRF model identification scheme, hence also about the topology identification phase.

9.1 MPL Estimates of MRF Model Parameters

The maximum pseudolikelihood estimation method was presented in its general form for MRF models in Chapter 6. When further exploiting the conditional independence properties of the MRF models according to Eq. (4.2), maximum pseudolikelihood parameter estimates $\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}}$ can be obtained as

$$\begin{aligned}\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}} &= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \prod_{l=1}^{L} \prod_{m=1}^{M} p(s_m^{(l)} | \{s_n^{(l)}\}_{n \in N(m)}, \boldsymbol{\theta}) \\ &= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{l=1}^{L} \sum_{m=1}^{M} \log p(s_m^{(l)} | \{s_n^{(l)}\}_{n \in N(m)}, \boldsymbol{\theta}),\end{aligned} \tag{9.1}$$

where the full conditional of node m depends only on its neighbours, $p(s_m | \mathbf{s}_{-m}) = p(s_m | \{s_n\}_{n \in N(m)})$.

Let us now try to find MPL estimates for the general MRF model, given by Eq. (4.3). First, by using the conditional independence properties, we can write Eq. (4.3) as

$$p(\mathbf{s}) = Z^{-1}\prod\nolimits_{m=1}^{M}\prod\nolimits_{n\in N(m)}\psi_V(s_m, s_n)\prod\nolimits_{m=1}^{M}\psi(s_m). \tag{9.2}$$

The full conditional of node m is now obtained from Eq. (9.2) by fixing the states of its neighbours. Because constants, the product terms in Eq. (9.2) that do not include node m can be moved to the normalisation term. The conditional probability of Eq. (4.2) for node m with the modified normalisation term Z_m is now

$$p(s_m|\mathbf{s}_{-m}) = p(s_m|\{s_n\}_{n\in N(m)}) = Z_m^{-1}\psi(s_m)\prod\nolimits_{n\in N(m)}\psi_V(s_m, s_n). \tag{9.3}$$

Indexing the observations with superscript l, we can now estimate parameters for the general MRF model by the MPL method as

$$\begin{aligned}\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}} &= \operatorname*{argmax}_{\boldsymbol{\theta}}\prod\nolimits_{l=1}^{L}\prod\nolimits_{m=1}^{M} p(s_m^{(l)}|\{s_n^{(l)}\}_{n\in N(m)}, \boldsymbol{\theta})\\ &= \operatorname*{argmax}_{\boldsymbol{\theta}}\prod\nolimits_{l=1}^{L}\prod\nolimits_{m=1}^{M}\left[Z_m^{(l)^{-1}}\psi(\mathrm{s}_m^{(l)}|\boldsymbol{\theta})\prod\nolimits_{n\in N(m)}\psi_V(\mathrm{s}_m^{(l)}, \mathrm{s}_n^{(l)}|\boldsymbol{\theta})\right].\end{aligned} \tag{9.4}$$

Seeking for the position of maximum of the objective function's logarithm rather than the function itself, we get

$$\begin{aligned}\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}} &= \operatorname*{argmax}_{\boldsymbol{\theta}}\sum_{l=1}^{L}\sum_{m=1}^{M}\log\left[Z_m^{(l)^{-1}}\psi(\mathrm{s}_m^{(l)}|\boldsymbol{\theta})\prod_{n\in N(m)}\psi_V(\mathrm{s}_m^{(l)}, \mathrm{s}_n^{(l)}|\boldsymbol{\theta})\right]\\ &= \operatorname*{argmax}_{\boldsymbol{\theta}}\sum_{l=1}^{L}\sum_{m=1}^{M}\left\{\log[\psi(\mathrm{s}_m^{(l)}|\boldsymbol{\theta})] + \sum_{n\in N(m)}\log[\psi_V(\mathrm{s}_m^{(l)}, \mathrm{s}_n^{(l)}|\boldsymbol{\theta})]\right.\\ &\qquad\left. -\log(Z_m^{(l)})\right\}.\end{aligned} \tag{9.5}$$

Let us next consider MPL estimation of Ising model parameters. The full conditional of node m was previously given by Eq. (4.13) and at observation l it can be rewritten as $p(s_m^{(l)}|\{\mathrm{s}_n^{(l)}\}_{n\in N(m)}) = Z_m^{(l)^{-1}}\exp(s_m^{(l)}h_m^{\mathrm{eff},(l)})$. Eq. (4.14) showed that the conditional partition function $Z_m^{(l)}$ for node m at observation l is $Z_m^{(l)} = \exp(-h_m^{\mathrm{eff},(l)}) + \exp\,(h_m^{\mathrm{eff},(l)}) = 2\cosh(h_m^{\mathrm{eff},(l)})$, where $h_m^{\mathrm{eff},(l)} = \sum_{n\in N(m)} s_n^{(l)} + H(h_m^{(l)} - h_0)$ was defined by Eq. (4.5). For the Ising model, the other two terms in Eq. (9.5) are now as follows: $\log[\psi(\mathrm{s}_m^{(l)}|\boldsymbol{\theta})] = s_m^{(l)}H(h_m^{(l)} - h_0)$, and $\sum_{n\in N(m)}\log[\psi_V(\mathrm{s}_m^{(l)}, \mathrm{s}_n^{(l)}|\boldsymbol{\theta})] = s_m^{(l)}\sum_{n\in N(m)} s_n^{(l)}$. Consequently, we can write the MPL estimation of the Ising model parameters $\boldsymbol{\theta} = (J, H, h_0)$ as

$$\begin{aligned}\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}} &= \operatorname*{argmax}_{\boldsymbol{\theta}}\sum_{l=1}^{L}\sum_{m=1}^{M}\left\{s_m^{(l)}H(h_m^{(l)} - h_0) + s_m^{(l)}\sum_{n\in N(m)} s_n^{(l)} - \log(Z_m^{(l)})\right\}\\ &= \operatorname*{argmax}_{\boldsymbol{\theta}}\sum_{l=1}^{L}\sum_{m=1}^{M}\left\{s_m^{(l)}h_m^{\mathrm{eff},(l)} - \log\,[2\cosh(h_m^{\mathrm{eff},(l)})]\right\},\end{aligned} \tag{9.6}$$

from which MPL parameter estimates are easy and efficient to solve numerically.

9.2 Uncertainties of MRF Model Parameter Estimates

The MPL method yields only optimal parameter values but ignores the uncertainties related to these parameter estimates. In pseudolikelihood approximation, parameter uncertainties can be assessed by the Bayesian approach, discussed in Chapter 6. Because of no prior information about MRF model parameters, the prior probability distribution in Bayes's theorem can be set as a constant and thus, according to Eq. (6.7), with pseudolikelihood approximation, the posterior distribution approximately equals the product of the constant multiplying the pseudolikelihood term: $p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L}) \approx N_{C3}\prod_{l=1}^{L}\prod_{m=1}^{M}p(s_m^{(l)}|\mathbf{s}_{-m}^{(l)},\boldsymbol{\theta})$. When conditional independence properties are further used, as in Eq. (9.5), the pseudolikelihood posterior parameter distribution becomes now

$$p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L}) \approx N_{C3}\prod_{l=1}^{L}\prod_{m=1}^{M}p(s_m^{(l)}|\{s_n^{(l)}\}_{n\in N(m)},\boldsymbol{\theta}). \tag{9.7}$$

This is the general form of the parameter probability distribution of an MRF model, when prior information is assumed absent and the pseudolikelihood approximation is given by Eq. (6.6). The distribution reveals all the available information about the uncertainties of the MPL parameter estimates. Because MPL parameter estimates $\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}}$ are obtained by maximising Eq. (9.7), the mode of the posterior distribution is obviously $\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}}$. In the general MRF model, the posterior distribution $p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L})$ is not known in its analytical form, but in some special cases it is. For example, in a GMRF model, all conditional distributions are Gaussians, thus also the posterior distribution. In the general case, uncertainty is not as easily assessed as in the Gaussian case.

However, the general posterior distribution can be approximated with a Gaussian distribution, and if the approximation is appropriate, the uncertainty of the Gaussian distribution, i.e., that given by the covariance matrix, can be considered a parameter uncertainty of the posterior distribution. To obtain a Gaussian approximation, let us first denote the gradient (first-order derivative) and Hessian (second-order derivative) operators with respect to parameters $\boldsymbol{\theta}$ by ∇ and ∇^2, and make a second-order Taylor series approximation to the logarithm of the posterior distribution $p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L})$ at around $\boldsymbol{\theta} = \widehat{\boldsymbol{\theta}}_{\mathrm{MPL}}$ as

$$\begin{aligned}&\log[p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L})] \approx \log[p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L})]\big|_{\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}}}\\&+\nabla\log[p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L})]\big|_{\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}}}(\boldsymbol{\theta}-\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}})\\&+\tfrac{1}{2}(\boldsymbol{\theta}-\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}})^{\mathrm{T}}\nabla^2\log[p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L})]\big|_{\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}}}(\boldsymbol{\theta}-\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}}).\end{aligned} \tag{9.8}$$

The gradient term goes to zero, because the estimate maximises the posterior distribution. By taking the exponent from both sides and denoting $D = p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L})|_{\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}}}$ and $\mathbf{C}^{-1} = -\nabla^2\log[p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L})]\,|_{\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}}}$ (inverse covariance matrix), Eq. (9.8) is simplified to

$$p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L}) \approx D\exp[-\tfrac{1}{2}(\boldsymbol{\theta}-\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}})^{\mathrm{T}}\mathbf{C}^{-1}(\boldsymbol{\theta}-\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}})]. \tag{9.9}$$

This is the Gaussian approximation of the true parameter posterior distribution $p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L})$ at $\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}}$ with a covariance matrix $\mathbf{C}$, or precision matrix $\mathbf{C}^{-1}$. If the Gaussian approximation is close enough to the true posterior distribution, uncertainties in the estimated model parameters can now be studied through the covariance matrix $\mathbf{C}$, where the diagonal elements give the variances of the respective parameters and the off-diagonal elements inform us about the correlations between the parameter uncertainties.

By denoting the log-pseudolikelihood by $L_{p,\log} = \log[p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L})]$, the inverse covariance matrix $\mathbf{C}^{-1}$ is obtained for the Ising model parameters $\boldsymbol{\theta} = (J, H, h_0)$ as

$$\begin{aligned}\mathbf{C}^{-1} &= -\nabla^2 L_{p,\log}\big|_{\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}}} \\ &= -\begin{bmatrix} \frac{\partial^2}{\partial J^2} L_{p,\log} & \frac{\partial^2}{\partial J \partial H} L_{p,\log} & \frac{\partial^2}{\partial J \partial h_0} L_{p,\log} \\ \frac{\partial^2}{\partial H \partial J} L_{p,\log} & \frac{\partial^2}{\partial H^2} L_{p,\log} & \frac{\partial^2}{\partial H \partial h_0} L_{p,\log} \\ \frac{\partial^2}{\partial h_0 \partial J} L_{p,\log} & \frac{\partial^2}{\partial h_0 \partial H} L_{p,\log} & \frac{\partial^2}{\partial {h_0}^2} L_{p,\log} \end{bmatrix}_{\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}_{\mathrm{MPL}}},\end{aligned} \tag{9.10}$$

whose matrix elements are specified in Table 9.1.

To verify the fitness of the Gaussian approximation, the approximate Gaussian distribution must be compared to the respective true posterior distribution, which can be done as follows. First, the true posterior distribution $p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^{L})$ is evaluated in a region of parameter space where probability values are (clearly) non-zero. Approximately this region can be found as an ellipsoid of the Gaussian contour of the constant probability p_0 of the approximative Gaussian distribution. Parameter values can then be sampled from the area of the ellipsoid and the true posterior distribution evaluated at these points. The true posterior distribution and the Gaussian approxi-

Table 9.1. Elements of $\mathbf{C}^{-1}$ defined in Eq. (9.10) for Ising model parameters $\boldsymbol{\theta} = (J, H, h_0)$.

$-\frac{\partial^2}{\partial J^2} L_{p,\log} = \sum_{l=1}^{L} \sum_{m=1}^{M} \left[\sum_{n \in N(m)} \mathrm{s}_n^{(l)} \right]^2 \cosh^{-2}(h_m^{\mathrm{eff},(l)})$
$-\frac{\partial^2}{\partial H^2} L_{p,\log} = \sum_{l=1}^{L} \sum_{m=1}^{M} (h_m^{(l)} - h_0)^2 \cosh^{-2}(h_m^{\mathrm{eff},(l)})$
$-\frac{\partial^2}{\partial {h_0}^2} L_{p,\log} = H^2 \sum_{l=1}^{L} \sum_{m=1}^{M} \cosh^{-2}(h_m^{\mathrm{eff},(l)})$
$-\frac{\partial^2}{\partial J \partial H} L_{p,\log} = -\frac{\partial^2}{\partial H \partial J} L_{p,\log} = \sum_{l=1}^{L} \sum_{m=1}^{M} (h_m^{(l)} - h_0) \cosh^{-2}(h_m^{\mathrm{eff},(l)}) \left[\sum_{n \in N(m)} \mathrm{s}_n^{(l)} \right]$
$-\frac{\partial^2}{\partial J \partial h_0} L_{p,\log} = -\frac{\partial^2}{\partial h_0 \partial J} L_{p,\log} = -H \sum_{l=1}^{L} \sum_{m=1}^{M} \cosh^{-2}(h_m^{\mathrm{eff},(l)}) \left[\sum_{n \in N(m)} \mathrm{s}_n^{(l)} \right]$
$-\frac{\partial^2}{\partial H \partial h_0} L_{p,\log} = -\frac{\partial^2}{\partial h_0 \partial H} L_{p,\log} = \sum_{l=1}^{L} \sum_{m=1}^{M} s_m^{(l)} - \tanh(h_m^{\mathrm{eff},(l)}) - H \cosh^{-2}(h_m^{\mathrm{eff},(l)})(h_m^{(l)} - h_0)$

mation can now be compared by evaluating the marginal distribution for each parameter for both true and Gaussian approximation of the posterior distribution, and by then comparing the respective marginal distributions obtained. If the marginal distributions are similar, the approximation is sound and can be used to study parameter uncertainties.

Let us further examine the practical calculation of true marginal probability distributions in the Ising model, and assume that a set of parameter values, $\{J^r, H^r, h_0^r\}_{r=1}^R$, is uniformly and randomly drawn inside the contour of the constant probability p_0 with the superscript r indexing the sample values $r = 1, \dots, R$. Given an observed data set, $\{s_m^{(l)}, h_m^{(l)}\}_{m=1}^M$, $l = 1, \dots, L$, the marginal distribution of each parameter is obtained similarly. Here J is considered an example for which the marginal distribution is derived by marginalising the joint distribution as

$$p(J|\{\mathbf{s}^{(l)}\}_{l=1}^L) = p(J) = \sum_{\substack{H \in D(H),\\ h_0 \in D(h_0)}} p(\boldsymbol{\theta}|\{\mathbf{s}^{(l)}\}_{l=1}^L) = \sum_{\substack{H \in D(H),\\ h_0 \in D(h_0)}} p(J, H, h_0), \tag{9.11}$$

where D denotes the domains of parameters H and h_0. However, because the data set of parameter values, $\{J^r, H^r, h_0^r\}_{r=1}^R$, is sampled, the true marginal distribution is obtained in practice by first dividing the sample values of J, $\{J^r\}_{r=1}^R$, into a set of constant-size intervals, say $\{\Delta J^g\}_{g=1}^G$, with $g = 1, \dots, G$ indexing the intervals. Then Eq. (9.11) is applied to each interval separately, and at a certain interval ΔJ^g, the marginal probability of parameter J is obtained by summing over all probability values $p(J^r)$ of those sample values J^r that appear at that interval, and then normalising the values to probabilities:

$$p_{\Delta J^g}(J) = \frac{\sum_{J^r \in \Delta J^g} p(J^r)}{\sum_{g=1}^G \sum_{J^r \in \Delta J^g} p(J^r) d(\Delta J^g)}. \tag{9.12}$$

Here $d(\Delta J^g)$ is the width of the interval ΔJ^g, and considered a constant. The denominator in Eq. (9.12) normalises the distribution to probabilities. Hence values $\{p_{\Delta J^g}(J)\}_{g=1}^G$ form a discrete probability distribution estimate of the true marginal probabilities of J inside the contour of the constant probability p_0.

According to the definition of pseudolikelihood in Eq. (6.7), the joint probability distribution estimate of the parameters can be written by using the pseudolikelihood, L_p, as $p(J, H, h_0) \approx N_{C3} L_p$. Writing Eq. (9.12) by using this formula and the marginalisation in Eq. (9.11) yields

$$\begin{aligned} p_{\Delta J^g}(J) &= \frac{\sum_{J^r \in \Delta J^g} \sum_{H \in D(H), h_0 \in D(h_0)} N_{C3} L_p}{\sum_{r=1}^R \sum_{J^r \in \Delta J^g} \sum_{H \in D(H), h_0 \in D(h_0)} N_{C3} L_p \, d(\Delta J^g)} \\ &= \frac{\sum_{J^r \in \Delta J^g} \sum_{H \in D(H), h_0 \in D(h_0)} L_p}{\sum_{r=1}^R \sum_{J^r \in \Delta J^g} \sum_{H \in D(H), h_0 \in D(h_0)} L_p \, d(\Delta J^g)}, \end{aligned} \tag{9.13}$$

where the terms N_{C3} appearing both in the nominator and the denominator cancels. In practice, parameters are estimated from the log-pseudolikelihood $L_{p,\log} = \log[L_p]$, which is computationally easier to evaluate than evaluating the pseudolikelihood directly. Hence when the log-pseudolikelihood is used in Eq. (9.13), it becomes

$$p_{\Delta J^g}(J) = \frac{\sum_{J^r \in \Delta J^g} \sum_{H \in D(H), h_0 \in D(h_0)} \exp[L_{p,\log}]}{\sum_{r=1}^{R} \sum_{J^r \in \Delta J^g} \sum_{H \in D(H), h_0 \in D(h_0)} \exp[L_{p,\log}]\, d(\Delta J^g)}. \tag{9.14}$$

In some practical calculations, taking the exponential in Eq. (9.14) from $L_{p,\log}$ may not be successful, because its values may be very large. It may then help to add a constant to the log-pseudolikelihood values inside the exponents of both the nominator and denominator in Eq. (9.14). These values cancel out each other, and if the constant is chosen appropriately, the log-pseudolikelihood values can be reduced to enable practical calculation.

Eq. (9.14) can be directly used to estimate the true marginal probabilities of parameter J, when log-pseudolikelihoods have been calculated for sampled parameter values. Marginal probability is calculated similarly for the other two parameters. In the Gaussian approximation, a marginal probability distribution is obtained for each parameter by simply evaluating a univariate Gaussian distribution, with the expectation value determined by the respective maximum pseudolikelihood parameter value and variance by the respective diagonal element in the estimated covariance matrix $\mathbf{C}$.

9.3 Evaluation Methods of Parameter Estimation

When an estimated network topology is used for the graph structure of an MRF model, and when model parameters are further estimated, evaluation of the quality of the parameter estimates always includes evaluation of the quality of the estimated graph structure. An exception is the evaluation of model parameter identification with synthetic data with a known graph structure, where estimated parameters can simply be compared to true ones. But when the methods are applied to real data and when true parameters are unknown, also the graph structure is involved, and additional evaluation methods must be adopted. One way to evaluate the quality of estimated parameters, or the entire model, is to consider the predictions made with the identified model. However, because networked systems usually contain a large number of network nodes, compared to the amount of available data, all available data must be exploited in the modelling phase.

Because of a data shortage, a scheme similar to cross-validation is adopted here by considering single node state predictions based on full conditionals of nodes and on data already applied in the identification phase. For example, in the Ising model, for each node m with each observation l, probabilities of the two states of that node are predicted with the full conditional, yielding, e.g., for state -1 the probability $p(s_m^{(l)} = -1|\{s_n^{(l)}\}_{n \in N(m)})$. Then the average of these state probabilities is taken over all the observations, again giving for state -1 the average probability $\frac{1}{L}\sum_{l=1}^{L} p(s_m^{(l)} = -1|\{s_n^{(l)}\}_{n \in N(m)})$. Respective marginal state probabilities can be calculated from the state observation set of node m, which for state -1 yields $\frac{1}{L}\sum_{l=1}^{L} |s_m^{(l)}|\delta(s_m^{(l)} + 1)$, where δ denotes the Kronecker delta-function. Predicted probabilities for state -1 and their respective data-calculated probabilities can then be compared for node m; repeating the comparison for each node then yields an overall picture of the fitness of the identified model.

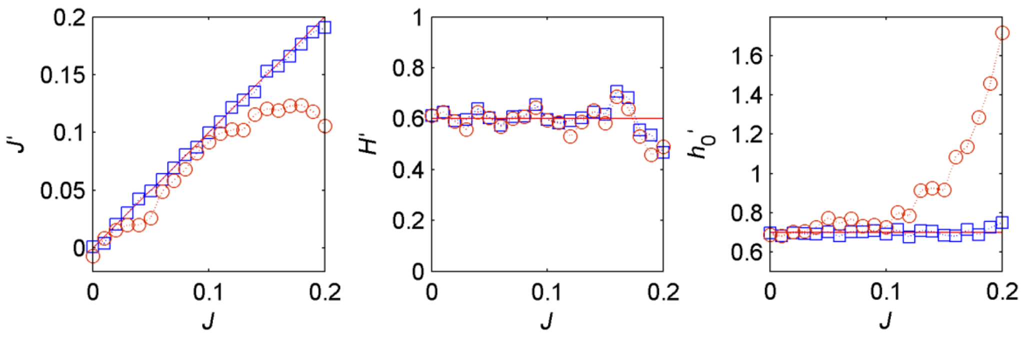

Figure 9.1. Ising model parameter estimates J' (left), H' (centre), and $h_0{}'$ (right) as functions of J. Results are shown with true (squares) and estimated topology (circles). Solid lines denote reference curves of optimal parameter estimates. Calculation: parameter values are medians over the respective values with three ensembles.

9.4 Results with MCMC-Generated Synthetic Data

In this section, MRF model parameter identification is examined with the Ising model by studying the various Ising model parameterisation cases introduced in Subsection 8.5.1 and used in Chapter 8 to evaluate the topology identification. Parameter identification and parameter uncertainties are first studied with these synthetic data cases in Subsection 9.4.1. Evaluation of the parameter identification method and entire MRF model identification are studied in Subsection 9.4.2 via model predictions. The effect of various data characteristics on parameter estimation is tested in Subsection 9.4.3 with the cases used to evaluate topology identification in Subsection 8.5.4. Again, most results are median values over the three parallel cases (see Subsection 8.5.1).

9.4.1 Parameter Estimates and Uncertainties

This subsection deals with MRF model parameter identification in the reference case introduced in Subsection 8.5.1 in the context of topology identification. In particular, in each reference case parameterisation, the graph structure identified in Subsection 8.5.1 is applied here for structure in the Ising model. However, the focus here is on identifying parameters, and using only identified graph structures would complicate separating between graph identification and parameter identification. Consequently, to gain information on parameter identification alone, all cases are considered also when the true synthetic graph structure is applied for structure in the Ising model. Ising model parameters are estimated with the pseudolikelihood method, and uncertainties of estimated parameters are studied via Gaussian approximations of true parameter distributions.

With the reference case of 30 nodes the estimated Ising model parameters (J', H', h_0'), with both estimated and true graph structures are presented in Figure 9.1 as functions of the true interaction parameter J, while the other two parameters are constants: $H = 0.6$ and $h_0 = 0.7$. According to Figure 9.1, J and h_0 are estimated almost perfectly with the true graph structure, whereas with the estimated graph structures, estimates appear fairly good up to about $J = 0.13$, beyond which the effect of the neighbourhood start to increase with an increasing interaction term magnitude. In addition, with large J values, graph estimates deteriorate, as shown, e.g., in Figure 8.5. Consequently, J' and h_0' are worse with large J values. Though graph estimates are the worst with small J values, this does not show in parameter estimates because of the very small magnitude of the interaction term, which makes the neighbourhood relations quite insignificant.

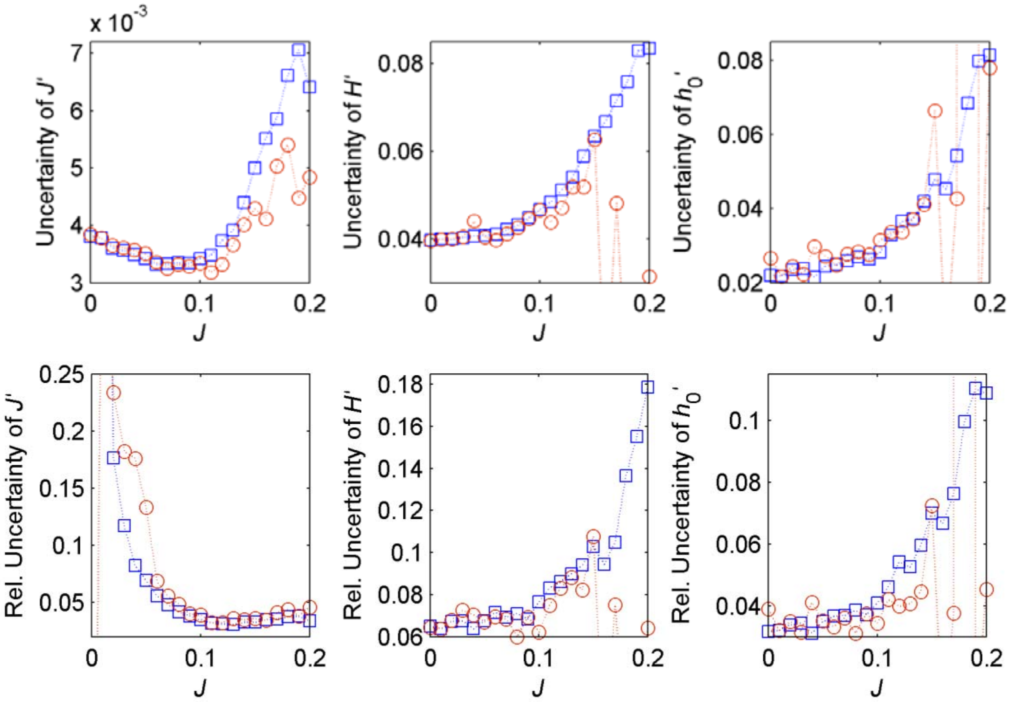

Figure 9.2. Uncertainties (top row) and relative uncertainties (bottom row) of estimated Ising model parameters. Uncertainties of J' (left column), H' (centre column), and $h_0{}'$ (right column) are shown as functions of J. Results are shown with true (squares) and estimated topology (circles). Calculation: uncertainty values are medians over the respective values with three ensembles.

Parameters J and h_0 have essentially the same effect on network coherence; i.e., keeping the first fixed and changing the second has the same effect as keeping the second fixed and changing the first, hence their similar, but opposite behaviour as a function of J in Figure 9.1. Using the estimated graph structure seems also to induce a bias error in both J' and $h_0{}'$ with J' being constantly smaller than J, and $h_0{}'$ being larger than h_0. Because an estimated graph structure is never as good as a true one, coherence may always appear less than what it truly is, and thus possibly cause J to be constantly underestimated and h_0 thereby to be overestimated. No bias error exists in H'; instead it fluctuates similarly between the true and the estimated graphs with the magnitude increasing at large J values.

Uncertainties related to estimated parameter values in cases of estimated and true graph structures are shown in Figure 9.2, which also shows relative parameter uncertainties, i.e., uncertainty values divided by their respective parameter estimate values. The uncertainty of each parameter is here the standard deviation of the marginal probability distribution of the parameter, calculated from the Gaussian joint distribution approximation of the true joint parameter distribution. Parameter uncertainties behave similarly in both graph structure cases, but after about $J = 0.13$, uncertainties based on estimated graph structures start to fluctuate widely. Relative uncertainties act somewhat like absolute values, except that at small J they are large for J'. Overall, in the reference case, parameters are estimated quite successfully, but as expected, because of imperfect graph estimates, respective parameter estimates also deviate somewhat from true ones.

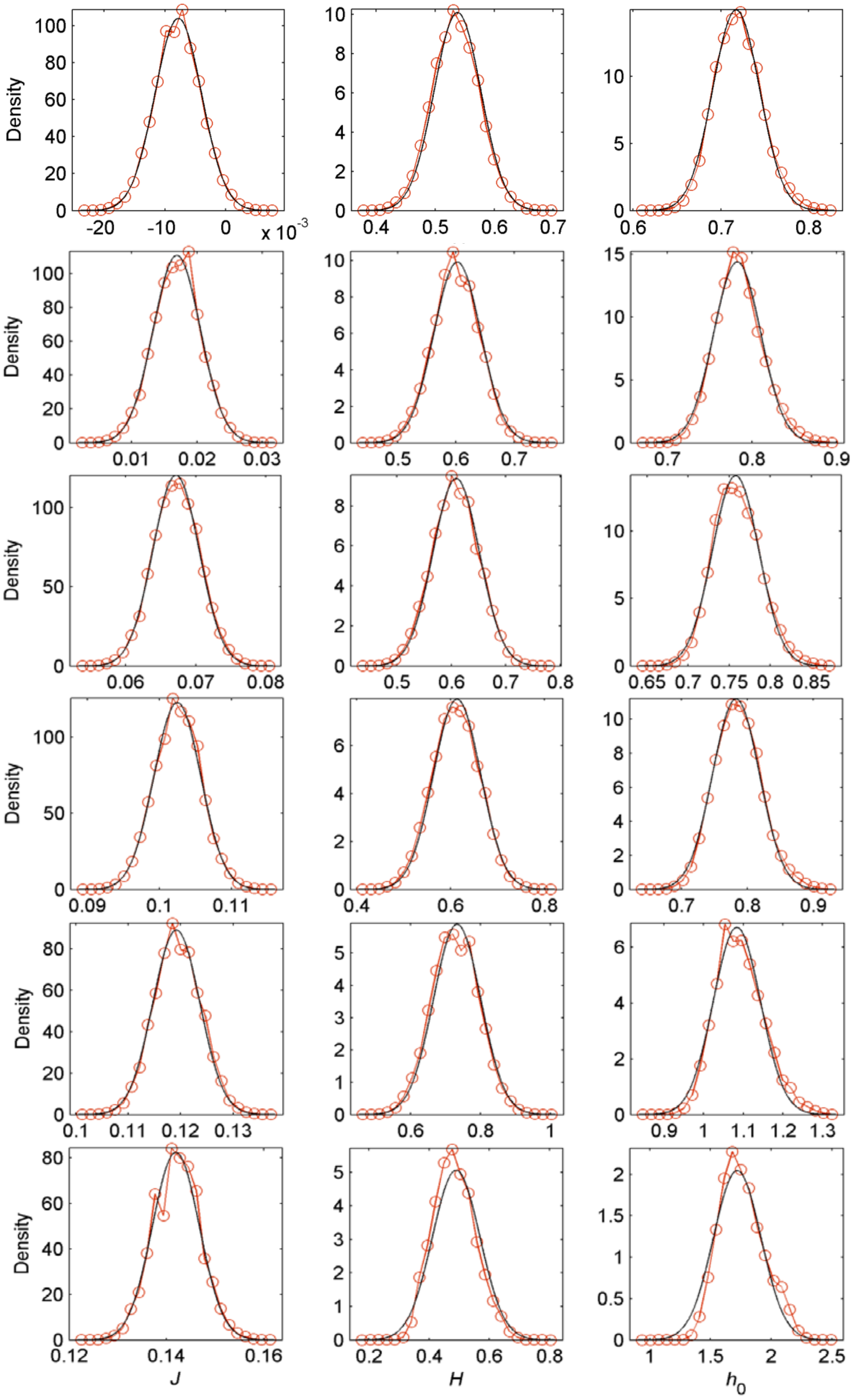

Figure 9.3. True marginal probability distributions of Ising model parameters (circles) and their Gaussian approximations (solid curves), when estimated graph structures are used. Distributions are shown for J (left column), H (centre column), and h_0 (right column), and from top to bottom correspond to the following six J values: 0, 0.04, 0.08, 0.12, 0.16, and 0.20. Calculation: in each case, distributions are shown only for a single randomly picked ensemble.

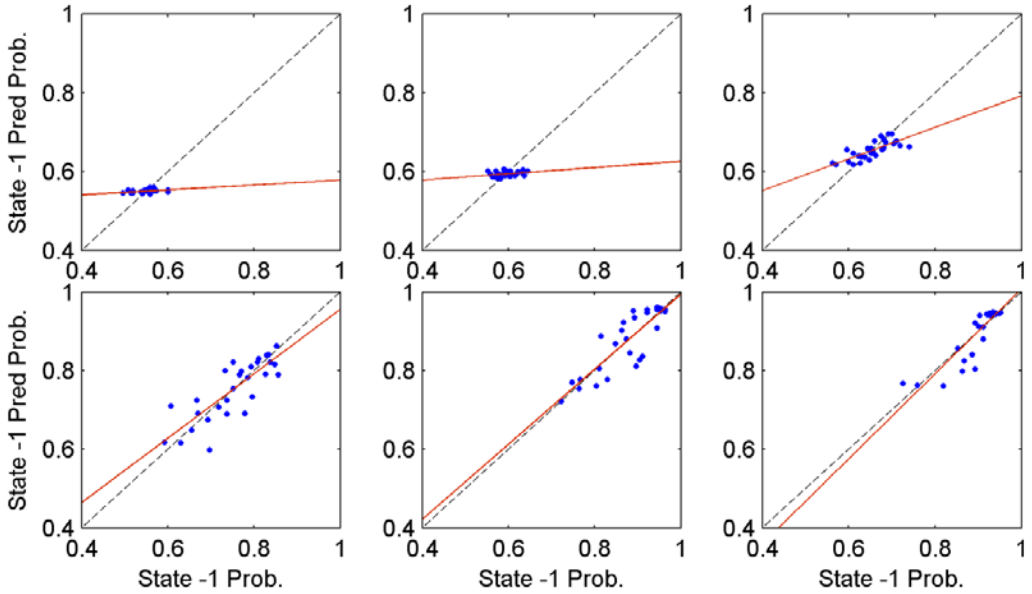

Figure 9.4. Predictions with estimated (structure and parameters) Ising models. Node state -1 probability predictions are shown for each node as a function of data-calculated node state -1 probabilities. Predictions are shown in dots, linear regression lines fitted to predictions in solid lines, and reference curves of optimal predictions in dashed lines. Predictions from top-left to bottom-right correspond to the following six J values: 0, 0.04, 0.08, 0.12, 0.16, and 0.20. Calculation: of three ensembles, shown are those that correspond to minimum average node state -1 prediction errors.

For estimated graph structures, Figure 9.3 shows true and Gaussian approximation-based marginal probability distributions of Ising model parameters with parameterisations corresponding to J values 0, 0.04, 0.08, 0.12, 0.16, and 0.20 (see Section 9.2). With each parameterisation, marginal distributions appear well approximated with Gaussians. Because of the bias errors in J' and $h_0{}'$ with estimated graph structures, the positions of the respective marginal distributions assume the same bias errors. Evidently, parameter uncertainties increase with large J values, as already shown in Figure 9.2.

9.4.2 Model Predictions

To evaluate the entire model identification scheme, the model predictions need to be analysed, as discussed in Section 9.3. Such analysis is particularly important for real networked systems whose true topology and true model parameters are unknown. Here we consider the model predictions using synthetic data like real data to have later a reference to which model predictions obtained with real network data can be compared. In six cases of parameters, Figure 9.4 shows prediction results for state -1 for each node with estimated Ising models. The figure also shows linear regression lines fitted to node state predictions and reference lines indicating optimal predictions. Predictions are poor with a small J, as the neighbouring nodes hardly interact, and conditional state probabilities are mostly determined by the random external loading. With a larger J, the neighbours contribute more, and neighbourhoods are estimated more accurately, improving predictions.

Absolute prediction errors and slope coefficients of fitted linear regression curves are shown in Figure 9.5 as functions of J for all 21 model parameterisations using both estimated and true

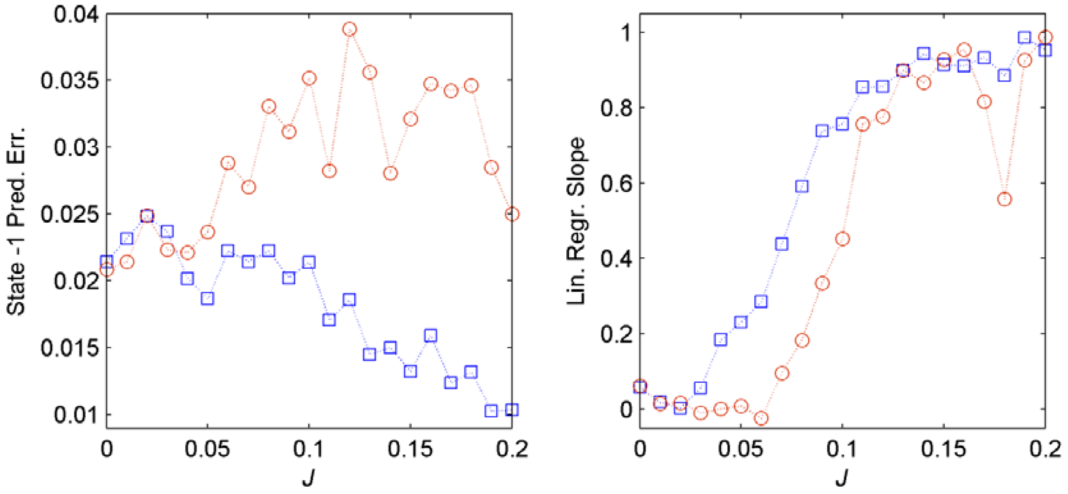

Figure 9.5. Node state -1 absolute predictions errors (left) and slopes of fitted linear regression lines (right) as functions of J. Results are shown for true (squares) and estimated Ising models (circles). Calculation: absolute prediction errors are medians over three ensembles, for which each error is calculated as an average over all nodes. Slope coefficients are medians of the respective coefficients obtained with each ensemble.

Ising models. For the true Ising model, the error is largest with a small J and then constantly decreases as J increases. For the estimated Ising model, however, the error grows as J increases, assuming the largest values at about $J = 0.12$. Consequently, the error alone seems a poor measure for the goodness of the predictions, because they are clearly poor at some of the smallest J where node-specific behaviour cannot be predicted.

With both the estimated and true graph structure, the slopes assume their smallest values at a small J and then quickly rise close to one. However, with the estimated model, the rise happens at larger J than with the true model. The slope coefficients have the problem that they ignore fluctuations taking place around the regression lines. Therefore, both errors and slopes must be studied to judge the goodness of predictions. According to errors and slopes, the best predictions are achieved with the largest J values, though the best topology and parameter estimates are obtained at around $J = 0.12$. In Figure 9.4 again, deviations in average states between nodes are largest at medium coherence values, i.e., at around $J = 0.12$, which probably complicates prediction and explains at least the large prediction errors.

9.4.3 Effect of Data Characteristics

This subsection studies the effect of data characteristics on model parameter identification with the same cases as the effect of data characteristics on topology identification was tested in Subsection 8.5.4, exploiting the respective graph estimates as Ising model structures. Thus not only parameter identification but, in fact, the whole model identification scheme is considered here.

Again the type of node load distribution is first changed from that of the reference case's $\mathrm{Uni}[0,1]$ distribution to an $\mathcal{N}(0.5, 0.25^2)$ distribution and then to an $\mathrm{Exp}(0.58)$ distribution. Model parameter estimates corresponding to the topology identification results in Figure 8.8 are shown in Figure 9.6. Back in Subsection 8.5.4, graph estimates were nearly equally good in all three cases. However, according to the ASSMI, in the exponential distribution case, coherence is

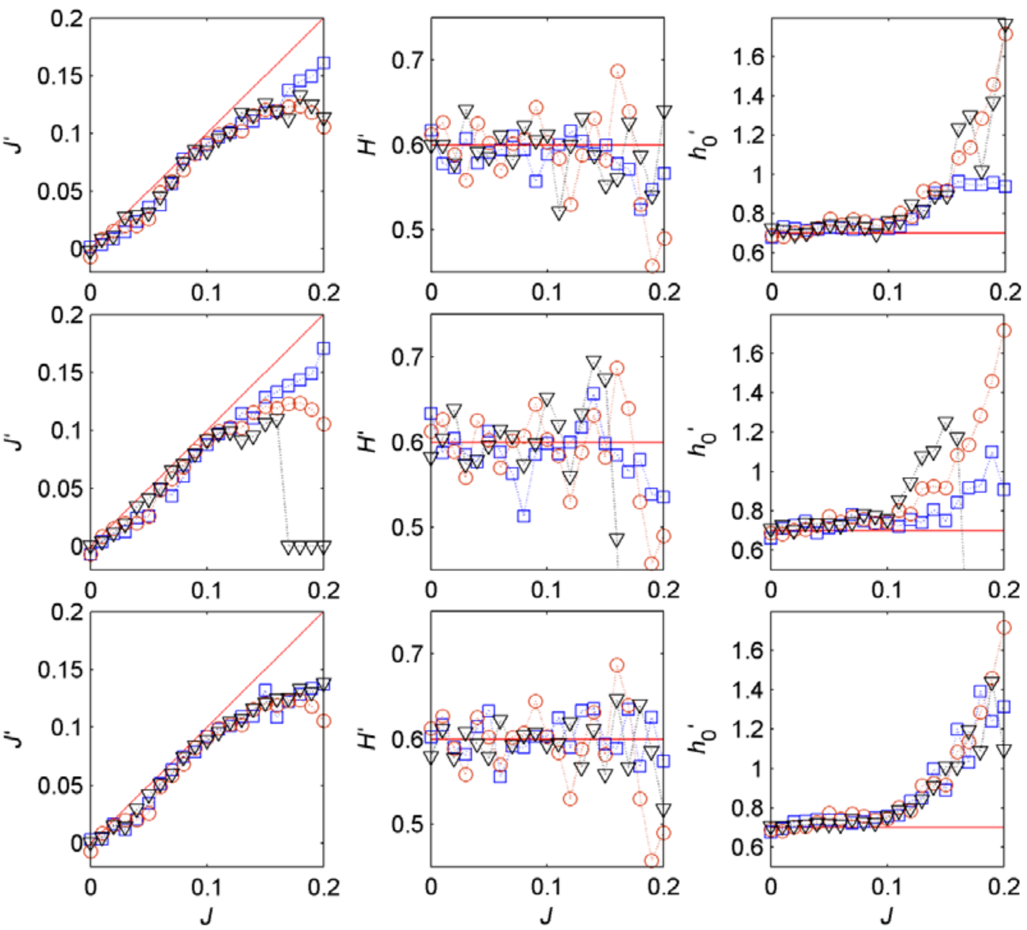

Figure 9.6. Effect of load distribution type (top row), node neighbourhood size (middle row), and data set size (bottom row) on parameter identification. Estimates J' (left), H' (centre), and $h_0{}'$ (right) are shown as functions of J. Top row: exponential (squares), uniform (circles), and normal (triangles) 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). Calculation: measures are medians over the respective values with three ensembles.

greater at larger J values, which probably explains the better parameter estimates obtained in the same case here with large J values.

As Figure 9.6 shows, when the average node neighbourhood size is altered from the reference case's 8.8 to 6.8 and to 10.8, model parameter estimates seem best at $A = 6.8$, where J' and $h_0{}'$ are both quite good even at the largest J values, and where fluctuations in H' are relatively small. In Subsection 8.5.4, we saw that also graph estimates were better with a small A at large J values. At $A = 10.8$, parameter estimates are the worst, even though with small J values, they are as good as in the two other cases. Apparently, with larger A poor graph structure estimates cause poor parameter estimates.

In Subsection 8.5.4, with the two largest data sets, $L = 540$ and $L = 1080$, topology estimates were slightly better than with the reference case's $L = 270$. In addition, the two largest data sets differed only slightly in favour of the larger one. Their differences in parameter estimates are small, as shown in Figure 9.6. In the reference case, parameter estimates are only slightly worse

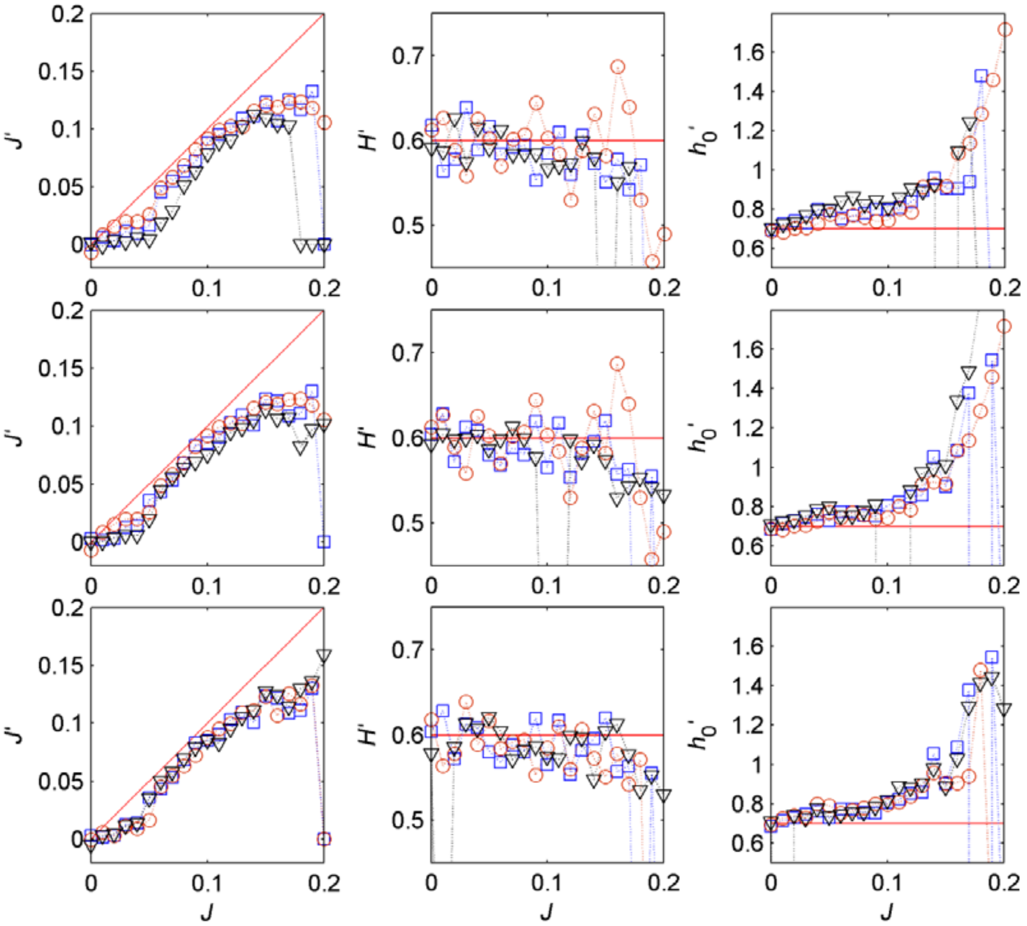

Figure 9.7. Effect of network size on parameter identification. Estimates J' (left), H' (centre), and h_0' (right) are shown as functions of J. Top row: $M = 30$ (circles), $M = 60$ (squares), and $M = 120$ (triangles), each with $L = 270$. Middle row: $M = 30$, $L = 270$ (circles), $M = 60$, $L = 540$ (squares), and $M = 120$, $L = 1080$ (triangles). Bottom row: $L = 270$ (circles), $L = 540$ (squares), and $L = 1080$ (triangles), each with $M = 60$. Calculation: for $M = 30$ measures are medians over the respective values with three ensembles. For other network sizes, measures are calculated from single ensembles.

than in the two other cases, the differences occurring mostly at some of the largest J values. Between $L = 540$ and $L = 1080$, parameter estimates are similar.

The effect of network size on topology estimation was tested in Subsection 8.5.4 with three different schemes. First, $L = 270$ was used for each network size, $M = 30$, $M = 60$, and $M = 120$ nodes, and with the two larger networks, graph estimates appeared quite poor. The model parameter estimates corresponding to this case are presented in Figure 9.7. As expected, $M = 30$ yields the best parameter estimates, though the differences are small in particular between $M = 60$ and $M = 120$. When L was increased linearly in M in Subsection 8.5.4, with all three network sizes, topology estimates were almost equally good. As Figure 9.7 shows, also differences in the corresponding parameter estimates are now smaller, even if slightly, than with a constant L. Finally, L was increased quadratically in M, but was tested only with $M = 60$ in Subsection 8.5.4, where the worst results were clearly those with $L = 270$, whereas the two larger L showed slight differences. However, since parameter estimates here with all three data sizes are similar, the differences shown in graph estimates do not appear in parameter estimates. Yet, the

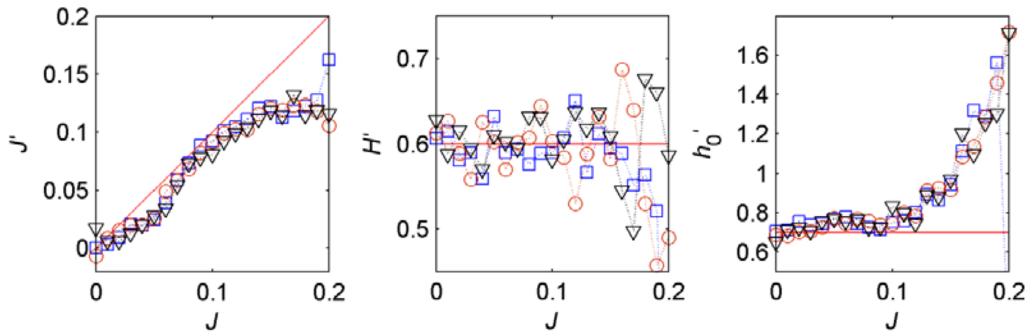

Figure 9.8. Effect of MCMC burn-in steps on parameter identification. Estimates J' (left), H' (centre), and $h_0{}'$ (right) are shown as functions of J. Results are shown with the following number of MCMC burn-in steps: $250x30$ (squares), $500x30$ (circles), and $1000x30$ (triangles). Calculation: measures are medians over the respective values with three ensembles.

results confirm our previous conclusions in Subsection 8.5.4 that L must be increased at least linearly in M for data to be representative.

Finally, in Subsection 8.5.4, the quality of the data used in the analyses was tested with a network of $M = 30$, by studying its topology identification with three MCMC burn-in periods, $250x30$, $500x30$, and $1000x30$ steps. Topology estimates turned out similar in all three cases. Figure 9.8 shows that this occurs also with model parameter estimates, which are similar for all burn-in periods, thus confirming the quality of the reference data set.

10. System Properties

When an MRF model has been successfully identified for a real networked system, the model's qualitative behaviour under varying external conditions becomes of interest. This chapter demonstrates via MCMC simulations of the Ising model how to analyse the dependence of the model phenomena on external node loadings, and how transitions occur in the average node state. Hence it will be shown how the Ising model can be used with MCMC simulations to study network behaviour under particular external conditions. Therefore, simulations make use of true rather than identified graph structure and parameters.

Chapters 8–9 evaluated the model identification method with a somewhat small synthetic 30-node network, because many model parameterisation cases were studied there. This chapter examines a larger, 300-node network, with a topology generated as in Chapter 8. Large network is used for Ising model simulations, because, first, less computation time is now needed owing to fewer parameterisations, and, second, to reduce sample-dependent effects on the model's behaviour caused by a small-size network. With the Ising model defined by this network topology, in Sections 10.1–10.2 qualitative model behaviour is analysed under varying global and local external loadings, after which transient dynamics and model state fluctuations are studied in Section 10.3.

10.1 Behaviour under Global External Loading

As discussed in Chapter 4, network coherence affects drastically the qualitative properties of the Ising model and determines how the average node state $\langle \mathrm{s} \rangle$ changes under varying global external loadings or when subjected to a heavy local loading. In particular, according to Chapter 4, with small coherence, $\langle \mathrm{s} \rangle$ changes smoothly under varying external loading from one extreme state into another, whereas with large coherence, $\langle \mathrm{s} \rangle$ undergoes discontinuous phase transitions. This section studies these phenomena in MCMC Ising model simulations in synthetic network cases.

The Ising model is simulated here with the following cases. The parameter J is given three values, 0.02, 0.2, and 2, while the two other parameters are kept constant, $H = 1$ and $h_0 = 0$. With each J, three threshold values, defining the graph structure from the node location map shown in Figure 10.1, are used with A assuming values 8.76, 11.76, and 14.76. The middle case corresponds to the average number of nodes in the true MTN topology, studied in Chapter 14. With all three A values, all nodes are connected on the topology to form one large node cluster, except for $A = 8.76$, at which two nodes (5 and 46 in Figure 10.1) exist outside this cluster with no neighbours.

A total of nine J–A combinations are thus considered here with focus on the adiabatic dynamics of the Ising model, which is studied as the trajectory of stationary model states produced by MCMC model simulations under a uniformly, and gradually, changing global external loading h; i.e., for each node m $h_m = h$. Adiabatic dynamics means that changes in node loadings are slow enough for MCMC simulation to reach a stationary model state before the next loading change.

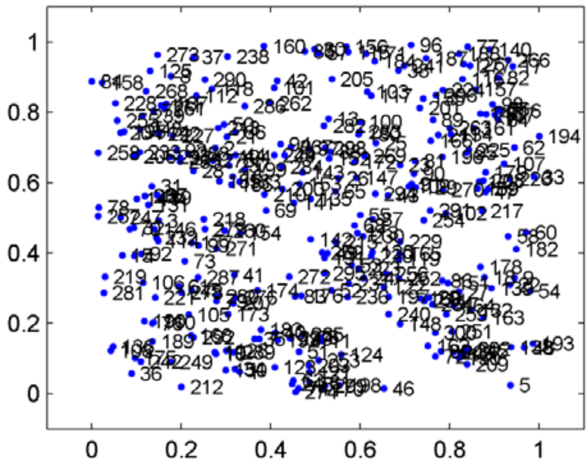

Figure 10.1. Node location map of a synthetic network of 300 nodes.

Section 10.3 later examines transient dynamics and state fluctuations in the Ising model between two stationary states.

With each J, a total of 51 external node loading cases with constant intervals are considered, the range of h depending on the respective J value. With each h, only a single network observation is MCMC-simulated, and the random number generator of MCMC is always initialised to the same state. Because hysteresis may appear in transitions, and the network state may then depend on the initial state, MCMC simulations are run twice with each h: one starting from $\langle s \rangle = -1$ and the other from $\langle s \rangle = +1$. With each loading, an MCMC simulation could just as well start from a previously simulated stationary state, because the model is always simulated until it has converged to its stationary state. Owing to hysteresis, it makes a radical difference in the region of two stable solutions if simulations are started near $\langle s \rangle = -1$ or $\langle s \rangle = +1$. The choice determines the branch of the hysteresis curve selected. Because the interaction term in the Ising model is additive in the number of neighbours, the ratio $\tilde{J} = J/A$ is essential to the model's behaviour. Table 10.1 summarises the value of this ratio in all nine cases.

Simulation results are shown in Figure 10.2 with $\langle s \rangle$ given as a function of uniform global external loading value. At each of the three smallest $\tilde{J}$ values, transitions between $\langle s \rangle = -1$ and $\langle s \rangle = +1$ occur smoothly; the two state curves corresponding to two simulation initialisations overlap and hence no hysteresis occurs. However, from $J = 0.2$ and $A = 8.76$ on, the two simulation paths diverge. In addition, changes in $\langle s \rangle$ become steeper as a function of loading and, with the interval applied when varying the external loading values, occur abruptly within a few

Table 10.1. Values of ratio $\tilde{J} = J/A$ with respect to J and A.

J	A	$\tilde{J}$	J	A	$\tilde{J}$	J	A	$\tilde{J}$
0.02	8.76	$2.28\text{x}10^{-3}$	0.2	8.76	$2.28\text{x}10^{-2}$	2	8.76	$2.28\text{x}10^{-1}$
0.02	11.76	$1.70\text{x}10^{-3}$	0.2	11.76	$1.70\text{x}10^{-2}$	2	11.76	$1.70\text{x}10^{-1}$
0.02	14.76	$1.36\text{x}10^{-3}$	0.2	14.76	$1.36\text{x}10^{-2}$	2	14.76	$1.36\text{x}10^{-1}$

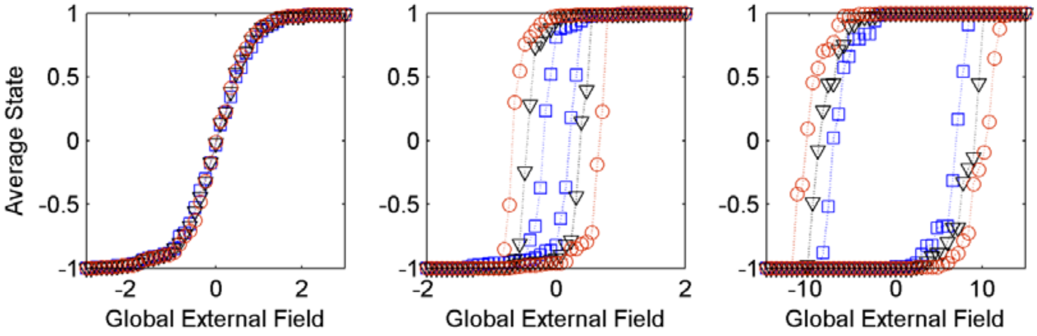

Figure 10.2. Model state behaviour under global uniform external loading. Average state $\langle s \rangle$ is shown as a function of h with three J values, 0.02 (left), 0.2 (centre), and 2 (right), and each with three A values, 8.76 (squares), 11.76 (triangles), and 14.76 (circles). Calculation: results with each h are average node states in a single simulated ensemble. With each h, two ensembles are simulated, one starting from $\langle s \rangle = -1$ and the other from $\langle s \rangle = +1$.

external loading steps. If the coherence were even larger, eventually all (or almost all) nodes would undergo simultaneous state changes with a discontinuity appearing in $\langle s \rangle$. Evidently, the larger the $\tilde{J}$ value, the larger the gap between two simulation paths; i.e., the more coherently the nodes act, the larger the external loading needed for state changes to occur. These are well-known properties of the standard Ising model and repeated here to illustrate hysteresis in finite-size but large networks and to demonstrate the effectiveness of MCMC simulation.

10.2 Behaviour under Local External Loading

According to the previous Ising model simulations, coherent networks with high connectivity and high interaction have the advantage that they seem to bear rather large external loadings without marked changes in $\langle s \rangle$. However, their drawback is that under heavy global external loading, they may exhibit a sharp threshold value for h, above (below) which they abruptly switch from one coherent state into another. Hence a coherent network near this threshold is very sensitive to even the smallest changes in local or global external loading.

A network's sensitivity to a heavy local external loading under uniform global external loading is

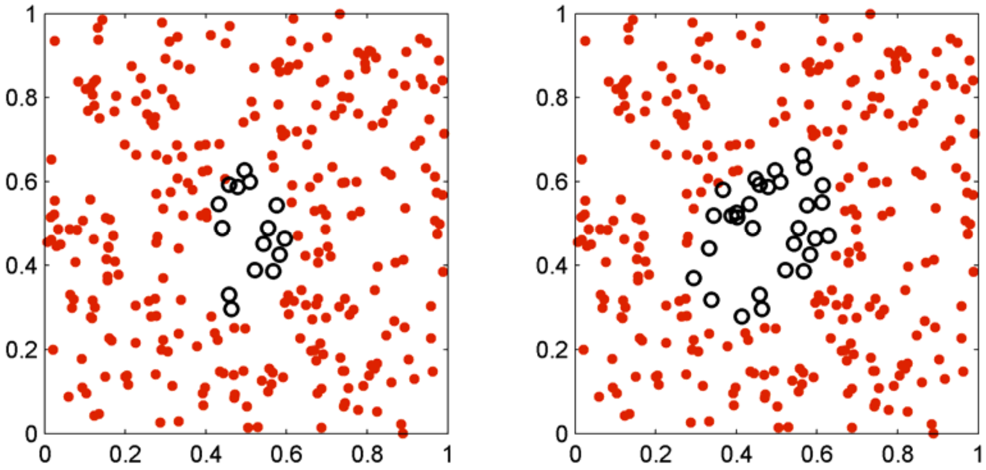

Figure 10.3. Heavy local external loading (circles) affecting 15 (left) and 30 nodes (right).

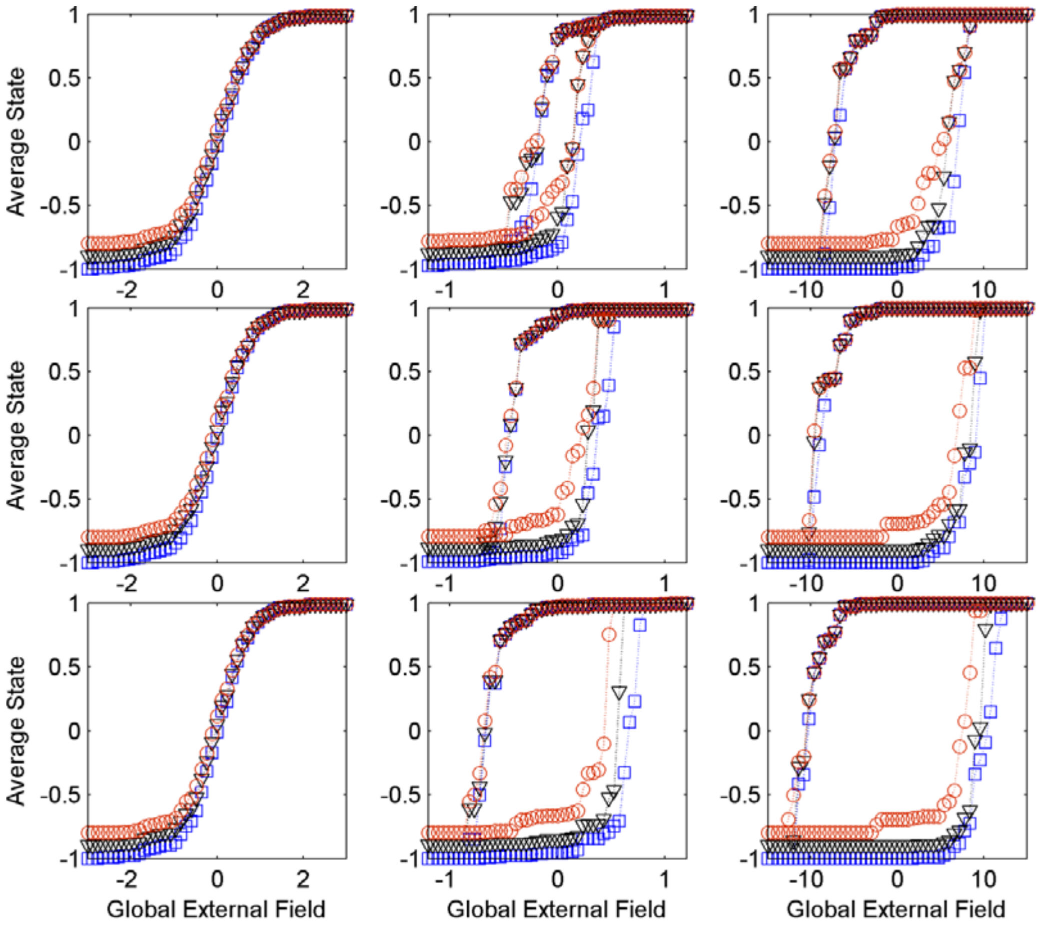

Figure 10.4. Model state behaviour with a heavy local external loading under global uniform external loading. The average state $\langle s \rangle$ is shown as a function of h with three J values, 0.02 (left column), 0.2 (centre column), and 2 (right column), and each with three A values, 8.76 (top row), 11.76 (middle row), and 14.76 (bottom row). Heavy local loading affects 0 nodes (squares), 15 nodes (triangles), and 30 nodes (circles). Calculation: results with each h are average node states in a single simulated ensemble. With each h, two ensembles are simulated, one starting from $\langle s \rangle = -1$ and the other from $\langle s \rangle = +1$.

studied here by simulating the Ising model under the previous nine J–A combinations. In all these cases, two local loading situations are considered: first, a 15-node cluster is selected from the center of the network, and the nodes are subjected to a constant heavy local loading ($h_{local} = 60$), and, second, the number of affected nodes in the cluster is increased to 30, including the previously 15 nodes. Figure 10.3 presents two node location maps that show the nodes that are affected by the local external loadings in the two cases.

Figure 10.4 shows Ising model simulation results of nine J–A combinations with local loading affecting clusters of 15 and 30 nodes (results of the previous section without local loading are shown for reference). In all cases with $J = 0.02$, a heavy local external loading hardly affects $\langle s \rangle$, because the nodes act nearly independently. At $J = 0.2$ and $J = 2$, nodes act more coherently, and because of local loadings, transition now takes place already at smaller h values. Not surprisingly, the larger the group of affected nodes, the earlier the transition appears when going from $\langle s \rangle = -1$ to $\langle s \rangle = +1$ with an adiabatically increasing h.

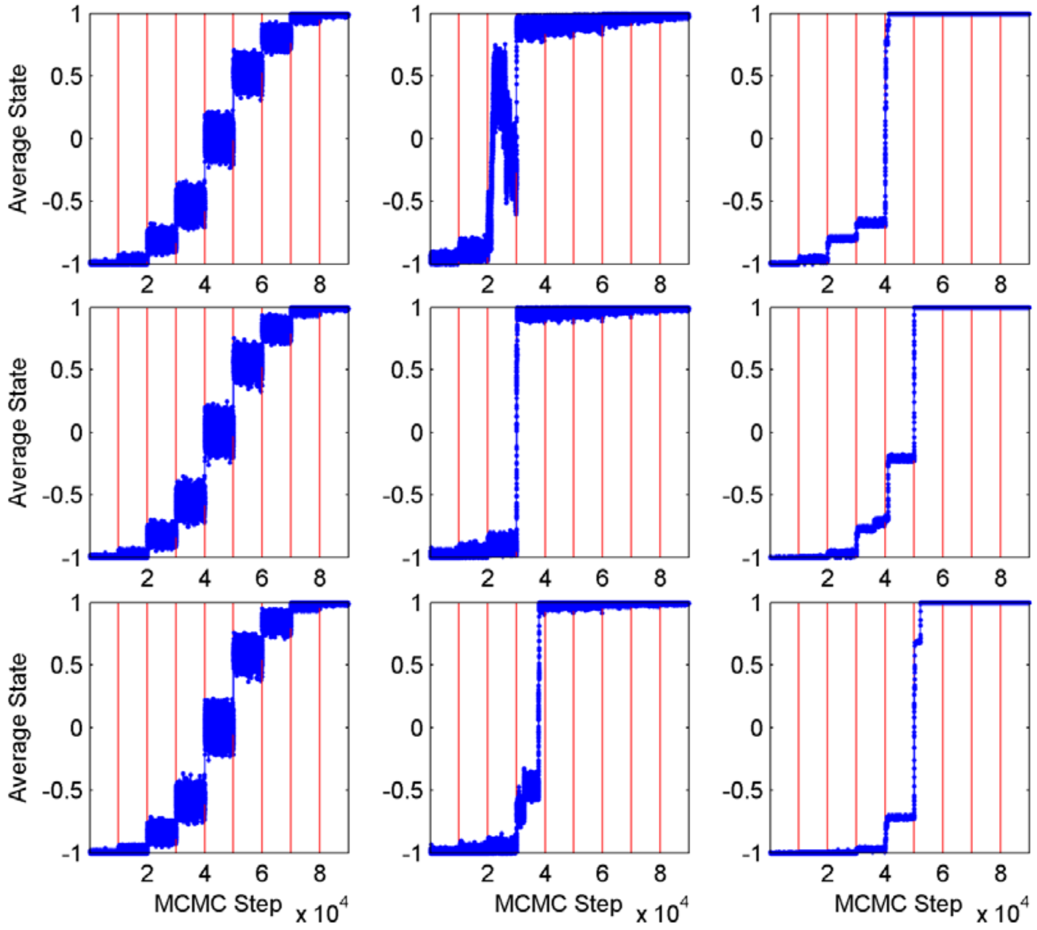

Figure 10.5. Model state fluctuations and transient dynamics under varying global external loading. Average state $\langle s \rangle$ is shown as a function of MCMC simulation step (dotted lines) for three J values, 0.02 (left column), 0.2 (centre column), and 2 (right column), and each with three A values, 8.76 (top row), 11.76 (middle row), and 14.76 (bottom row). Vertical lines mark the spots were h changes. Calculation: results at each MCMC step are average node states in a single simulated ensemble. The first MCMC simulation step starts from $\langle s \rangle = -1$, and the rest always from a previous simulated state.

10.3 MCMC Dynamics

Because MRF models are statistical models describing joint probabilities of node states, true dynamics does not exists for the models themselves. Instead, when sampling the joint probability distribution, one may define the dynamics to be that of MCMC. This approach is used here to study dynamic response to dynamic changes under global uniform external loading. We assume that true network dynamics can be described coarsely with a trajectory of samples of model states simulated with MCMC methods. In the previous two sections, changes were made in h so that the MCMC of the model always converged to its stationary state before a new change was made. This led to a trajectory of stationary, or equilibrium, model states, which were then considered to describe the equilibrium, or adiabatic, dynamics of the system. This section examines transient dynamics and the model's convergence to its stationary state after a change in h, describing thus a trajectory of intermediate model states between the initial and final stationary state.

An MCMC step in which each node is updated once in a regular order is here considered to produce one transient state. A series of samples produced in this way forms a trajectory of transient

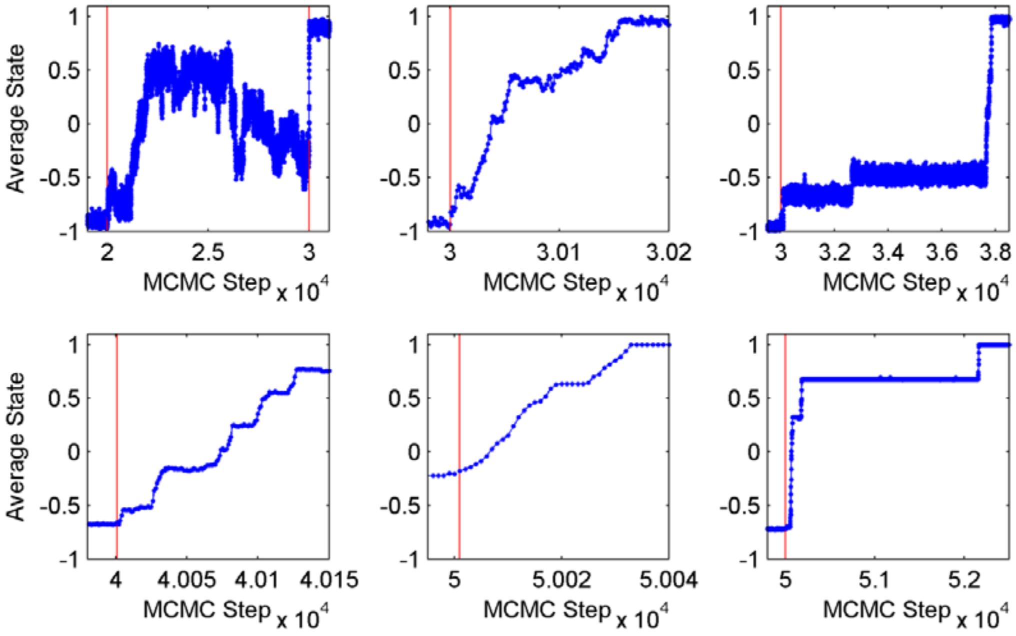

Figure 10.6. Zoom-in plots of transient dynamics of changes in $\langle s \rangle$ shown in Figure 10.5. The centre column in Figure 10.5 corresponds here to the top row, whereas the right column in Figure 10.5 corresponds here to the bottom row. Original plots are zoomed here to show the ranges of MCMC steps with most drastic changes.

states such that the next transient state depends only on the previous state, and that the samples are thus produced by a Markov chain. Furthermore, this trajectory describes how the network moves from one stationary state into another. Of course, because the MRF model is a statistical model, it has no single trajectory but instead a huge number of possible trajectories. Which trajectory is assumed depends on both randomness related to the updates of node states and the ordering of the node state updates. Similarly, instead of a single stationary state, it has a stationary distribution, which is the MRF joint probability distribution under given conditions. The aim is here to study one such state-path realisation, because in a true network only a single state-path realises among all the possible ones.

As an example of transient dynamics and state fluctuations, a synthetic network of 300 nodes is again considered with nine J–A value combinations. Node loadings are uniform through the nodes and increased gradually by the following load values: $[-3, -2, -1, -0.5, 0, 0.5, 1, 2, 3]$ for $J = 0.02$, $[-1, -0.5, 0, 0.3, 0.5, 0.6, 0.7, 1, 1.5]$ for $J = 0.2$, and $[-1, 1, 3, 5, 7, 9, 11, 13, 15]$ for $J = 2$. After each change in the node loadings, the Ising model is simulated from the previous state 10^4 MCMC steps before another change. During MCMC updates, the model is assumed to converge to its stationary probability distribution. The initial state of the first simulation is always the global state -1; then after a loading change, the next simulation always starts from the end of the previous one.

Simulation results for the nine J–A cases are shown in Figure 10.5. In general, the model converges to its stationary distribution quite fast and usually requires only a few MCMC steps. However, with abrupt or drastic changes in $\langle s \rangle$, it clearly requires many extra MCMC steps to reach a stationary state. With $J = 0.2$ and $A = 8.76$, a loading change from $h = -0.5$ to $h = 0$ induces particular instability with the network state fluctuating heavily and not converging on a stationary state before another loading change is made. Apparently, a loading at $h = 0$ is just large enough to cause an abrupt change from one coherent state into another and heavy fluctuations, similar to those occurring in continuous phase transitions in the Ising model. A further loading increase finally drives the network into a coherent state $+1$. A similar slow convergence also takes place in the other cases corresponding to the two larger J values, better seen in Figure 10.6, which shows zoom-ins of discontinuous changes occurring in $\langle s \rangle$ in Figure 10.5.

Figure 10.6 shows that though transitions in these cases appear discontinuous when viewed only from stationary states, relatively many MCMC steps are yet needed to achieve stationary states. Consequently, transitions, in fact, occur as cascades of avalanches of node states, as discussed in Chapter 4, where node clusters change their states simultaneously, forcing other node clusters to undergo state changes as well. The larger the coherence, the larger these node clusters become. As an anticipated conclusion, the larger the interaction and the average neighbourhood size, i.e., the larger the coherence, the smaller the state fluctuations, and the more rapidly $\langle s \rangle$ changes and hence converges to a stationary state. In addition, according to Figure 10.5, in general, the closer the network is to the point where $\langle s \rangle$ is about to change, the larger the state fluctuations become.

11. Introduction to MTN Data

In Chapters 11–14, the topology and parameter identification methods introduced and evaluated in Chapters 8–10 are applied to real mobile telecommunications network (MTN) data. The Ising model is chosen to represent a real MTN case. The identification methods are applied to networks of 30 and 132 BTS cells. The smaller network demonstrates the methods' applicability more easily, whereas the larger network puts the methods to a more realistic test.

The present chapter introduces GSM MTN data with variables of the data introduced in Section 11.1. Section 11.2 focuses on preprocessing the state data from a multivariable format into a single state variable per BTS cell node, which in Section 11.3 is then further discretised for MRF modelling. Finally, Section 11.4 examines the logical and physical topologies of the MTN and discusses the similarities of the two types of topology information.

11.1 BTS Data

Raw BTS data is continuous-valued and originates from thousands of counters measuring various aspects of network performance. Key Performance Indicators (KPI) summarise various cost efficiency and quality-related properties of BTS cells [145], [80], [156]. Each KPI is calculated by combining the information given by the counter variables, and because averaged over a time period, such as an hour or a day, KPI values are typically occurrence percentages of defined types of events during a time period. Hence a typical KPI variable assumes a value in the interval [0, 100].

Here the KPI variables analysed are evaluated hourly. The following KPIs are considered: Handover Failure (HF), Handover Failure Due to Blocking (HFB), Stand-alone Dedicated Control Channel Success (SDCCHS), Traffic Channel Success (TCHS), and Calls KPI. Handover means transferring an on-going mobile station connection from one cell to another, either within a single BTS or between two BTSs, when the quality of the connection deteriorates between mobile station and first cell [150]. In addition, an intra-cell handover is possible [150]. HF is the percentage of failed outgoing handovers from a cell. An HFB is similar to an HF, but describes the proportion of failed handovers due to blocking of the receiving cell to all handover failures.

Traffic channels (TCHs) carry speech and data between mobile stations and the MTN. A TCHS contains both speech and data traffic and indicates the rate of successful TCH transactions out of all TCH transactions. A TCH transaction is an event of transferring either speech or data. In a GSM MTN, control channels (CCHs) transmit service data, such as signalling or synchronising data. The SDCCHS indicates the rate of successful SDCCH transactions among all SDCCH transactions, signifying the various control signals between network and mobile stations. A Calls KPI is the number of calls handled by a single BTS cell and thus an exceptional KPI, because it takes non-negative integer values rather than occurrence percentages. Calls KPI is related to the Erlang unit that is generally used for measuring the volume of traffic in telecommunications networks. However, in addition of the number of calls handled, Erlang measure also takes into account, e.g., the duration of the calls.

11.2 Preprocessing of BTS Data

Because in MRF modelling node states and loadings are assumed to be univariate, the information given by several KPI data variables must be compressed into single state and load variables. With the Ising model, state data must also be binarised into -1s and $+1$s, whereas load data may assume either discrete or continuous values. The Calls KPI is used here as load data, because it provides information about the loading of cell nodes. The Calls KPI is only rescaled by dividing all its values by its median value over all cell nodes. Scaling here provides a typical load value somewhat equal in magnitude to state data values and affects only the parameters of MRF modelling. Because the capacity, the amount of traffic a cell can handle, may vary, another approach that could have been taken here is to scale the loading values of each cell node separately with their respective capacity. However, here the capacities are unknown in the data set, but assuming that the loading for each cell reflects the cell's capacity, the loading of each cell could have been scaled, e.g., with the median cell load value.

The other four KPIs considered here, HF, HFB, SDCCHS, and TCHS, are related more to the states of BTS cell nodes. To combine these KPIs, they are first scaled trivially to the range $[0, 1]$ such that values 0 and 1 correspond to the worst and best possible node performance. As for HF and HFB KPIs, the meaning of their extreme values is opposite to that of SDCCHS and TCHS KPIs, their scale is reversed. The obtained multivariable KPI data is compressed here into a single measure with the following procedure adopted directly from [79].

Let us denote the values of the scaled KPI variables for (cell) node m in a network observation l by $s_{\mathrm{HF}}(m,l)$, $s_{\mathrm{HFB}}(m,l)$, $s_{\mathrm{SDCCHS}}(m,l)$, and $s_{\mathrm{TCHS}}(m,l)$. Given that each KPI value 1 indicates optimal node performance, a univariate continuous-valued node state performance measure $s_{\mathrm{c}}(m,l)$ is obtained by taking the Euclidean distance of the KPI variables from the optimal:

$$\begin{aligned} s_{\mathrm{c}}(m,l) = \{&[1-s_{\mathrm{HF}}(m,l)]^2 + [1-\ s_{\mathrm{HFB}}(m,l)]^2 \\ &+[1-s_{\mathrm{SDCCHS}}(m,l)]^2 + [1-s_{\mathrm{TCHS}}(m,l)]^2\}^{\frac{1}{2}}. \end{aligned} \tag{11.1}$$

Values $s_{\mathrm{c}}(m,l)$ are continuous with the minimum value zero and the maximum value 2.

The KPI variables used here as state data contain a lot of missing values, except SDCCHS, which has only a few missing values for each cell on the average. In the other state-related KPIs, on the average of about 10% of the cell observations are missing, whereas the loading-related Calls KPI contains no missing values at all. From the original data set, first those cell nodes and observations containing a lot of missing values are removed. Then for each cell observation, each missing KPI value is replaced by the average of the respective non-missing KPI values; e.g., if the values of HF and HFB were missing, then both of these values would be replaced by the average of the SDCCHS and TCHS values at that observation. This rather harsh handling of the missing values can be justified by the use of Eq. (11.1) to combine the KPI variables into a joint measure, for which each KPI contributes equally. By using the four KPIs to construct the joint measure of Eq. (11.1), therefore provides reasonable state estimates even when some of the KPI observations are missing.

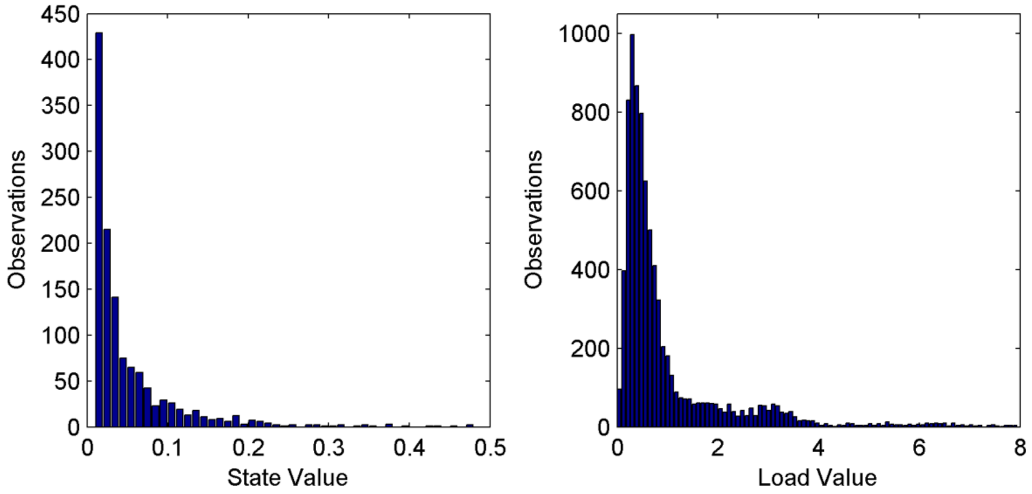

Figure 11.1. Histograms of continuous-valued state (left) and load (right) data of an MTN of 30 BTS cell nodes.

Figure 11.1 shows overall distributions as histogram estimates and Figure 11.2 the distribution information of individual cell nodes as box plots for pre-processed continuous-valued state and load data. Because most state data assume zero value, or are close to it, the first bar is omitted from the histogram in Figure 11.1, which covers the range 0–0.005 and contains 6821 observations of a total of 8070 observations, and from which 1993 assume the exact value of zero, which is 24.7% of all the observations. The mean, median, and standard deviation values of the state and load data are summarised in Table 11.1.

The state data histogram resembles an exponential distribution, whereas the load data is close to the Poisson distribution but with a fatter tail. Indeed, in telecommunications networks the arrival of calls is typically modelled as a Poisson process, whereas some other quantities related to the state of a node are described with exponential distributions, such as the service time in a node [60]. Figure 11.2 shows clear individual differences in the values of cell node variables between both state and load values; on average, some nodes perform worse than others, and some are clearly more heavily loaded than the average node. Yet, a heavy loading and a poor node state do not seem to correlate clearly, a phenomenon partly explained by node interaction by which neighbouring nodes affect each other's state. The capacity of nodes may also vary, and thus a heavily loaded node may perform well due to its high capacity. Indeed, if this is the case in the data set applied here, it is not taken into account in the MRF modelling, because the cell loadings are not scaled with the respective capacities, which are unknown.

Table 11.1. Mean, median, and standard deviation of continuous-valued state and load data.

	Mean	Median	Standard Deviation
State values	0.00930	0.00053	0.03337
Load values	0.964	0.532	1.185

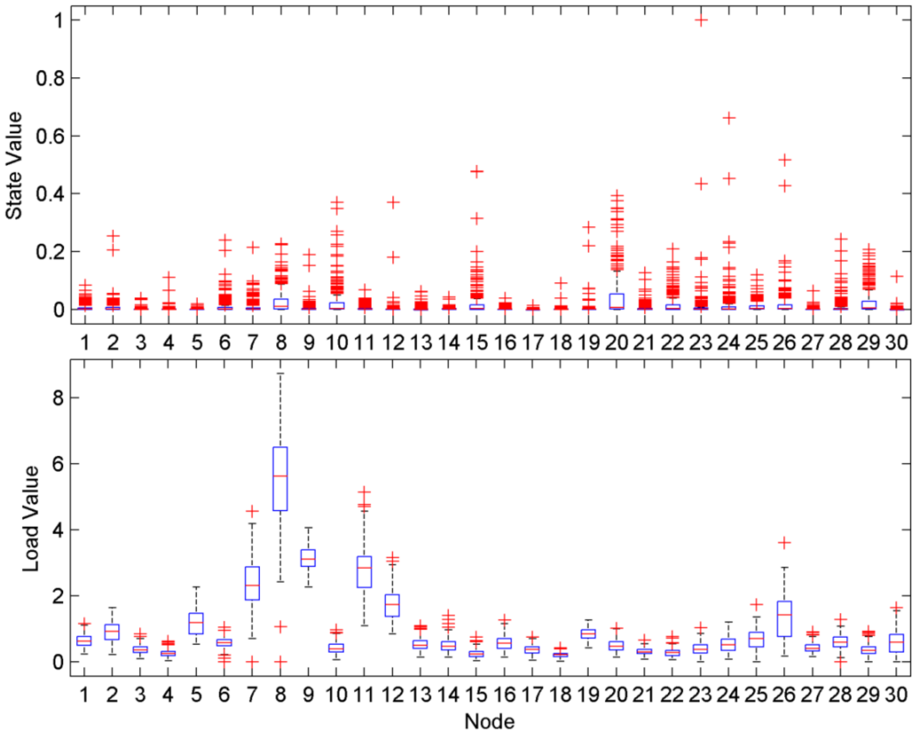

Figure 11.2. Box plots of continuous-valued state (top) and load (bottom) data as functions of 30 BTS cell nodes. The horizontal lines inside the boxes denote median values, while the bottom- and top horizontal lines of the boxes define the lower- and upper quartile values. The dashed lines (whiskers) at the ends of each box extend to the adjacent values in the data. The plus signs denote outliers.

11.3 Discretisation of BTS Data

In general, for discrete MRF models, the continuous-valued state data $s_c(m,l)$, defined by Eq. (11.1), must be discretised. For the Ising model, $s_c(m,l)$ is here discretised into binary values by thresholding the state values. If a continuous state value is below the threshold, state -1 is assumed, and if above, state $+1$ is assumed. Hence, state -1 is here chosen to indicate good node performance, whereas state $+1$ indicates poor node performance. Alternatively, also other discretisation methods, such as clustering-based classification, can be used [80], [156], [157].

The threshold, chosen here uniform for all cell nodes, may be chosen according to a desired ratio for the amount of data in each state, e.g., using the domain knowledge about the usual proportions of decent and poor node states. Because 24.7% of the continuous-valued state data assume zero value, this marks the minimum amount of data that can be interpreted as having the state value -1. In principle, an appropriate threshold value can be determined by inspecting the distribution of the state data. However, here we seem to have no such obvious threshold value, nor any prior knowledge to help choose one.

For a thorough analysis, several discretisation thresholds are analysed here, which divide the data into two binary states so that the proportion of data in state -1 ranges from 0.275 to 0.725 at 0.025 intervals, yielding a total of 19 cases. With discretisation thresholds 0.3, 0.35, 0.4, 0.45, 0.5, 0.55, 0.6, 0.65, and 0.7, the proportion of observations in state -1 is shown in Figure 11.3 for each cell node. In some nodes, the proportion of -1 states is strikingly more sensitive to the

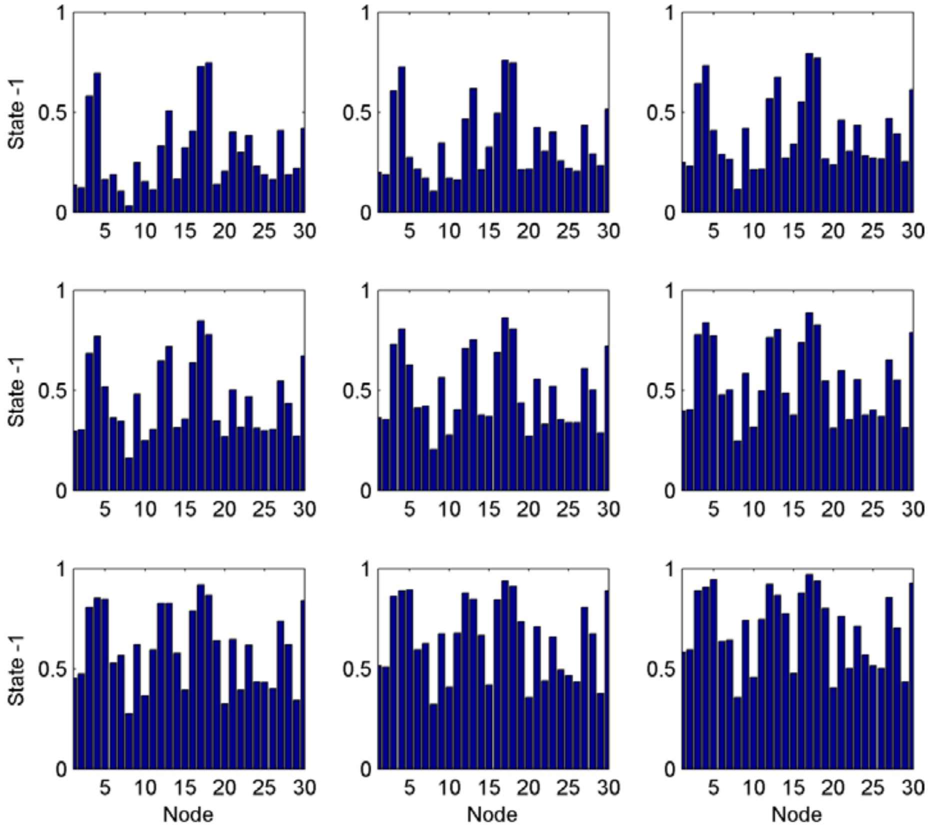

Figure 11.3. Proportions of observations assuming state value -1 as functions of 30 BTS cell nodes. Results from top-left to bottom-right correspond to the following discretisation threshold values: 0.3, 0.35, 0.4, 0.45, 0.5, 0.55, 0.6, 0.65, and 0.7.

threshold than in other nodes. For example, as the threshold is increased, the data in state -1 for node 5 increases dramatically compared to most other nodes. Therefore, the threshold value is expected markedly to affect MRF modelling as well.

11.4 Logical and Physical Topologies

As discussed in Chapter 2, MTNs comprise topologies because of both physical geographic locations of BTS cell nodes and their logical connections. Physical locations can be directly visualised as a two-dimensional map of BTS cell locations—overlapping nodes represent cells belonging under the same BTS. Visualisation of logical connections, however, is more complex, and here logical (graph) distances are first calculated for each $M(M-1)/2$ node pair, after which non-metric multidimensional scaling, introduced in Chapter 5, is applied to the distances to form an approximate node location map for the original logical distances in two dimensions.

Figure 11.4 shows both node location maps. For better visual comparison, the maps are first scaled with their respective Frobenius norms, and then the map based on logical distances is Procrustes-transformed with respect to the physical location map. Though the two maps differ in many details, overall they are similar. In particular, for most nodes their neighbourhoods look very much alike. Obviously then, nodes physically close to each other are also logical neighbours. This is further justified in Figure 11.5, which shows mean and median values of distances between nodes on the physical node location map as a function of their respective logical distances.

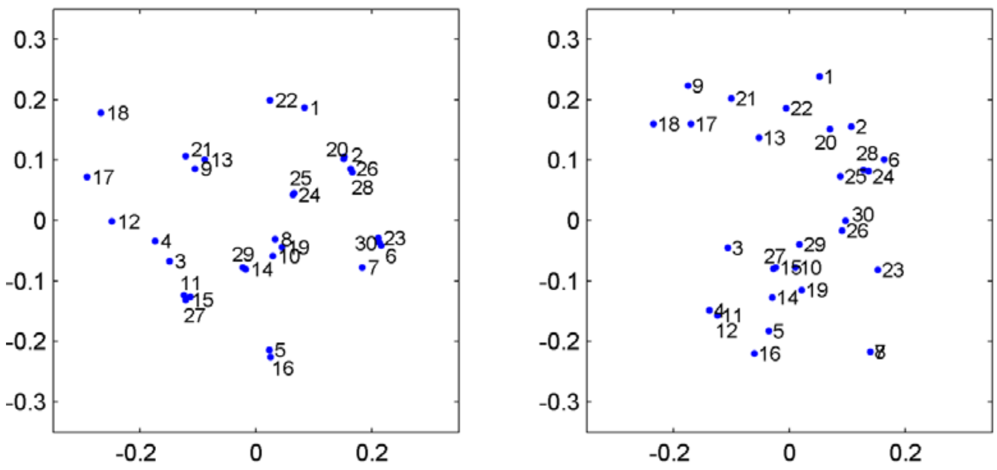

Figure 11.4. Logical (left) and physical (right) location map presentations of 30 BTS cell nodes. Calculation: the physical location map simply shows the physical locations of nodes scaled with the respective Frobenius norm. The logical location map is obtained by applying non-metric MDS to logical distances, calculated for nodes from logical neighbourhood relations. The map is then further scaled with the respective Frobenius norm and Procrustes-transformed with respect to the physical location map.

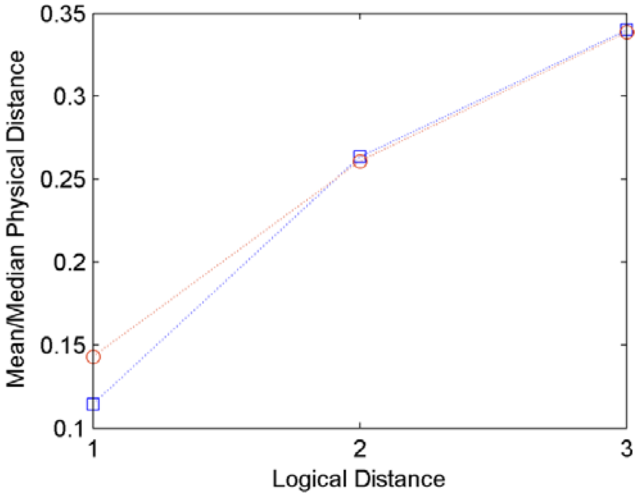

Figure 11.5. Comparison of physical and logical internode distances of BTS cells. Mean (circles) and median (squares) values of physical internode distances according to a physical node location map are shown as a function of corresponding internode logical distance according to graph structure. Calculation: for each node pair with a certain logical distance, mean and median values are taken over the physical (Euclidean) distances between the nodes.

12. Topology Identification for MTNs

In this chapter, the MGMN topology identification method introduced and evaluated in Chapter 8 is applied to real MTN data presented in Chapter 11 to identify the MTN's topology from the data and to use it as a graph structure in MRF modelling of the MTN. As discussed in Chapters 2 and 11, MTNs contain topology information because of both their logical node connections and the physical locations of the nodes. Though these two pieces of topology information led to rather similar neighbour relations in Chapter 11, they complement each other, and their combined effect manifests itself in MTN data. Hence a data-estimated topology combines the effect of the two topologies into a single graph structure, which can then be used for graph structure in MRF modelling.

As discussed in Chapter 11, to apply the Ising model, the state data of BTS cell nodes must be thresholded into two binary states. However, because we have no prior information available about choosing the threshold value, we study topology identification using several discretisation threshold values. On the other hand, we use prior information to define a graph structure from an obtained MDS node location map by selecting the distance threshold value so as to have an average of 8.73 neighbours for each cell node, which is the average number of neighbours according to the MTN logical topology. Though we use this domain-knowledge-based prior information to define the threshold distance value, we also test the impact of the threshold value on results by varying it around the value based on the prior information.

In this chapter, Section 12.1 first studies network coherence as a function of discretisation threshold value. Section 12.2 presents the data-based node location map and graph structure estimates obtained by the MGMN method. The effect of the threshold distance on the graph structure is studied in Section 12.3, and Section 12.4 examines how the method works with large networks. MRF model parameter identification based on estimated graph structures is later discussed in Chapter 13. Though only a single data set is applied in Chapters 12 and 13, most calculations are repeated three times because of stochastic effects related to the calculation of MDS, SSMI, and MRF model parameter estimation.

12.1 Coherence in Network Data

In Figure 12.1, the network coherence measures R', R_N, and ASSMI, introduced in Subsection 8.5.2, are shown as functions of the discretisation threshold value for MTN data. Estimated neighbourhoods and estimated parameter values are used to calculate R' and R_N. All measures here seem rather similar in their functional forms, but R_N differs slightly, because it is relatively small at some of the smallest discretisation thresholds. Overall, the coherence is the highest at small discretisation thresholds and then degrades monotonically as the threshold value increases.

Figure 12.1 shows the mean and median values of the SSMI as functions of the logical (graph) distance of the MTN logical topology with the following discretisation thresholds: 0.325, 0.4, 0.475, 0.55, 0.625, and 0.7. Some later results in this chapter are also studied in depth with these six discretisations. The measures are calculated like those in Subsection 8.5.2 to describe

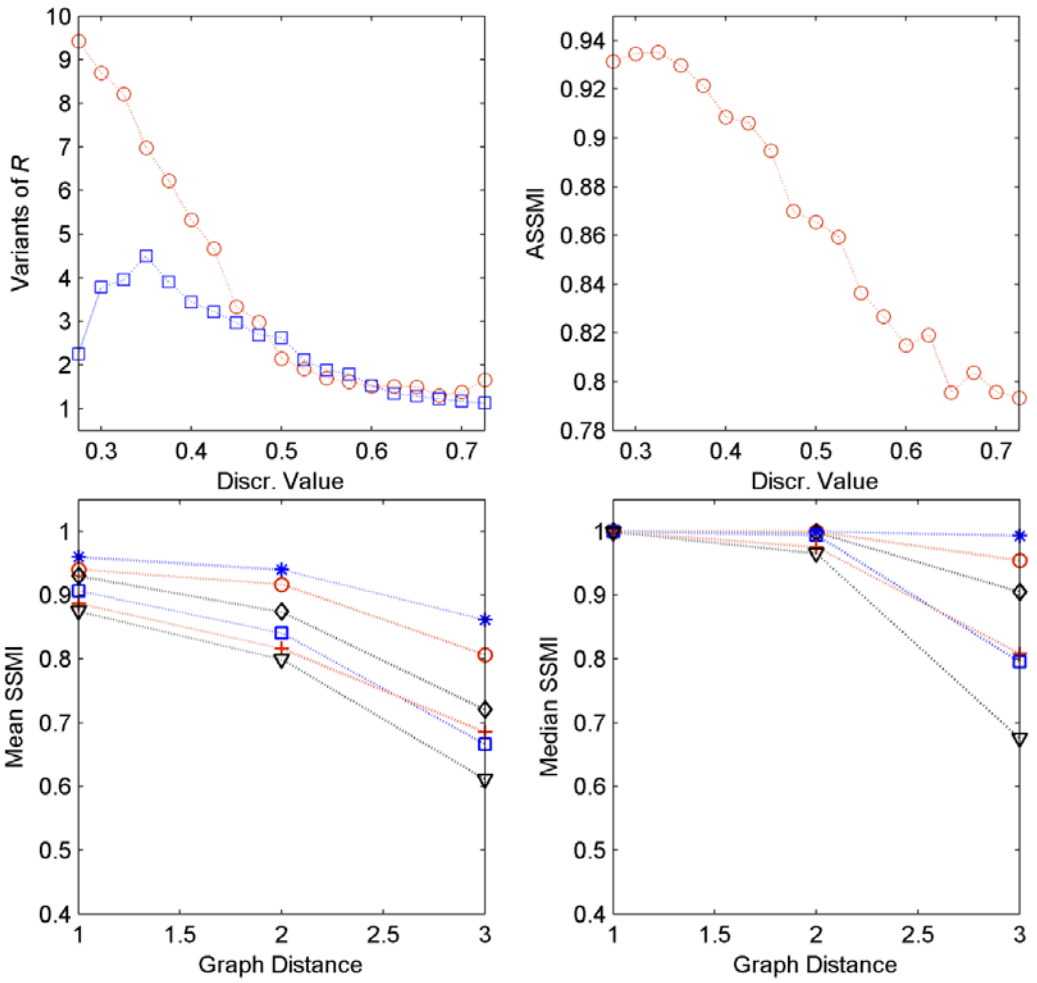

Figure 12.1. Network coherence with the real network of 30 nodes. Top-left plot presents R' (circles) and R_N (squares), and the top-right plot ASSMI, as functions of the discretisation threshold. Mean (bottom-left) and median (bottom-right) values of SSMI are given as functions of the logical (graph) distance of nodes with discretisation thresholds: 0.325 (asterisks), 0.4 (circles), 0.475 (diamonds), 0.55 (squares), 0.625 (plus signs), and 0.7 (triangles). Calculation: R' and R_N are median values over the respective measures calculated for the three cases. ASSMI is the median value over the respective measure calculated for the three cases as averages over the pairwise SSMI values. The mean and median SSMI values with each logical distance are obtained by first taking the median SSMI over the three cases for each node pair with that logical distance and then taking the mean and median over those node pair SSMI values.

the dependence of node coherence on logical distance. As expected, coherence is highest between neighbouring nodes and then drops as the logical distance grows. However, some nodes register clearly smaller coherence, because their mean values are smaller than the respective median values at logical distances one and two. Overall, as Figure 12.1 shows, at all logical distances, coherence is larger with the smaller discretisation thresholds.

12.2 Location Map and Graph Structure Estimates

This subsection studies identification of the network topology by the MGMN method at varying discretisation threshold values. MTN physical and logical topologies were already studied in Section 11.4. Here each estimated location map is obtained by first applying non-metric MDS to SSMI-based dissimilarity values and then scaling node coordinates with the Frobenius norm and finally Procrustes-transforming, without the scaling component, the map with respect to the

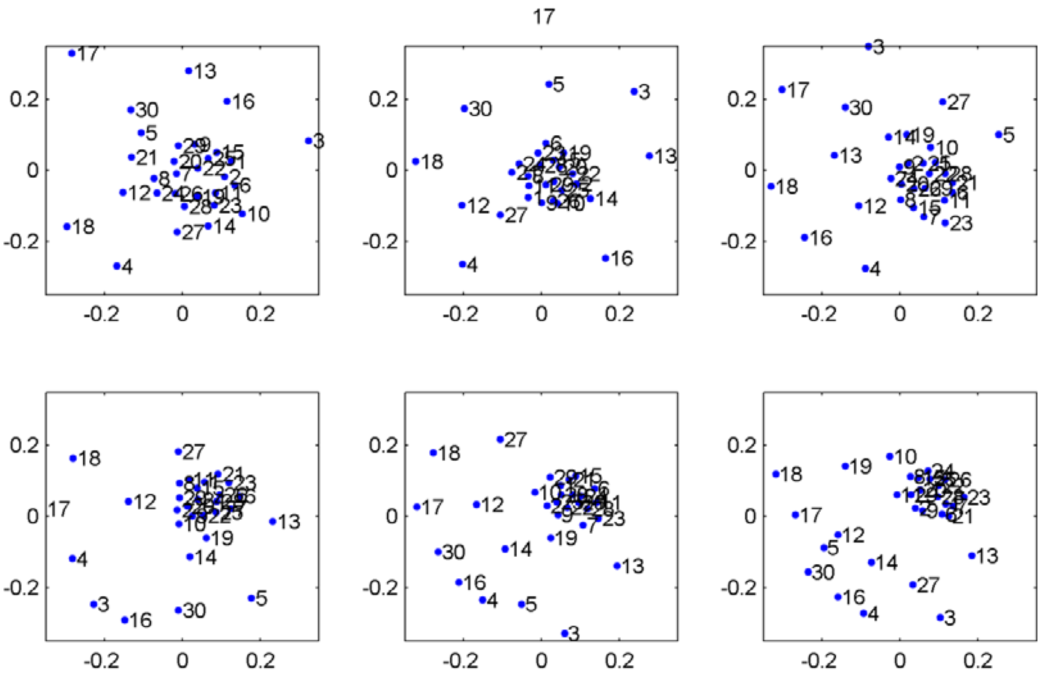

Figure 12.2. Estimated node location maps for a real network of 30 nodes. From top-left to bottom-right, the maps correspond to the following discretisation thresholds: 0.325, 0.4, 0.475, 0.55, 0.625, and 0.7. Calculation: each map shows a single randomly picked case, each Procrustes-transformed with respect to the MTN physical node location map.

MTN physical location map shown in Figure 11.4. The resulting node location maps appear in Figure 12.2 for the six discretisation thresholds studied in the previous section. Apparently, the larger the threshold value, the more the nodes are divided into two parts, into a node cluster consisting of tightly bound nodes and the remaining nodes loosely connected to any other nodes.

Throughout this chapter, again, to avoid local minima, MDS is always run 20 times from varying initial node coordinate values, similar to Chapter 8. After node location maps are obtained, their respective graph structures are finally produced using the MGMN method with the distance thresholding scheme (see Chapter 8). Here the threshold distance is chosen so that at each dis-

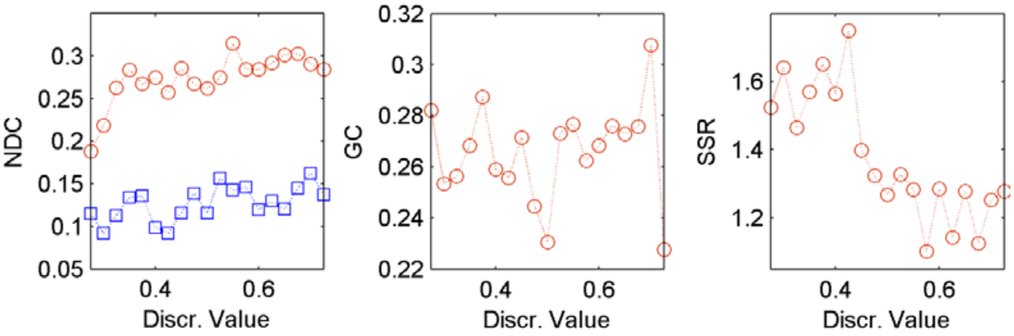

Figure 12.3. Similarity measures between estimated and MTN node location maps and their respective graph structures with a real network of 30 nodes. NDC, GC, and SSR are shown as functions of discretisation threshold. NDC is calculated with respect to MTN logical distances (circles) and MTN physical internode distances (squares). Calculation: with each discretisation threshold, measure values are median values over the respective measure values calculated with three cases.

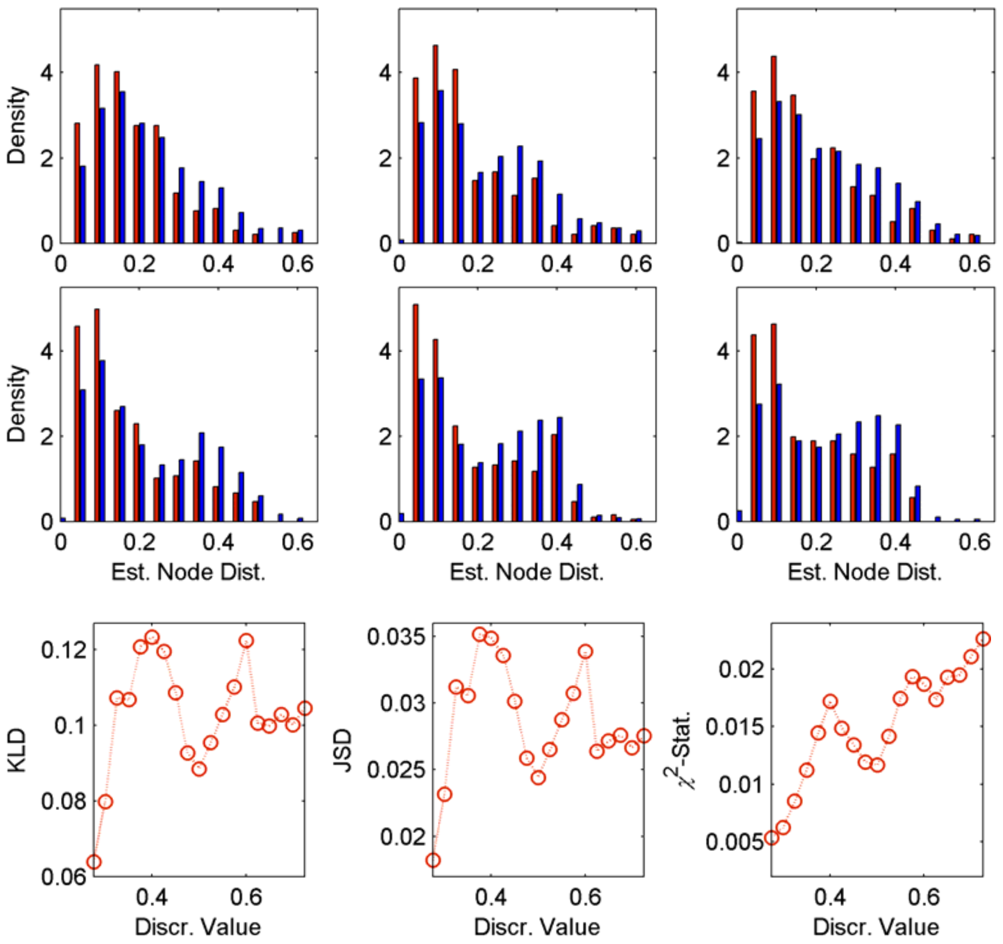

Figure 12.4. Distributions of internode distances on estimated node location maps of MTN logical neighbours (left-hand-side bars) and of all nodes (right-hand-side bars), and corresponding distribution similarity measures (bottom). Distribution plots from top-left to middle-right are shown for the following discretisation thresholds: 0.325, 0.4, 0.475, 0.55, 0.625, and 0.7. KLD (bottom-left), JSD (bottom-centre), and CSS approximation of KLD (bottom-right) are shown for all 19 discretisation thresholds. Calculation: with each discretisation threshold, histograms are calculated by using all the data of the three cases with bars defined at equal intervals. The histograms are then represented in the form of probability densities. With each discretisation and with each case, measures are calculated with the following bin widths in the histogram estimates: from 0.008 to 0.08 at constant intervals of 0.001. All histograms are estimated at a distance range from 0.008 to 0.8. With each discretisation, each measure is then averaged over the respective measure values obtained with each bin width, and the values are then further averaged over the three cases.

cretisation threshold, the average number of neighbours per node, A, is equal to 8.73, which is the average number of neighbours according to the MTN logical topology. Figure 12.3 shows the node distance (NDC) and graph correlation (GC) measures used in Chapter 8 for estimated maps and graphs. Because two pieces of topology information are available from the MTN, the NDC is calculated at each discretisation threshold with respect to both, integer-valued logical distances in the MTN logical topology and continuous-valued internode distances according to the MTN physical node location map. GC is calculated only with respect to MTN logical distances. As the Procrustes transformation is taken with respect to the MTN physical location map, the final SSR criterion describes the similarity between the estimated location map and the MTN physical location map.

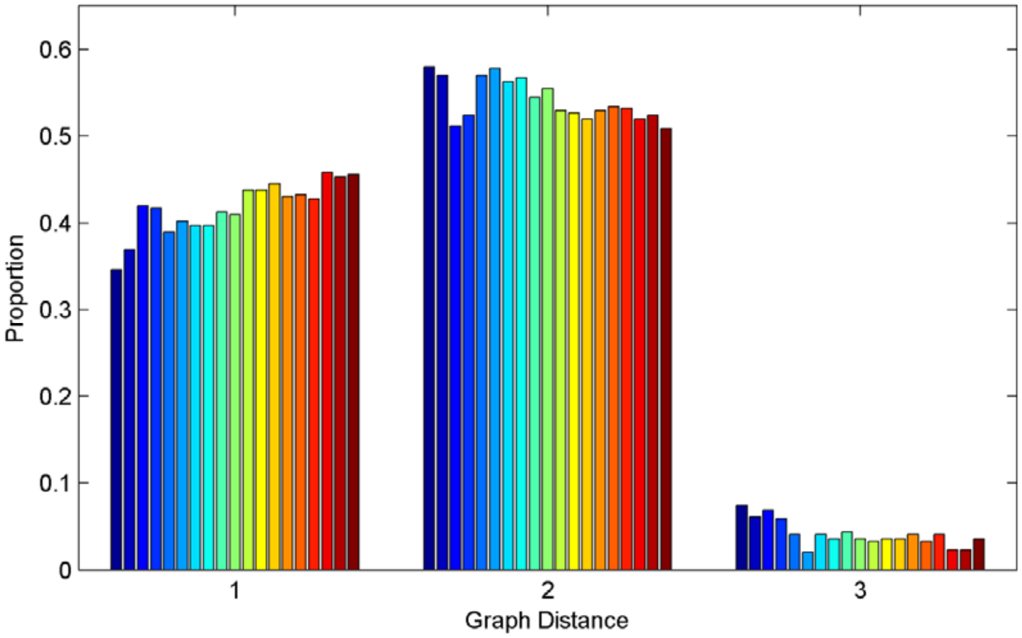

Figure 12.5. Histograms of MTN logical distances of estimated graph neighbours. Bars with each logical distance from left to right correspond to increasing discretisation threshold from 0.275 to 0.725 at even intervals of 0.025. Calculation: with each discretisation threshold, histograms are first calculated for all three cases, then the hits at each bar are summed over the hits in the three cases, and finally the number of hits at each bar is divided by the total number of hits in the three cases.

Correlation measures are shown in Figure 12.3 as functions of the discretisation threshold. As could be expected, the correlation values here are clearly smaller than in the synthetic network cases, where the largest NDC and GC values were well above 0.6. Here NDC lies near 0.3 when calculated with respect to the MTN logical topology, and GC fluctuates near 0.27. NDCs calculated with respect to the physical location map assume clearly smaller values than when calculated with respect to the MTN logical topology; hence estimated topologies seem similar to the MTN logical topology.

In Figure 12.1, according to R_N, the contribution of the Ising model interaction term is clearly larger than that of the loading term. This may explain why the effect of physical node locations does not show in the state data as strongly as that of logical relations, and why the NDC is clearly smaller when calculated with respect to MTN physical internode distances. Also the form of the MTN physical node location map, shown in Figure 11.4, may partly explain its small contribution with the map not being circular as typically produced by MDS in estimating logical and data-based maps. Altogether, the correlation measures are quite independent of the discretisation threshold, whereas according to the SSR criterion, the best node location map estimates are obtained with discretisation thresholds of roughly above 0.5.

Figure 12.4 shows distributions of internode distances on estimated location maps between logical neighbours according to the MTN logical topology and between all nodes. Calculations are here similar to those in Subsection 8.5.3, and results are again shown for the six discretisation thresholds. Because the distributions are difficult to compare visually, the KLD, JSD, and CSS approximation of the KLD, each introduced in Chapter 5, are calculated for each discretisation.

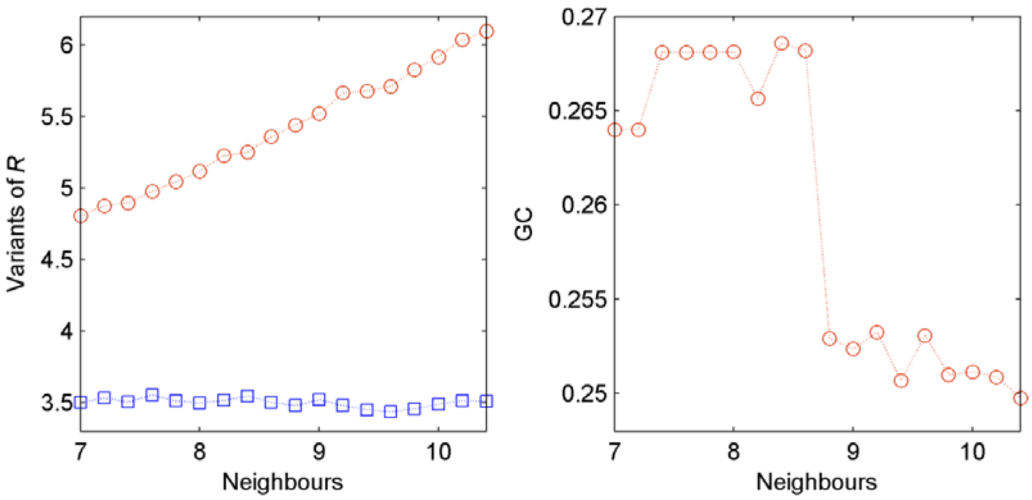

Figure 12.6. Effect of neighbourhood size on MGMN graph estimation. The left-hand plot shows R' (circles) and R_N (squares) as functions of A', and the right-hand plot shows ASSMI as a function of A'. Calculation: R' and R_N are median values over the respective measures calculated for three cases. ASSMI is the median value over the respective measure calculated for three cases as averages over pairwise SSMI values.

Because based on histogram estimates, these measures are sensitive to the number of bins in the histograms. Thus with each discretisation, several histograms with a varying number of bins are used, and final values are averaged over values obtained with the various bin numbers.

The measures are shown in Figure 12.4; the KLD and the JSD are very similar, whereas the CSS approximation of the KLD shows similarities also with the two other measures, and, additionally, a linearly increasing trend with an increasing discretisation threshold. In general, in two discretisation modes, at about 0.4 and 0.6, the two distributions show the closest similarity. The two humps were not observable in the distance correlation measures, and may have been caused by the properties of this particular data set. Yet the two humps can be clearly seen in each distribution similarity measure.

Figure 12.5 shows the distribution of MTN logical distances between estimated graph neighbours at all discretisation thresholds. The threshold distance increases in the bars from left to right, and the distributions are evaluated as in Subsection 8.5.3. For comparison, the proportions of node pairs assuming the three logical distances among all possible node pairs are 0.301, 0.561, and 0.138. For estimated graph neighbours, the proportions are clearly larger at small logical distances than the above among all possible node pairs. In shape, the bars at logical distance one in Figure 12.5 follow roughly the shape of the JSD measure in Figure 12.4. As could be expected, the proportion of nodes with node distance one in Figure 12.5 assumes on average smaller values than in the respective histograms with synthetic data, shown in Figure 8.7.

12.3 Effect of Neighbourhood Size

In Section 12.2, the threshold distance to define graph neighbours with the MGMN method was selected according to prior knowledge of A in the MTN logical topology. When prior information is not available, several distance thresholds can be tested and one chosen that leads to reasonable model predictions and small model parameter uncertainties. In this section, several

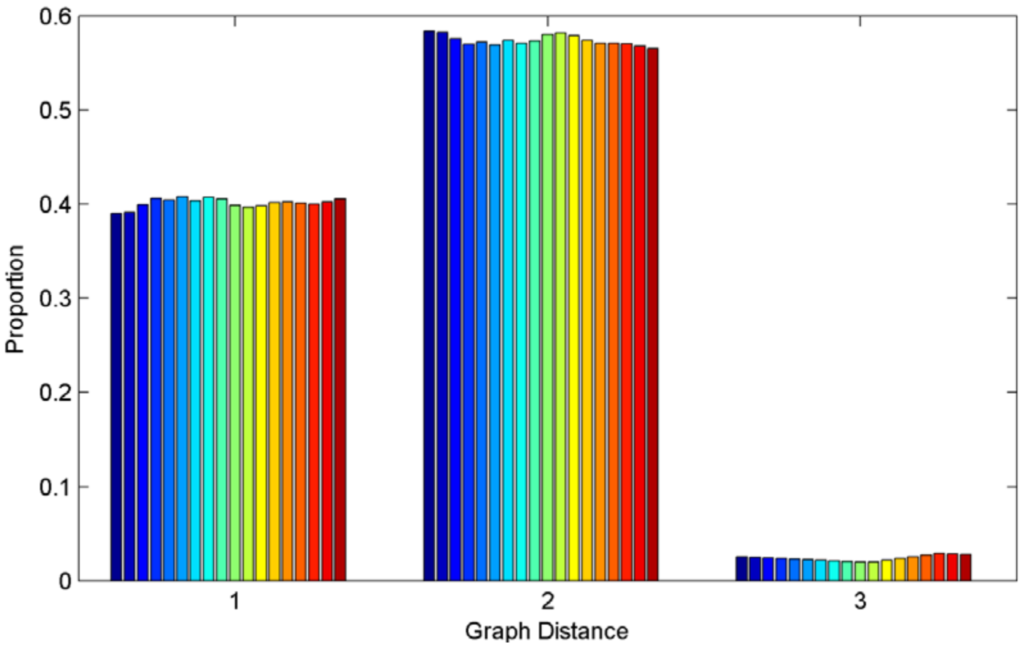

Figure 12.7. Histograms of MTN logical distances of estimated graph neighbours. Bars with each logical distance from left to right correspond to increasing A' from 7.0 to 10.4 at even intervals of 0.2. Calculation: with each A', histograms are first calculated for all three cases, then the hits at each bar are summed over the hits in the three cases, and finally the number of hits at each bar is divided by the total number of hits in the three cases.

threshold distances are tested to study the sensitivity of MGMN graph estimation and to ensure that use of prior information yields a good topology estimate. GC is applied here as a graph similarity measure, and distance threshold values are chosen such that on the estimated graph A, denoted here by A', ranges from 7.0 to 10.4 at intervals of 0.2. The discretisation threshold is chosen to be 0.4.

As the data and the true A are the same in each case, the ASSMI is not affected, but R' and R_N vary, because their calculation is based on estimated neighbourhoods. Figure 12.6 shows R', R_N, and GC as functions of A'. The term ratio R' increases linearly in A', as expected, while R_N is almost constant since J'/H' apparently compensates for the effect of changing neighbourhoods. According to the GC, best graph estimates are obtained between $A' = 7$ and $A' = 8.6$. Above $A' = 8.6$, the GC drops when A' is increased to 8.8, i.e., just around the prior-information-based A, where $A = 8.73$. According to Figure 12.3, with $A' = A = 8.73$, the GC is about 0.26, which lies in the middle of the two steps in Figure 12.6. Hence A appears as a sort of upper limit of a proper neighbourhood size, even though the GC values above it are only less than 10% smaller.

Figure 12.7 shows further the proportion of estimated node pairs at each MTN logical distance. At logical distance one, this proportion is nearly constant over the range of A'. The proportion of correct neighbours might have been expected larger with a small A', and then to decrease with an increasing A'. However, apparently the proportions of correct and false neighbours increase at the same rate when A' is increased. Later in Section 13.3, the effect of neighbourhood size is discussed in view of parameter estimation together with other methods for choosing the threshold distance correctly.

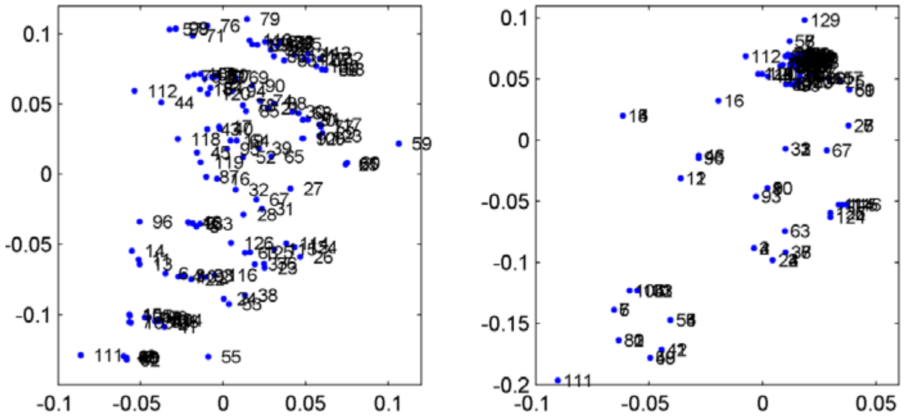

Figure 12.8. Logical (left) and physical (right) location map presentations of 132 BTS nodes. Calculation: the physical location map shows the physical node locations scaled with the respective Frobenius norm. The logical location map is obtained by applying non-metric MDS to logical distances, calculated for the nodes from logical neighbourhood relations. Then the map is further scaled with the respective Frobenius norm and Procrustes-transformed with respect to the physical location map.

12.4 Effect of Network Size

In this section, topology identification by the MGMN method is tested with a network of 132 nodes, having $A = 11.76$, at the same discretisation thresholds as above. The 30-node network studied earlier is a subnetwork of this larger network. In the GSM MTN, all 132 nodes belong to one BSC. Because of excessive computation time with the larger network, only the SSCSS, an approximation of the SSMI, is applied here as a node dependency measure. Because computation of the SSCSS requires no generation of synthetic data, it is much faster than that of the SSMI. In addition, because calculating the SSCSS is deterministic, calculations are not repeated thrice. However, as above, MDS is still repeated 20 times to reduce the stochastic effects related to the calculation of location map estimates.

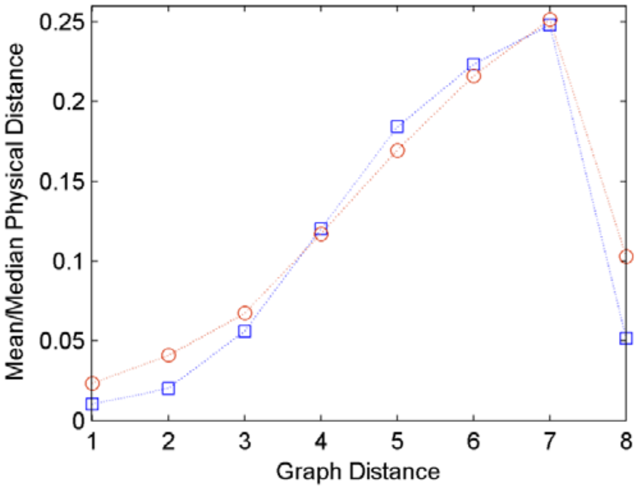

Figure 12.9. Comparison of physical and logical internode distances. Mean (circles) and median (squares) values of physical internode distances according to the physical node location map are shown as a function of corresponding internode logical distances according to the MTN logical topology. Calculation: for each node pair with a certain logical distance, mean and median values are taken over the physical (Euclidean) distances between the nodes.

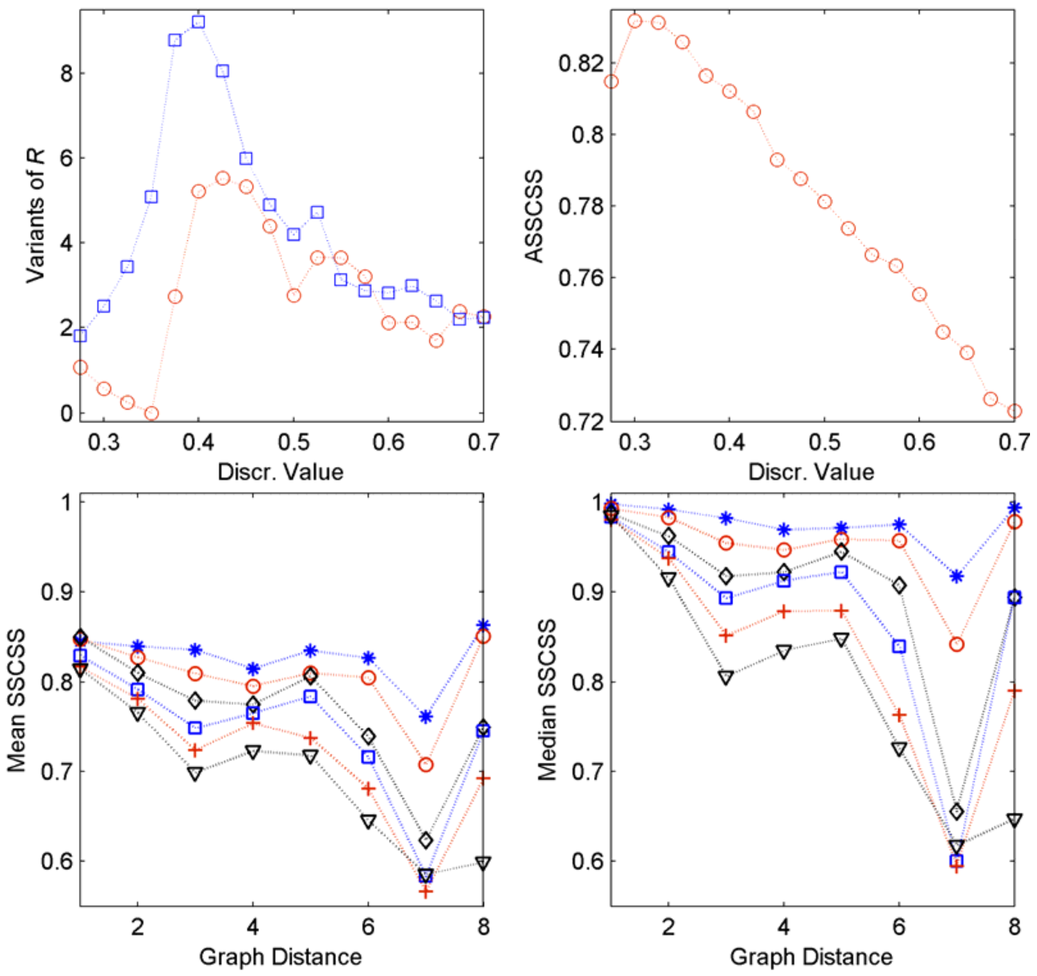

Figure 12.10. Network coherence with a real network of 132 nodes. Top-left plot shows R' (circles) and R_N (squares), and the top-right plot ASSCSS, as functions of discretisation threshold. Mean (bottom-left) and median (bottom-right) values of SSCSS are shown as functions of the logical (graph) distance of nodes with discretisation thresholds 0.325 (asterisks), 0.4 (circles), 0.475 (diamonds), 0.55 (squares), 0.625 (plus signs), and 0.7 (triangles). Calculation: R', R_N and ASSCSS are all calculated from a single case with the ASSCSS as the average over pairwise SSCSS values. Mean and median SSCSS values are obtained by taking, respectively, the mean and the median over the node pair SSCSS values with each logical distance.

The problem in testing the methods with a large MTN is that the number of data observations cannot be increased as with synthetic networks in Chapter 8. It was concluded in Chapter 8 that the number of observations should be increased at least linearly in the number of network nodes, and that with a synthetic network of 120 nodes the data with $L = 270$ was too small for accurate results. In any case, only 270 network observations are available here, and after some are further removed because of missing data values in many nodes, the data finally amounts to 260 observations. The data is preprocessed here like with the smaller network.

The location map, obtained when non-metric MDS is applied to the MTN logical topology, and the respective true geography of nodes are shown in Figure 12.8. The previous 30-node subset is part of the main cluster on the physical location map. Most nodes are physically located near each other and form a cluster, whereas the remainder form a few small clusters outside the main cluster. Because the two location maps are difficult to compare visually, and though they seem to

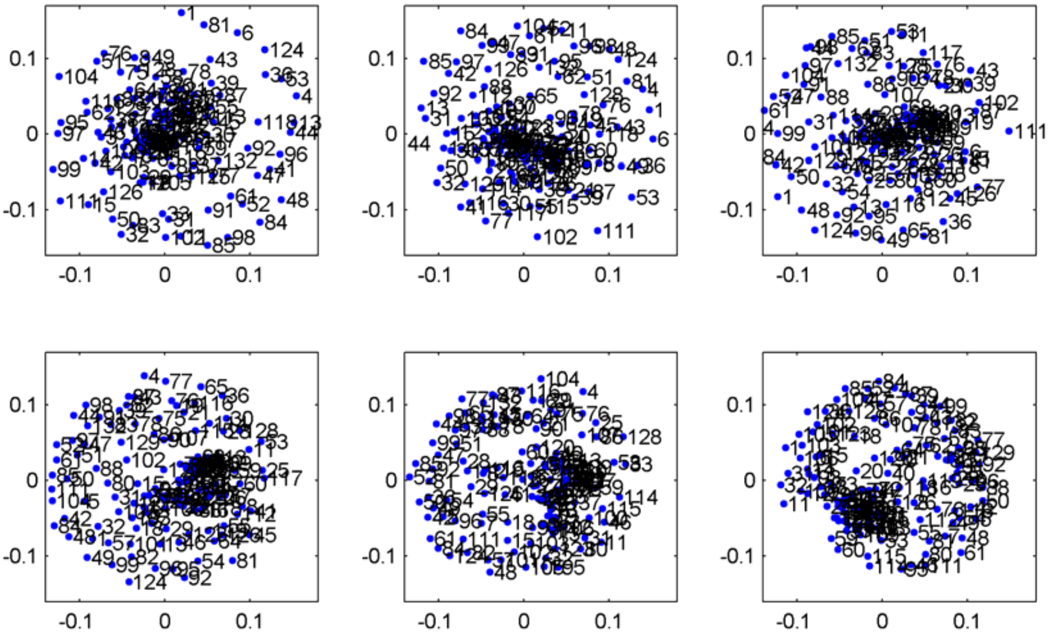

Figure 12.11. Estimated node location maps for a real network of 132 nodes. From top-left to bottom-right, maps correspond to the following discretisation thresholds: 0.325, 0.4, 0.475, 0.55, 0.625, and 0.7. Calculation: each map presents a single case, each Procrustes-transformed with respect to the MTN physical node location map.

assume a somewhat similar form, Figure 12.9 shows mean and median internode distances on the physical location map at each logical distance, i.e., graph distance according to the MTN logical topology. The physical distance increases monotonically as the logical distance is increased, but drops at the largest logical distance, at which there are only a few nodes. Overall, the two location maps appear to be similar.

Network coherence is studied in Figure 12.10, which shows R and the ASSCSS (analogous to ASSMI) as functions of the discretisation threshold. Like the ASSMI with the smaller, 30-node network in Section 12.1, the ASSCSS degrades nearly linearly as the discretisation threshold increases. Both R' and R_N peak near the discretisation threshold 0.4, after which the two measures decrease as the discretisation threshold increases. Apparently, the denominator in Eq. (8.7) is very small near the discretisation threshold 0.4. Figure 12.10 also shows mean and median SSCSS values as functions of the MTN logical distance with the following six discretisations thresholds: 0.325, 0.4, 0.475, 0.55, 0.625, and 0.7. The coherence decreases as the logical distance increases, the decrease becoming steeper at a larger discretisation threshold. At the largest logical distance, the number of data points is small; hence SSCSS values are highly uncertain. Estimated node location maps at the six discretisation thresholds are shown in Figure 12.11. The maps are difficult to compare visually, but at least each map consists of a large node cluster and some loosely connected nodes in accordance with the logical and physical location maps.

Figure 12.12 shows node distance and graph correlation values as functions of the discretisation threshold. The NDC calculated with respect to both physical and logical distances behaves similarly. The NDC and GC both are rather small except at about discretisation threshold 0.6, at which the correlation suddenly increases. At this particular threshold, the correlation is close to

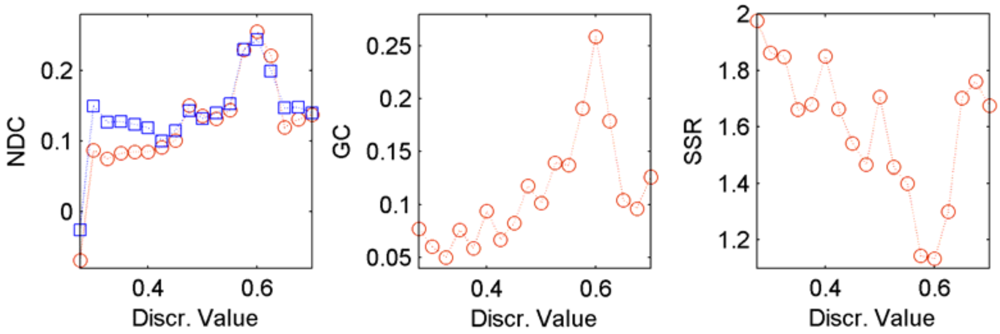

Figure 12.12. Similarity measures between estimated and MTN node location maps and the respective graph structures with a real network of 132 nodes. NDC, GC, and SSR are shown as functions of discretisation threshold. NDC is calculated with respect to MTN logical distances (circles) and MTN physical internode distances (squares). Calculation: with each discretisation threshold, measure values are calculated from a single case.

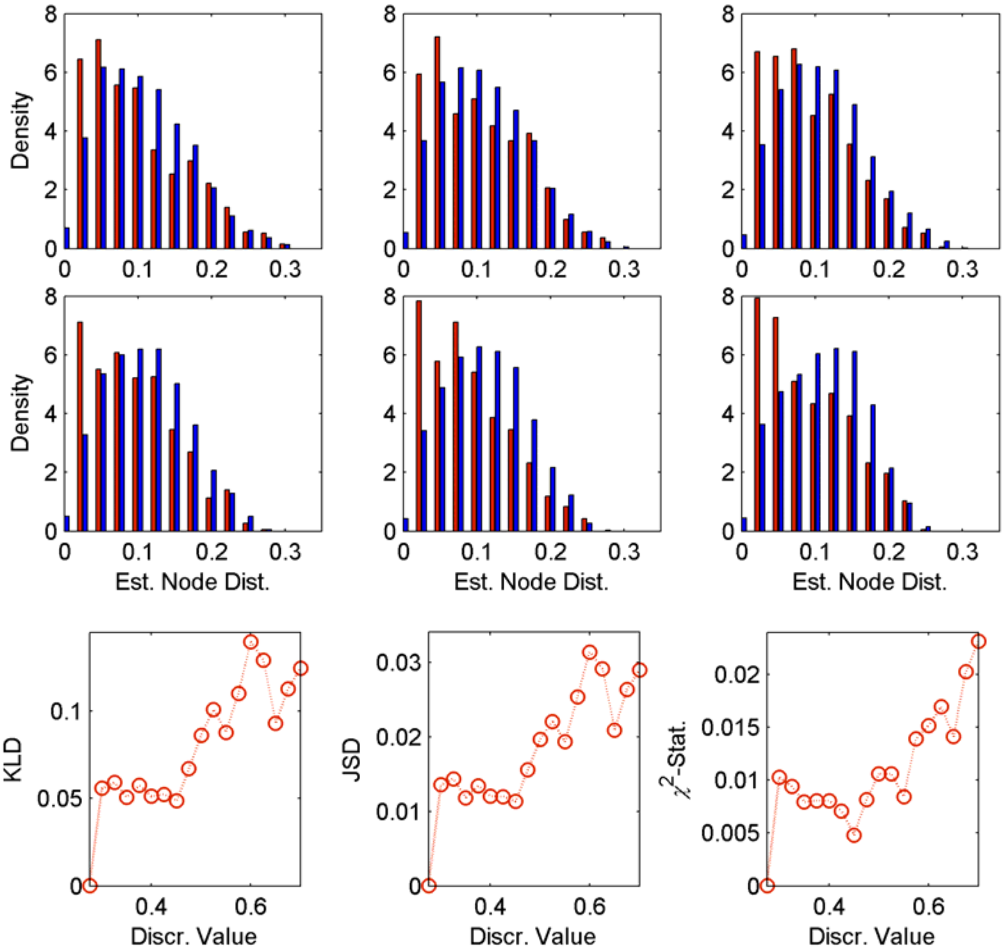

Figure 12.13. Distributions of internode distances on estimated node location maps of MTN logical neighbours (left-hand-side bars) and of all nodes (right-hand-side bars), and the corresponding distribution similarity measures (bottom). Distribution plots from top-left to middle-right are shown for the following discretisation thresholds: 0.325, 0.4, 0.475, 0.55, 0.625, and 0.7. KLD (bottom-left), JSD (bottom-centre), and CSS approximation of KLD (bottom-right) are shown for all 19 discretisation thresholds. Calculation: at each discretisation threshold, histograms are calculated from a single case, and represented in the form of probability densities. At each discretisation measures are calculated with the following bin widths in the histogram estimates: from 0.005 to 0.05 at constant intervals of 0.001. All histograms are estimated in a distance range from 0 to 0.35. At each discretisation, each measure is then averaged over the respective measure values obtained at each bin width with the single case used.

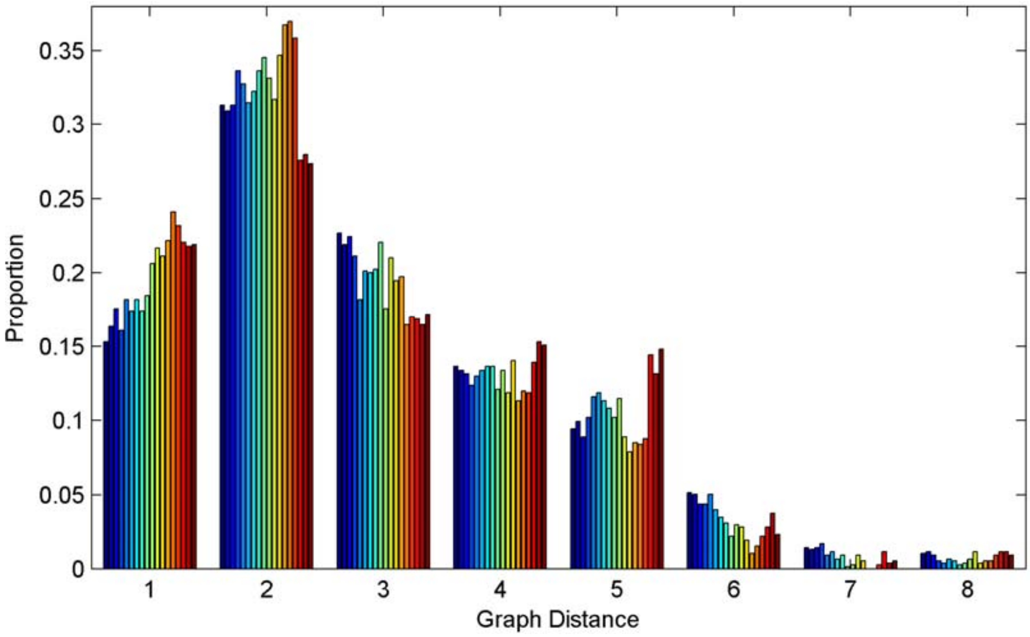

Figure 12.14. Histograms of MTN logical distances of estimated graph neighbours. Bars at each logical distance from left to right correspond to increasing discretisation thresholds from 0.275 to 0.725 at even intervals of 0.025. Calculation: at each discretisation threshold, histograms are calculated from a single case.

the values obtained with the smaller, 30-node network, though in the smaller network correlation appears in a wider range of discretisation thresholds. In addition, the SSR decreases at about 0.6, at which it shows values similar to those of the smaller network. Hence, the success of topology identification by the MGMN method with the larger network seems sensitive specially to the choice of discretisation threshold. However, if the threshold is chosen appropriately, results are almost as good as those obtained with the smaller network. This is somewhat unexpected, because the data available for the large network was deemed inadequately small. Furthermore, the best topology estimates for both the large and small network are obtained at the same discretisation threshold, 0.6, though the small network produced good results also at 0.4.

Figure 12.13 shows distributions of node distances on estimated location maps between MTN logical neighbours and all nodes. The histograms are calculated as in Section 12.2, and results are shown for the six discretisation thresholds. Compared visually, the two distributions seem dissimilar at large discretisation thresholds. To verify this, the KLD, JSD, and CSS approximation of the KLD are again calculated. To avoid measure sensitivity to the number of bins in the histogram, each measure is calculated with several histograms corresponding to a various number of bins and then averaged. Figure 12.13 shows that the three measures again behave similarly. In particular, the KLD and JSD both have a peak at discretisation threshold 0.6. Like in the 30-node network, the CSS is otherwise nearly similar to the other two measures but increases also between discretisation thresholds 0.6 and 0.7. We conclude that the topology correlation measures and distribution similarity measures give similar results.

Figure 12.14 shows distributions of MTN logical distances of estimated graph neighbours at all the analysed discretisation thresholds. The proportions of nodes at each logical distance among all possible node pairs are as follows: 0.090, 0.229, 0.258, 0.182, 0.124, 0.075, 0.026, and 0.015. Hence estimated neighbourhoods clearly assume smaller logical distance values. The peaks at logical distances one and two occur also at discretisation threshold 0.6. The curve shown in bars at logical distance one is similar to the curves formed by the previous distribution similarity measures in Figure 12.13.

13. Parameter Identification for MTNs

This chapter focuses on the parameter identification of MRF models for MTNs with the model graph structures defined by topologies identified by the MGMN method in Chapter 12. To evaluate parameter estimation independently of topology estimation, also the MTN logical topology is analysed by assuming that it is the true MRF model graph structure. The Ising model is applied again as the MRF model, and parameter identification is tested with the same data discretisation thresholds as in Chapter 12 and with the data discussed in Chapter 11. All calculations are repeated three times at each discretisation threshold to reduce stochastic effects related to MRF model parameter estimation. Model identification is also examined with varying node neighbourhood sizes and with the larger network used in Chapter 12.

Section 13.1 deals with Ising model parameter estimates and parameter uncertainty obtained at the varying data discretisation thresholds for the 30-node network. Node state probability predictions made with the identified models are then discussed in Section 13.2 as methods for evaluating the entire MRF model identification scheme. Section 13.3 continues the analysis started in Section 12.3 and considers the effect of the threshold distance in defining graph structures for model parameter identification. Section 13.3 also studies the uncertainty of parameter estimates to define an appropriate threshold distance value for graph structure. Similarly, Section 13.4 continues the analysis in Section 12.4 by studying model identification for the large, 132-node network.

13.1 Parameter Estimates and Uncertainties

Ising model parameters (J, H, h_0) are estimated by the pseudolikelihood method already applied in Chapter 9. Parameter identification is studied at all discretisation thresholds ranging from 0.275 to 0.725 at 0.025 intervals. At each discretisation threshold, the corresponding topology estimate in Section 12.2 is used for graph structure in the Ising model. However, the MTN logical topology is chosen as an alternative Ising model graph structure, and considered as a reference topology. Estimated parameter values based on both the estimated graph structure and the MTN logical topology are shown in Figure 13.1 as functions of the discretisation threshold.

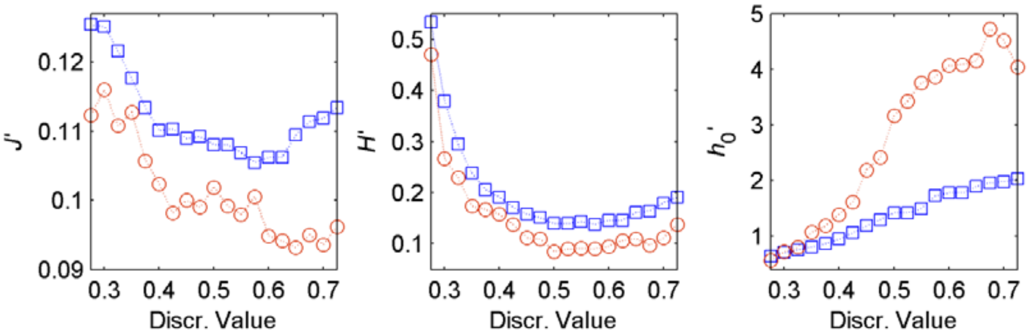

Figure 13.1. Ising model parameter estimates. Estimates J' (left), H' (centre), and h_0' (right) are shown as functions of discretisation threshold. The results are shown with the MTN logical topology (squares) and with the estimated topology (circles). Calculation: parameter values are medians over the respective values with the three cases.

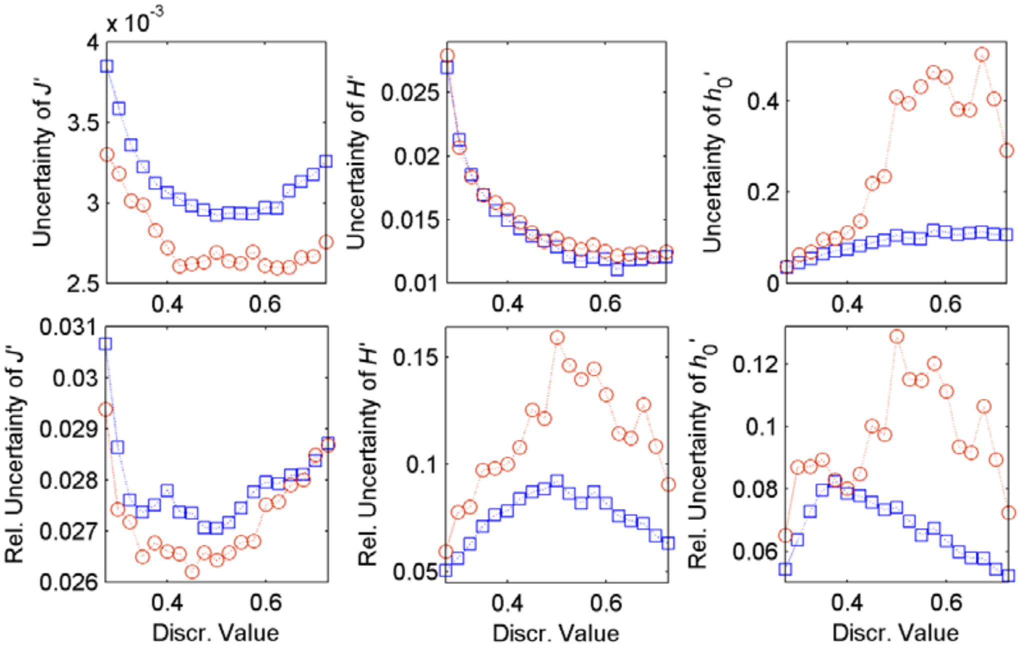

Figure 13.2. Uncertainties (top row) and relative uncertainties (bottom row) of estimated Ising model parameters. Uncertainties of J' (left column), H' (centre column), and h_0' (right column) are shown as functions of discretisation threshold. Results are shown with the MTN logical topology (squares) and with the estimated topology (circles). Calculation: uncertainty values are medians over the respective values in three cases.

It is difficult to evaluate parameter estimates when true parameter values are unknown. At least, no drastic changes or large fluctuations seem to occur in the parameter estimates at any discretisation threshold. H' behaves smoothly and assumes values similar between the estimated graph structure and the MTN logical topology. Apparently, H' is not very sensitive to the graph structure, because it is related to the external load term in the Ising model, whereas the neighbourhoods affect directly the interaction term. Estimates J' and h_0' are more sensitive to the graph structure, and particularly at the largest discretisation thresholds, the difference grows in the estimates between the two topologies.

In particular, changing the discretisation threshold, i.e., the proportion of node observations in state -1, directly affects h_0', because h_0 defines the threshold for loading values below which nodes favour states -1 and above which states $+1$ are favoured, respectively. Since a kind of reverse relation exists between J and h_0, discussed in Subsection 9.4.1, also J' is affected. Overall, at small discretisation thresholds, parameter estimates differ slightly for each parameter between the estimated and reference topology; however, at large discretisation thresholds, results differ markedly. Consequently, the latter difference suggests that also the physical topology affects MRF model behaviour, as previously discussed in Chapters 2 and 11. Therefore, the MTN logical topology should not be considered here as the true topology for MRF modelling.

Figure 13.2 shows uncertainties and relative uncertainties of parameter estimates, derived through Gaussian distribution approximations, explained in chapter 9. The uncertainty of H' is almost identical to the estimated graph and the MTN logical topology. In summary, the parameter uncertainties follow almost the respective parameter estimate values in Figure 13.1. Hence the

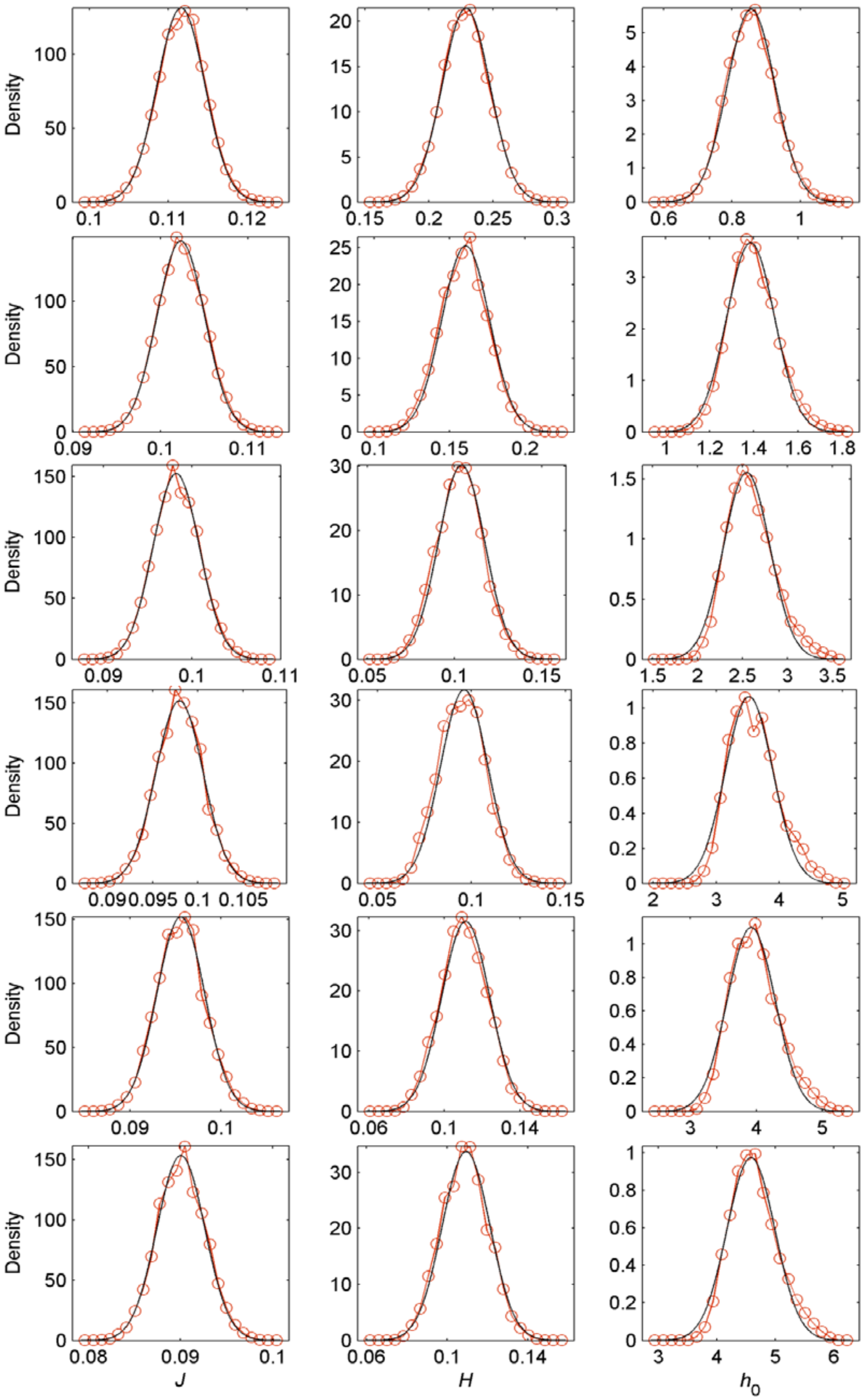

Figure 13.3. True marginal probability distributions of Ising model parameters (circles) and their Gaussian approximations (solid curves), when estimated graph structures are used. Distributions are shown for J (left column), H (centre column), and h_0 (right column), and from top to bottom correspond to the following six discretisation thresholds: 0.325, 0.4, 0.475, 0.55, 0.625, and 0.7. Calculation: in each case, distributions are shown only for a single randomly picked ensemble.

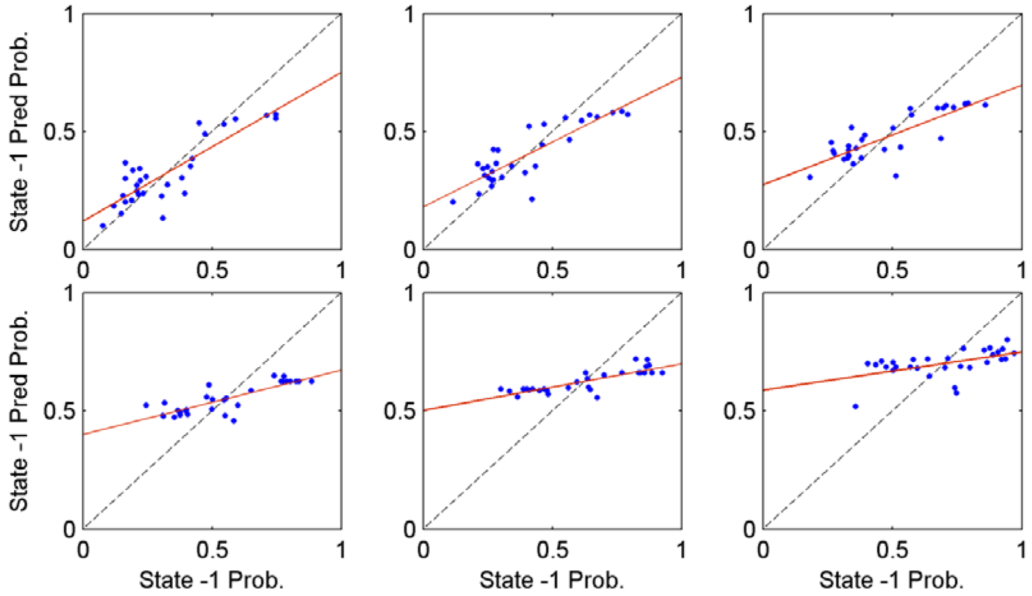

Figure 13.4. Predictions with estimated (structure and parameters) Ising models. Node state -1 probability predictions are shown for each node as a function of data-calculated node state -1 probabilities. Predictions are shown in dots, linear regression lines fitted to predictions in solid lines, and reference curves of optimal predictions in dashed lines. Predictions from top-left to bottom-right correspond to the following six discretisation thresholds: 0.325, 0.4, 0.475, 0.55, 0.625, and 0.7. Calculation: among the three cases, shown is the one that corresponds to the minimum average node state -1 prediction error.

changes in the relative uncertainties are quite small. With the estimated topology, the relative uncertainty for H' and $h_0{}'$ is at its highest between 0.5 and 0.6, and for J' at its lowest at discretisation threshold 0.5.

To justify the use of standard deviations of marginal Gaussian distributions as parameter uncertainties, marginal Gaussians and the respective marginal parameter probability distributions derived from the true joint probability distribution are shown in Figure 13.3 with estimated graph structures for the six discretisation thresholds. Though the true marginal distributions are slightly skewed at large discretisation thresholds, the marginal Gaussian distributions seem to approximate the true marginal distributions well enough to be used in uncertainty analysis.

13.2 Model Predictions

The logical and physical topologies of the MTN considered here are known, but in a general case its true network topology may be uncertain, unknown, or known only partially through some information. Here we have two, mostly overlapping but also somewhat different, pieces of topology information. Though an estimated topology can be compared to these topologies, we should consider other topology evaluation methods as well, because we cannot be certain as to which of the two topologies the estimated one should most resemble. Furthermore, because true MRF model parameters are never known for real networks, estimated parameters must somehow be evaluated.

Unfortunately, good methods are not available for evaluating topologies and parameters. Here the full conditionals of the Ising model are applied to predict node states with the same data set

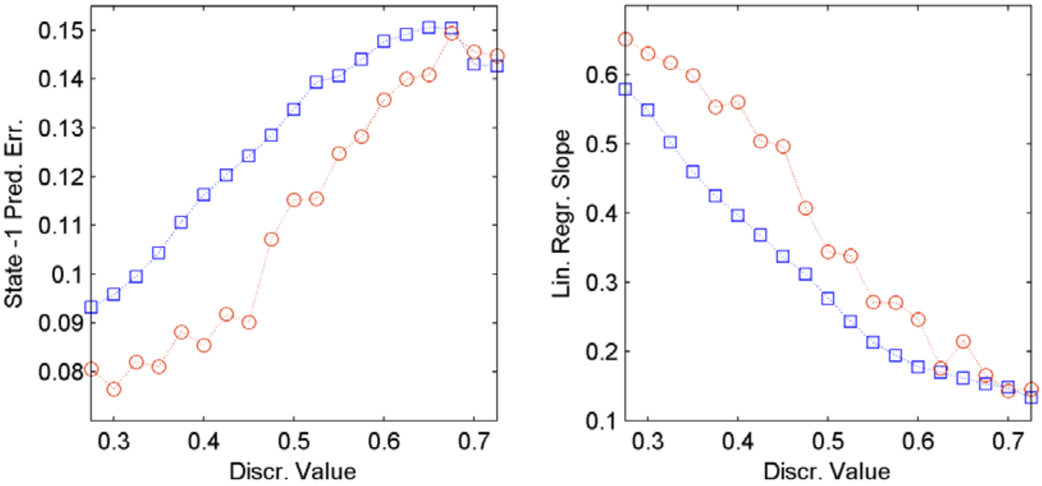

Figure 13.5. Node state -1 absolute predictions errors (left) and slopes of fitted linear regression lines (right) as functions of discretisation value. Results are shown for the MTN-logical-topology-based (squares) and estimated-graph-structure-based Ising models (circles). Calculation: absolute prediction errors are medians over three cases, for which each error is calculated as an average over all nodes. Slope coefficients are medians of the respective coefficients obtained in each case.

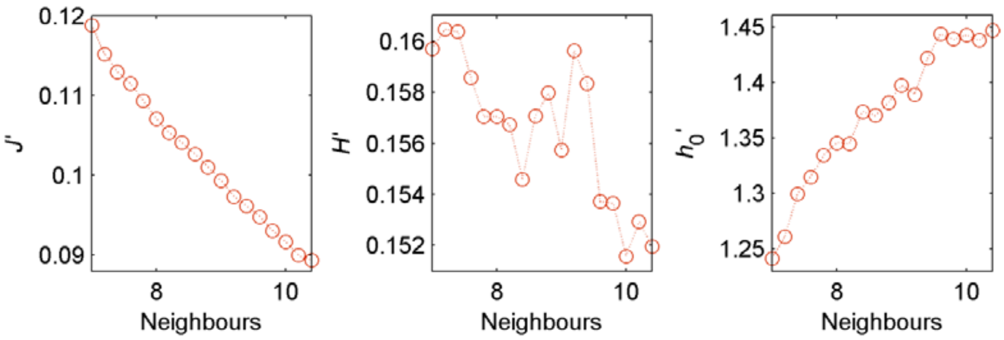

Figure 13.6. Ising model parameter estimates. Estimates J' (left), H' (centre), and $h_0{}'$ (right) are shown as functions of A'. Results are obtained with the estimated topology. Calculation: parameter values are medians over the respective values in three cases.

already used in the model identification phase. Due to the limited number of observations available, no validation data set can be separated from the overall data. The node state probability predictions applied here are based on the cross-validation scheme explained in detail in Chapter 9. In addition, state probability predictions are here compared to their respective state probabilities calculated directly from the data for each network node.

Figure 13.4 shows identified model-based probability predictions for node state -1 as functions of respective data-calculated probabilities for all the 30 nodes at the six discretisation thresholds. Linear regression lines are shown fitted to the prediction data. Evidently, predictions are at their best at small discretisation thresholds, whereas the nodes' characteristic features cannot be predicted at large discretisation thresholds.

Figure 13.5 shows absolute errors in state -1 probability predictions and slopes of fitted linear regression lines for models based on both estimated graph structures and the MTN logical topology. Clearly, prediction errors are smallest and slopes largest at the smallest discretisation

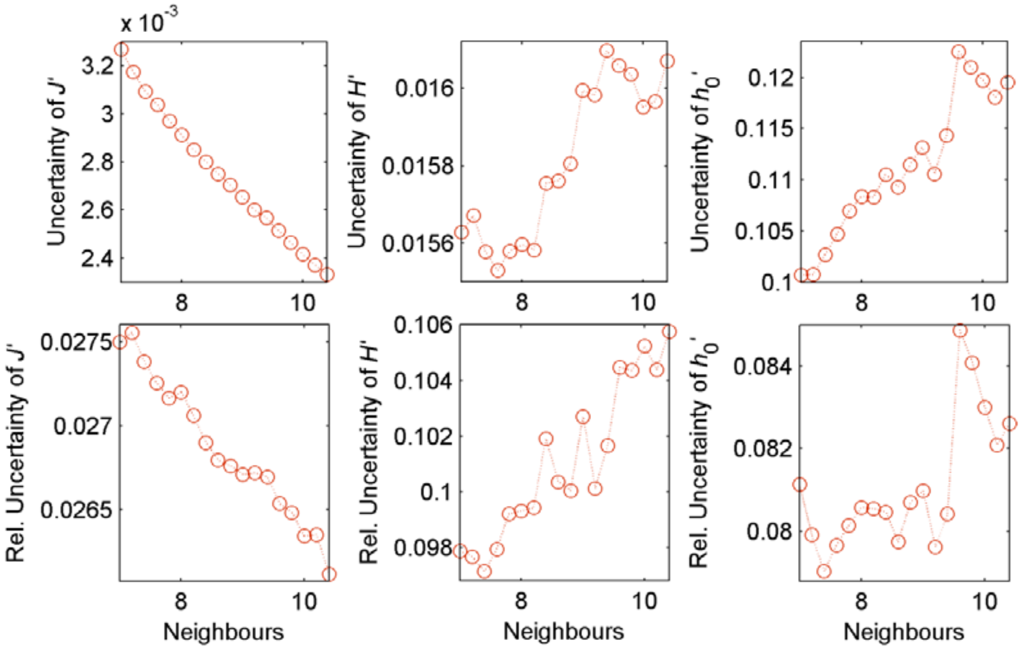

Figure 13.7. Uncertainties (top row) and relative uncertainties (bottom row) of estimated Ising model parameters. Uncertainties of J' (left column), H' (centre column), and $h_0{}'$ (right column) are shown as functions of A'. Results are obtained with the estimated topology. Calculation: uncertainty values are medians over the respective values in three cases.

thresholds. Hence best predictions are obviously obtained at the smallest discretisation thresholds. Interestingly, the models based on estimated graph structures constantly yield better predictions in the whole range of discretisation thresholds than those based on the MTN logical topology. Therefore, data-based graph structure estimates seem to be capable of capturing more information about node interactions than logical relations can do alone, presumably, a joint impact of the logical and physical MTN topologies.

13.3 Effect of Neighbourhood Size

The sensitivity of the topology identification results to the choice of the threshold distance value, which defines node neighbourhoods, was tested in Section 12.3. This section continues the analysis by studying how the respective parameter identification results are affected by the chosen neighbourhood size. As in Section 12.3, the average neighbourhood size on the estimated graph, again denoted by A', is varied from 7 to 10.4 at intervals of 0.2, and with each A', the corresponding graph structure estimate in Section 12.3 is applied for graph structure in the Ising model. Parameter estimates are shown in Figure 13.6 as functions of A'. The estimate J' decreases with the increasing A', apparently because the interaction term of the Ising model is additive in A'; thus the increasing A' is compensated for by the decreasing J'. The ratio J'/A' is almost constant with all the neighbourhood sizes analysed. Because J and h_0 have the same magnitude effect, though opposite in sign, on model coherence, changes in h_0' are opposite to those in J'. Estimate H' is a more random function of neighbourhood size than the two other parameters.

Figure 13.7 shows the uncertainty and relative uncertainty of the estimated parameters. According to both figures, the uncertainty of J' decreases as A' increases, whereas the uncertainties of

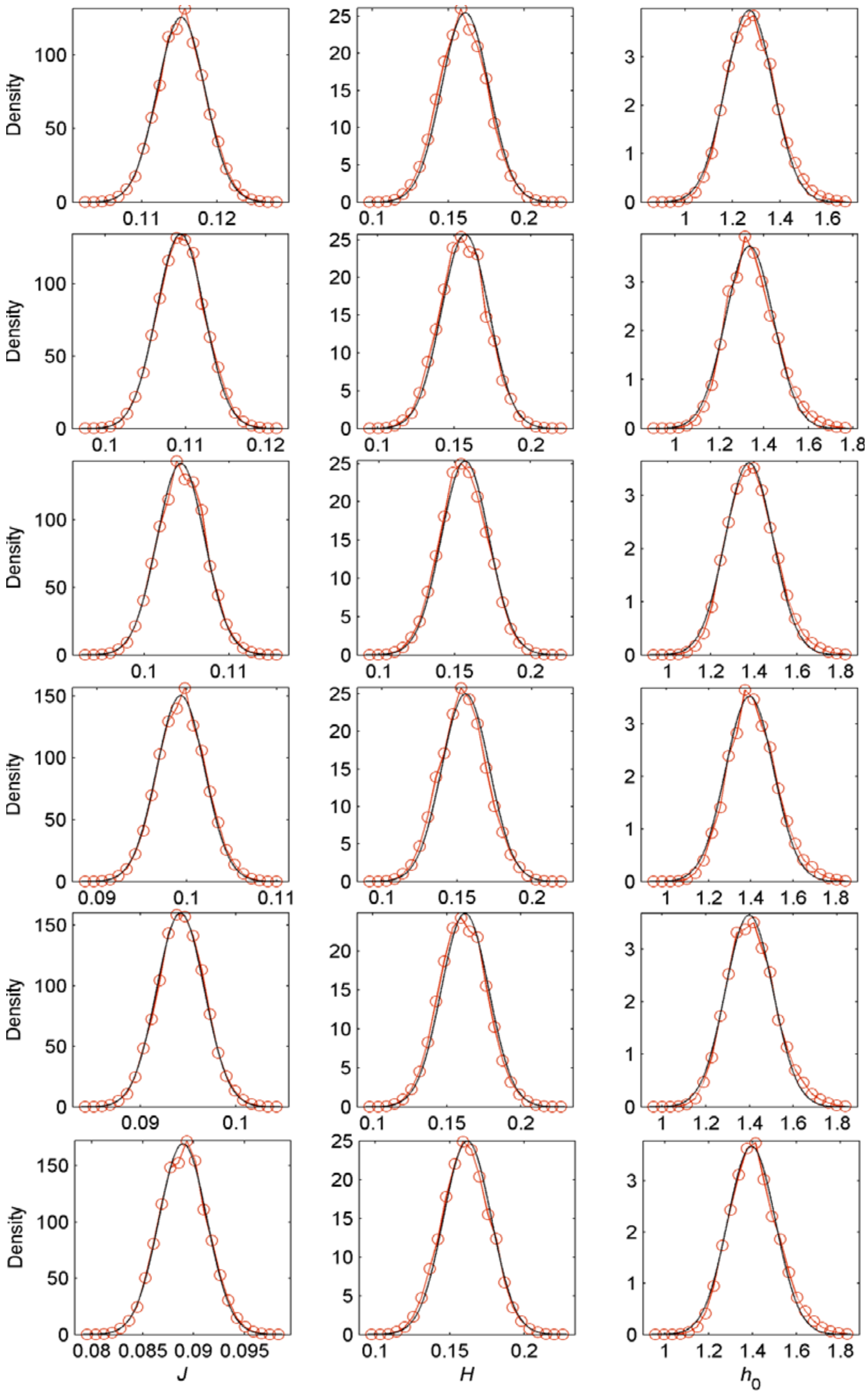

Figure 13.8. True marginal probability distributions of Ising model parameters (circles) and their Gaussian approximations (solid curves), when estimated graph structures are used. Distributions are shown for J (left column), H (centre column), and h_0 (right column), and from top to bottom correspond to the following six A' values: 7.2, 7.8, 8.4, 9.0, 9.6, and 10.2. Calculation: in each case, distributions are shown only for a single randomly picked ensemble.

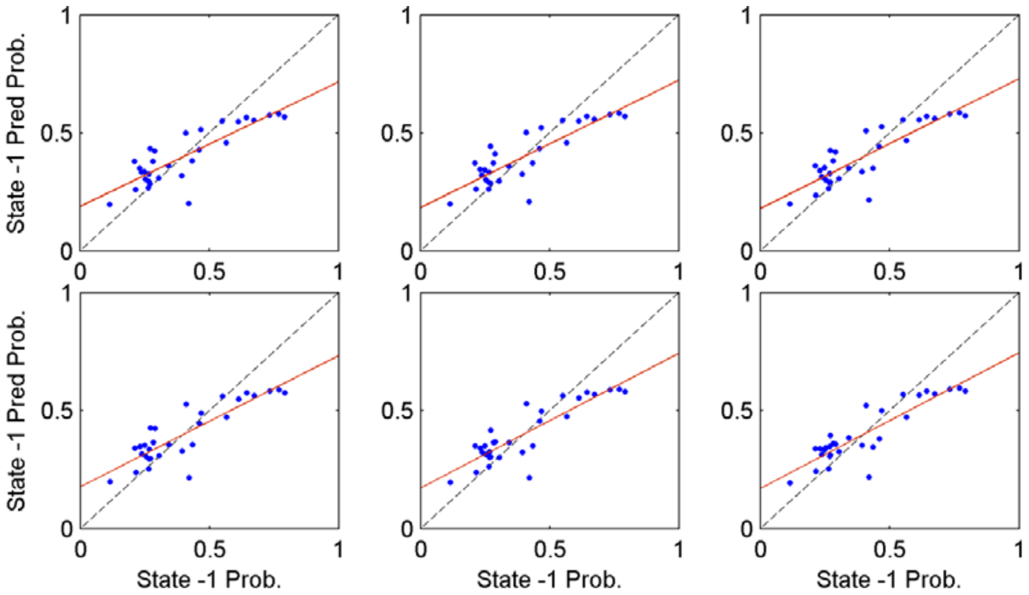

Figure 13.9. Predictions with estimated (structure and parameters) Ising models. Node state -1 probability predictions are shown for each node as a function of data-calculated node state -1 probabilities. Predictions are shown in dots, linear regression lines fitted to predictions in solid lines, and reference curves of optimal predictions in dashed lines. Predictions from top-left to bottom-right correspond to the following six A' values: 7.2, 7.8, 8.4, 9.0, 9.6, and 10.2. Calculation: among the three cases, shown is the one that corresponds to the minimum average node state -1 prediction error.

the other two parameters increase. Parameter uncertainty can also be used to choose an appropriate distance threshold to define the graph, if no prior information is available about the true A. Hence minimising the overall uncertainty of the parameters, with the weight on J relating to the model's interaction term, may lead to a reasonable graph structure. Here the uncertainty of all three parameters is relatively small around at $A' = 8.73$, which is the average neighbourhood size according to the MTN logical topology. Figure 13.8 shows marginal parameter distributions and the respective marginal Gaussian approximations at the following values of A': 7.2, 7.8, 8.4, 9.0, 9.6, and 10.2. Despite some minor deviations, the Gaussian approximations are again good.

Figure 13.9 presents the state probability prediction results with the identified models for six values of A'. Overall, changes in the prediction results are quite small. However, Figure 13.10, showing prediction errors and slopes of linear regression lines as functions of A', is more informative. Though the numerical values change minimally, prediction errors evidently become smaller and the slopes larger as A' increases, indicating better predictions with large neighbourhood sizes. In conclusion, neither topology nor model parameter estimates are particularly sensitive to A'. As already noted in Section 12.3, when A' is increased, correct neighbours increase in number at the same rate as incorrect ones, thus affecting results only slightly.

13.4 Effect of Network Size

In Section 12.4, a network of 132 BTS cell nodes and with 260 observations was analysed to demonstrate how the MGMN topology identification method works with large MTNs. Here the parameter estimation in the model identification is analysed for this network by using the corresponding estimated topologies for graph structures in the Ising model. Figure 13.11 shows esti-

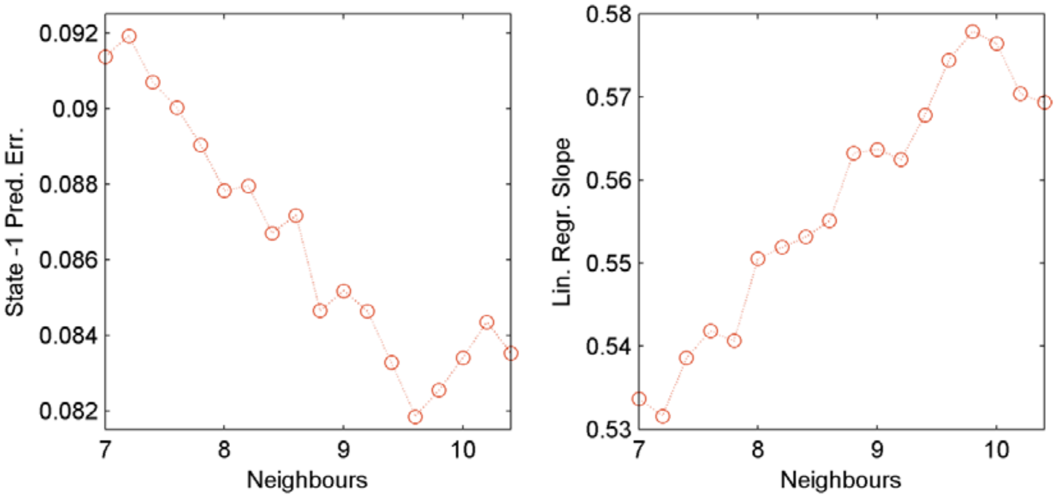

Figure 13.10. Node state -1 absolute predictions errors (left) and the slopes of the fitted linear regression lines (right) as functions of A'. The results are with the estimated Ising models. Calculation: absolute prediction errors are medians over the three cases, for which each the error is calculated as average over all the nodes. The slope coefficients are medians of the respective coefficients obtained with each case.

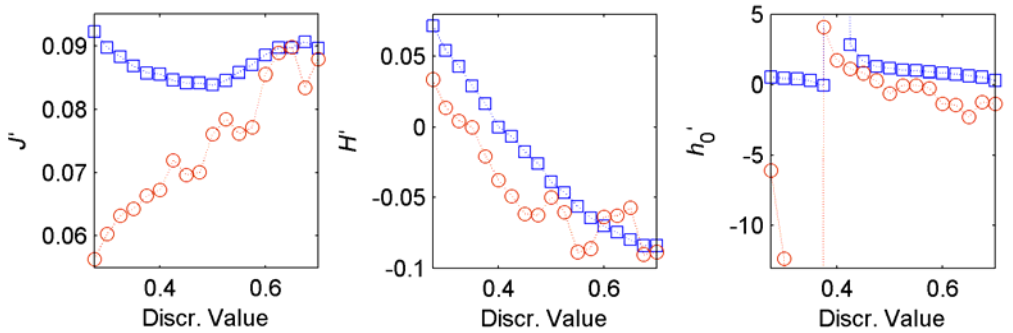

Figure 13.11. Ising model parameter estimates. Estimates J' (left), H' (centre), and h_0' (right) are shown as functions of discretisation value. Results are shown with the MTN logical topology (squares) and estimated topology (circles). Calculation: parameter values are based on a single case.

mated parameter values, based on estimated graph structures and the MTN logical topology, as functions of discretisation threshold. With the MTN logical topology, J' and H' are both rather smooth, J' assuming its minimum value at a discretisation threshold of about 0.5 and H' decreasing as the discretisation threshold increases. However, h_0' behaves more strangely, diverging at discretisation threshold 0.4.

With the estimated graph structure, J' increases as the discretisation threshold increases and, at about 0.6, nearly coincides with the respective J' value with the MTN logical topology. Furthermore, H' is similar with both topologies, and at about 0.6, the results with the two topologies are the most similar. With the estimated graph structure, also h_0' somewhat follows the behaviour of the respective h_0' with the MTN logical topology, except that with the estimated topology, h_0' diverges at about discretisation threshold 0.35. With both topologies, the divergence of h_0' occurs when the respective H' is close to zero. H' changing its sign, in fact, has major consequences to the model behaviour, because the impact of node loading to node state is reversed; e.g., a positive loading previously favouring one node state suddenly starts to favour the other state. Therefore, the sudden change in h_0', to some extent, compensates the change in the impact of H'.

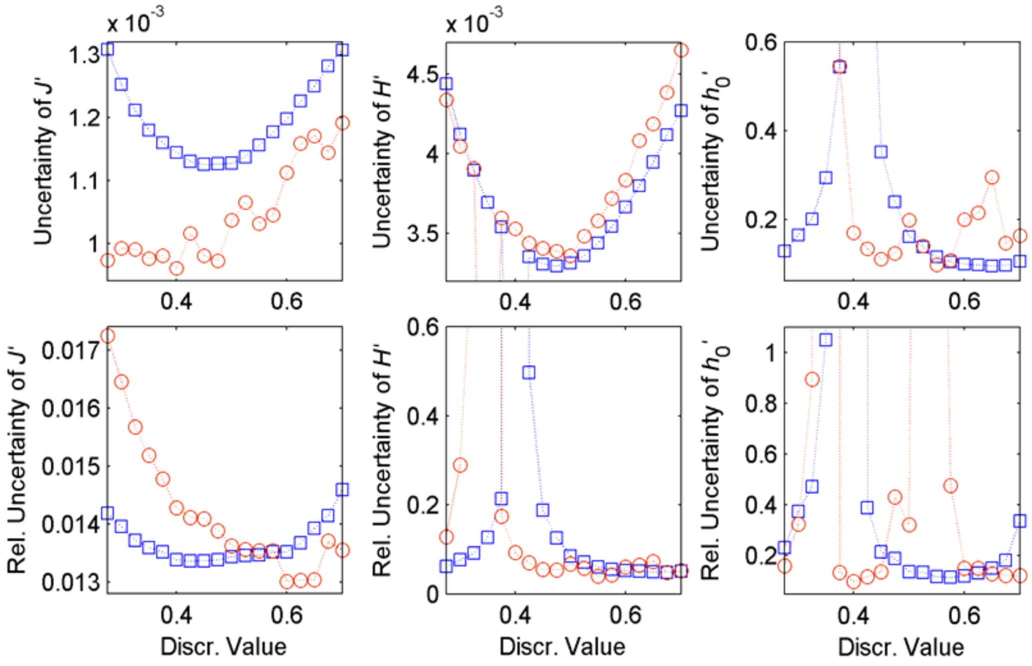

Figure 13.12. Uncertainties (top row) and relative uncertainties (in absolute values) (bottom row) of estimated Ising model parameters. Uncertainties of J' (left column), H' (centre column), and h_0' (right column) are shown as functions of discretisation threshold. Results are shown with the MTN logical topology (squares) and estimated topology (circles). Calculation: uncertainty values are based on a single case.

Parameter uncertainties are studied through Gaussian distribution estimates of true parameter probability distributions. Some numerical difficulties emerge when true marginal probability distributions are calculated, mostly because of the diverging values of h_0', causing wide variation in the log-pseudolikelihood values and spikiness in the true distributions when the exponential of the log-pseudolikelihood value is taken (see Chapter 9). Therefore, the estimated parameter values in Figure 13.11 are somewhat unreliable, in particular those of H' and h_0', which is seen in Figure 13.12, presenting the respective absolute and relative parameter uncertainties. The interaction term of the Ising model thus possibly dominates the external load term, which is also supported by the considerably large neighbourhood size (11.76) and Figure 12.10, where the term ratio measures peaked at about discretisation threshold 0.6. Moreover, the small data size compared to the size of the network may largely account for the problems of parameter estimation.

Figure 13.13 shows state probability predictions again with the six discretisation thresholds. The predictions are slightly better at large discretisation thresholds, but rather poor overall, a fact supported by Figure 13.14, showing respective absolute prediction errors and slopes of regression lines, the former decreasing and the latter increasing slightly at large discretisation thresholds. Based on the MTN logical topology, the prediction error peaks at a discretisation threshold of about 0.55 and the slope assumes its largest value at the smallest discretisation values and again at about discretisation value 0.6. Altogether, though some results are somewhat unreliable with the 132-node network, and probably because of the relatively limited data, most results support the selection of a rather large discretisation threshold, one at which nodes would assume state -1 in 60% of observations.

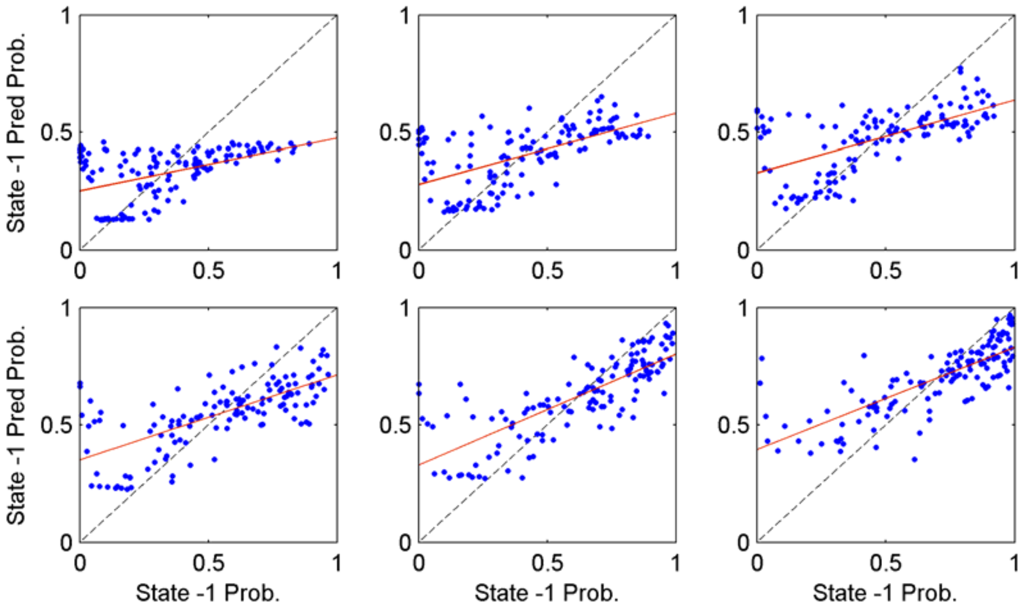

Figure 13.13. Predictions with estimated (structure and parameters) Ising models. Node state -1 probability predictions are shown for each node as a function of data-calculated node state -1 probabilities. Predictions are shown in dots, linear regression lines fitted to predictions in solid lines, and reference curves of optimal predictions in dashed lines. Predictions from top-left to bottom-right correspond to the following six discretisation thresholds: 0.325, 0.4, 0.475, 0.55, 0.625, and 0.7. Calculation: predictions are obtained with a single case.

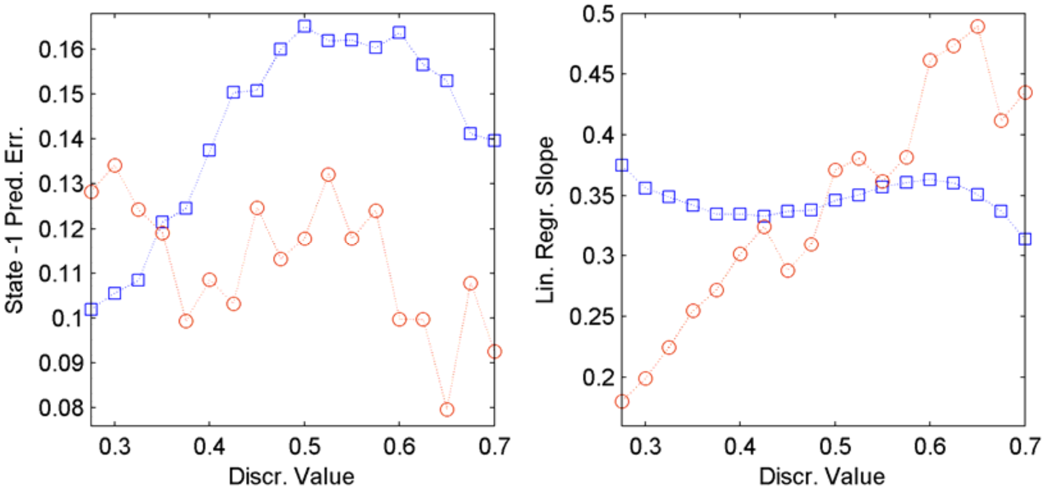

Figure 13.14. Node state -1 absolute predictions errors (left) and slopes of fitted linear regression lines (right). Results are shown for the MTN-topology-based (squares) and estimated-graph-structure-based Ising models (circles). Calculation: absolute prediction errors are medians over average node prediction errors in a single case. Slope coefficients are obtained in a single case.

14. System Properties for MTNs

This chapter studies the qualitative properties of MTNs with the Ising model by MCMC-simulating the models identified in Chapters 12–13. Because the focus is particularly on how MTNs behave under varying external loading conditions, these studies resemble Chapter 10, where the qualitative properties of the Ising model were studied under similar conditions. In Chapters 12–13, MRF models were identified mainly for a network of 30 BTS cell nodes, because much computation time was needed to test the identification with varying discretisations and neighbourhood sizes. However, here the large MTN of 132 nodes, also studied in the context of identification, is simulated, because simulation results on large networks are less sensitive to particular topological properties and to random fluctuations in node states.

In Chapters 12–13, with the network of 132 BTS cell nodes, the estimated graph structure resembled most closely the MTN logical topology if the discretisation threshold was chosen to be about 0.6; hence the graph estimate at discretisation 0.6 is used here along with the MTN logical topology. Because, to some extent, the parameter estimates turned out unreliable with the 132-node network, instead of only applying the estimated parameter here, we vary the interaction parameter of the Ising model as done with the synthetic network in Chapter 10. Nevertheless, model simulations are run also using estimated model parameters.

In structure, this chapter is similar to Chapter 10. In Section 14.1, the qualitative behaviour of the Ising model is studied via MCMC simulations under a varying global external loading using estimated graph structures and the MTN logical topology. In Section 14.2, a heavy local external loading is applied to certain nodes, and model behaviour is studied under a varying global external loading. Finally, Section 14.3 focuses on transient dynamics and state fluctuations in the identified Ising models via MCMC simulation steps.

14.1 Behaviour under Global External Loading

This section examines the global state of the Ising model under changing global external loading values using graph structures identified in Chapter 12 and the MTN logical topology for reference. Here the estimated graph structure corresponding to a discretisation threshold of 0.6 constitutes the main case, because with this discretisation the graph estimate was found appropriate in Chapters 12–13. The corresponding estimated node location map is shown in Figure 14.1.

The Ising model is simulated here first with J values of 0.02, 0.2, and 2. Therefore, simulations resemble those of the synthetic network in Section 10.1. In addition, at each J value, A' values of 8.76, 11.76, and 14.76 are used to derive graph structures with varying connectivity. The middle case of A' equals A according to the MTN logical topology. Because the case with a discretisation threshold of 0.6 is our main case, graph structures are obtained from the node location map shown in Figure 14.1. Table 14.1 shows the values of the ratio $\tilde{J} = J/A'$ with all tested J–A' combinations.

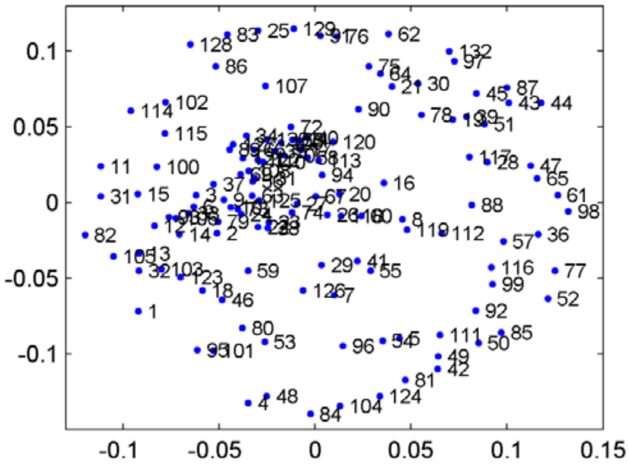

Figure 14.1. Estimated node location map of 132 nodes with a discretisation threshold of 0.6. Calculation: the node location map is Procrustes-transformed with respect to the MTN physical node location map.

A single network sample is MCMC generated with the Ising model with each nine J–A' combinations and external loading. Each simulation study is repeated, first, starting from the global initial state -1 and then from the global initial state $+1$, respectively. The two initial values are again applied to reveal possible hysteresis phenomena exhibited by the model. In addition, again in all simulations, the uniform external loading is increased and decreased adiabatically.

Figure 14.2 shows results with estimated graph structures applied in MCMC simulations. At $J = 0.02$, no hysteresis occurs, and transition is smooth between the two coherent states -1 and $+1$. Here simulations started with the two coherent states give exactly an equal global state dependence on loading, because the MCMC random number generator is seeded identically at the beginning of each simulation. These results are practically independent of neighbourhood size. At $J = 0.2$, the simulation paths corresponding to the two initial states diverge and the model exhibits hysteresis. The larger the A' and J values, the larger the coherence in the node states, requiring thus generally a large loading for $\langle s \rangle$ to change. Here differences are quite small between the three A' values.

The estimated node location map, shown in Figure 14.1, consists of a tight node cluster and nodes outside that cluster. Evidently, this structure causes the change in $\langle s \rangle$ at $J = 0.2$ to happen roughly in two steps. First, the loosely connected nodes undergo state changes more or less independently of each other. Only after the loading increases further, do the nodes in the node cluster finally undergo state changes almost simultaneously, causing discontinuity in $\langle s \rangle$. The nodes

Table 14.1. Values of ratio $\tilde{J} = J/A'$ with respect to J and A'.

J	A'	$\tilde{J}$	J	A'	$\tilde{J}$	J	A'	$\tilde{J}$
0.02	8.76	2.28x10^{-3}	0.2	8.76	2.28x10^{-2}	2	8.76	2.28x10^{-1}
0.02	11.76	1.70x10^{-3}	0.2	11.76	1.70x10^{-2}	2	11.76	1.70x10^{-1}
0.02	14.76	1.35x10^{-3}	0.2	14.76	1.35x10^{-2}	2	14.76	1.35x10^{-1}

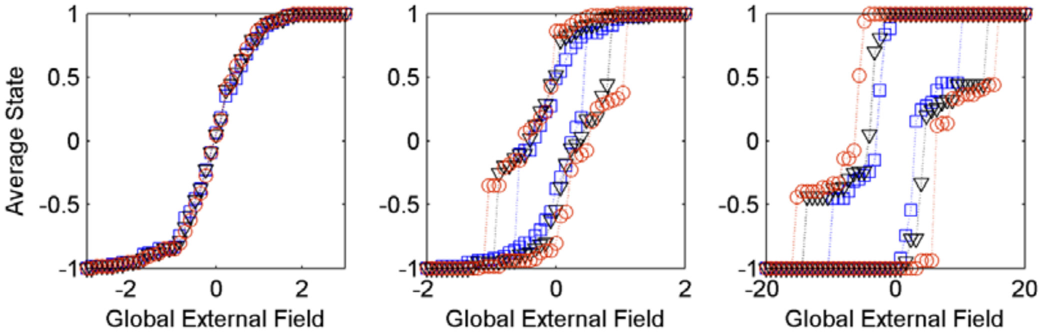

Figure 14.2. Model state behaviour with the estimated topology under global uniform external loading. The average state $\langle s \rangle$ is shown as a function of h with three J values, 0.02 (left), 0.2 (centre), and 2 (right), and each with three A' values, 8.76 (squares), 11.76 (triangles), and 14.76 (circles). Calculation: results with each h are average node states in a single simulated ensemble. With each h, two ensembles are simulated, one starting from $\langle s \rangle = -1$ and the other from $\langle s \rangle = +1$.

within the cluster change states almost simultaneously, because each has a large number of neighbours and thus the nodes behave very coherently. At $J = 2$, results are otherwise similar to those at $J = 0.2$, but because coherence is now even larger, also loosely connected nodes undergo state changes almost simultaneously. Large loadings are also required for any state change to occur; in particular, the loading value must be considerably increased for a final, discontinuous state jump to take place.

The network can be seen as consisting of two parts, which behave differently especially at a large J. The states of loosely connected nodes are still mostly determined by their external loadings, because they have only a few neighbours, whereas an increase in the interaction term makes the states of more interconnected nodes very dependent on the states of their neighbours. Consequently, the more interconnected the nodes, the more resistant the network to an increasing external loading, until at some critical loading, the nodes simultaneously change states, marking a discontinuous abrupt change in $\langle s \rangle$. Similar results are expected also with the other graph structures tested in Chapters 12–13, because they all consist of a tight cluster and loosely connected outside nodes.

The MCMC simulations are repeated using the MTN logical topology for graph structure in the Ising model. Figure 14.3 shows results at the same three J values. For comparison, results from Figure 14.2 are repeated here with a case corresponding to A in the MTN logical topology, $A = 11.76$. At $J = 0.02$, results are similar to those with the estimated topology. However, with the two larger J values, no such two-phased transition occurs in $\langle s \rangle$ as with the estimated topology. Rather (nearly) all nodes change states simultaneously, obviously because neighbourhood sizes are more evenly distributed in the MTN logical topology, giving rise to no highly connected node cluster. Evidently, the model changes its behaviour when the properties of the graph structure change.

14.2 Behaviour under Local External Loading

This section again studies the Ising model as a function of uniform global external loading, but now a subset of nodes has a constant heavy local loading. The study resembles Section 10.2,

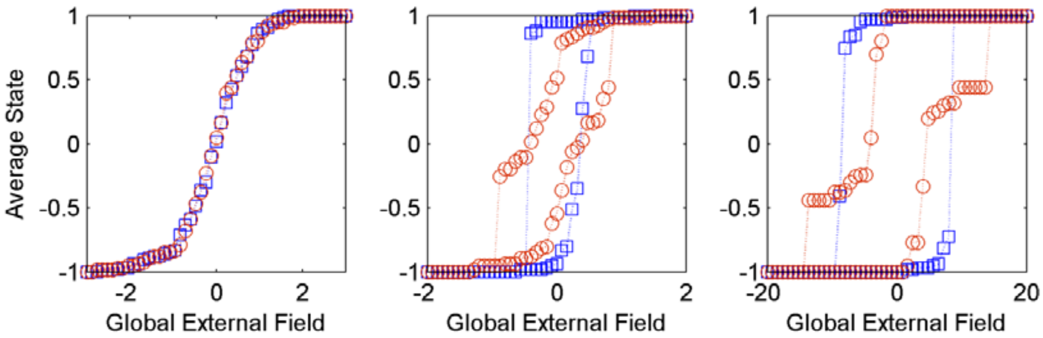

Figure 14.3. Model state behaviour under global external loading with the MTN logical topology (squares) and estimated graph structure (circles). The average state $\langle s \rangle$ is shown as a function of h with three J values, 0.02 (left), 0.2 (centre), and 2 (right), and each with $A = 11.76$. Calculation: results with each h, are average node states in a single simulated ensemble. With each h, two ensembles are simulated, one starting from $\langle s \rangle = -1$ and the other from $\langle s \rangle = +1$.

where changes in $\langle s \rangle$ started earlier but appeared more gradually than without a heavy local loading. Because loading typically correlates between physically close nodes, and to emulate true local loading changes, the heavy local loading again affects a group of nearby nodes. However, instead of choosing affected nodes according to the MTN physical topology, we select them here according to the estimated node location map with a discretisation threshold of 0.6.

Because the estimated node location map consists of a tight node cluster and a loosely connected part, the model may change its behaviour drastically depending on the network part that comes under a heavy local loading. Therefore, we consider local loading of nodes both inside and outside the node cluster and use the MTN logical topology as well. In simulations, first, a group of seven nodes is subjected to a constant heavy local loading, and loading is then further increased to cover 14 nodes; as in Section 10.2, the numbers correspond roughly to five and ten percent of a total of 132 nodes. Nodes under a constant heavy local loading ($h_{local} = 60$) are shown in Figure 14.4, which depicts the estimated node location map in Figure 14.1.

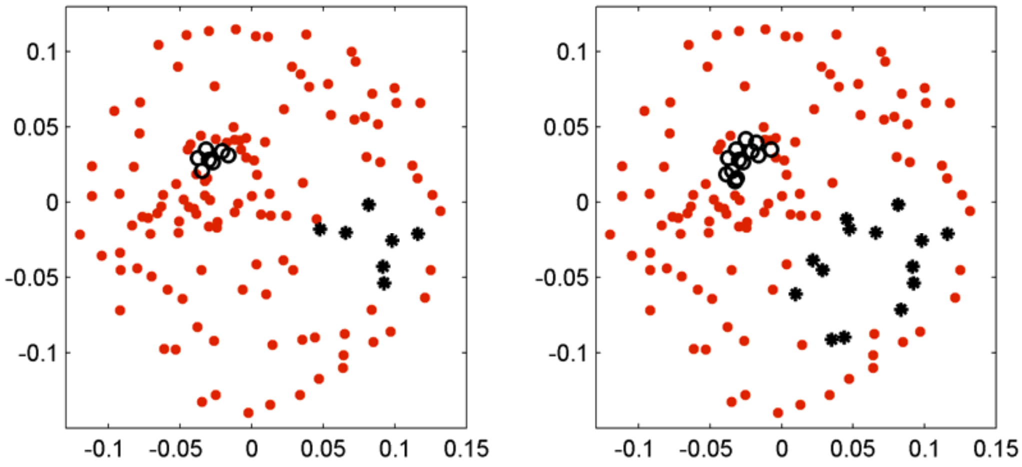

Figure 14.4. Heavy local external loading affecting 7 (left) and 14 nodes (right). Nodes inside the tight node cluster (circles) and nodes outside the node cluster (asterisks) are affected by the heavy local loading.

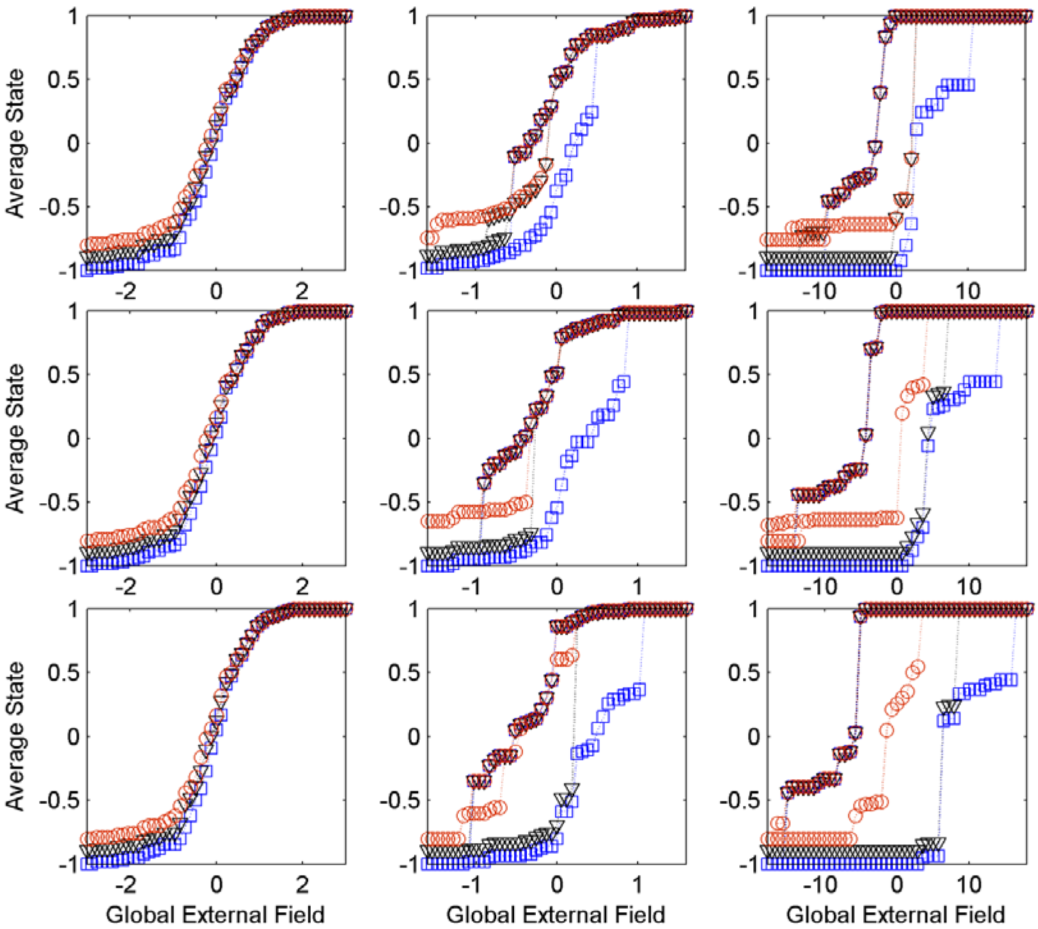

Figure 14.5. Model state behaviour with the estimated topology with a heavy local external loading affecting nodes inside the node cluster. The average state $\langle s \rangle$ is shown as a function of h with three J values, 0.02 (left column), 0.2 (centre column), and 2 (right column), and each with three A' values, 8.76 (top row), 11.76 (middle row), and 14.76 (bottom row). Heavy local loading affects 0 nodes (squares), 7 nodes (triangles), and 14 nodes (circles). Calculation: results with each h are average node states in a single simulated ensemble. With each h, two ensembles are simulated, one starting from $\langle s \rangle = -1$ and the other from $\langle s \rangle = +1$.

Let us first discuss a case with affected nodes chosen inside the node cluster, and consider the previously used nine J–A' combinations. Figure 14.5 shows the results and also the reference cases without effects of local loading. In all these cases, transitions occur earlier than in the reference cases, and in some all nodes even change their states simultaneously. However, in most cases, a two-phased transition in $\langle s \rangle$ occurs, though to complete the second phase requires a much smaller increase in loading than without the heavy local loading. Furthermore, with 14 nodes affected, the model can bear much smaller global loading values than with only seven nodes affected; thus transition occurs at a lower uniform loading. Moving from $+1$ to -1, local loading has the opposite effect with affected nodes slowing the state changes of the rest of the nodes inside the node cluster. However, transitions from $+1$ to -1 are almost identical whether local loadings are included or not, except that affected nodes do not change states at all.

Figure 14.6 shows results with all the nine J–A' combinations when affected nodes are chosen outside the node cluster. Moving from -1 to $+1$, transitions now appear similar to those without a local loading. A two-phased transition in $\langle s \rangle$ appears again with nodes outside the node

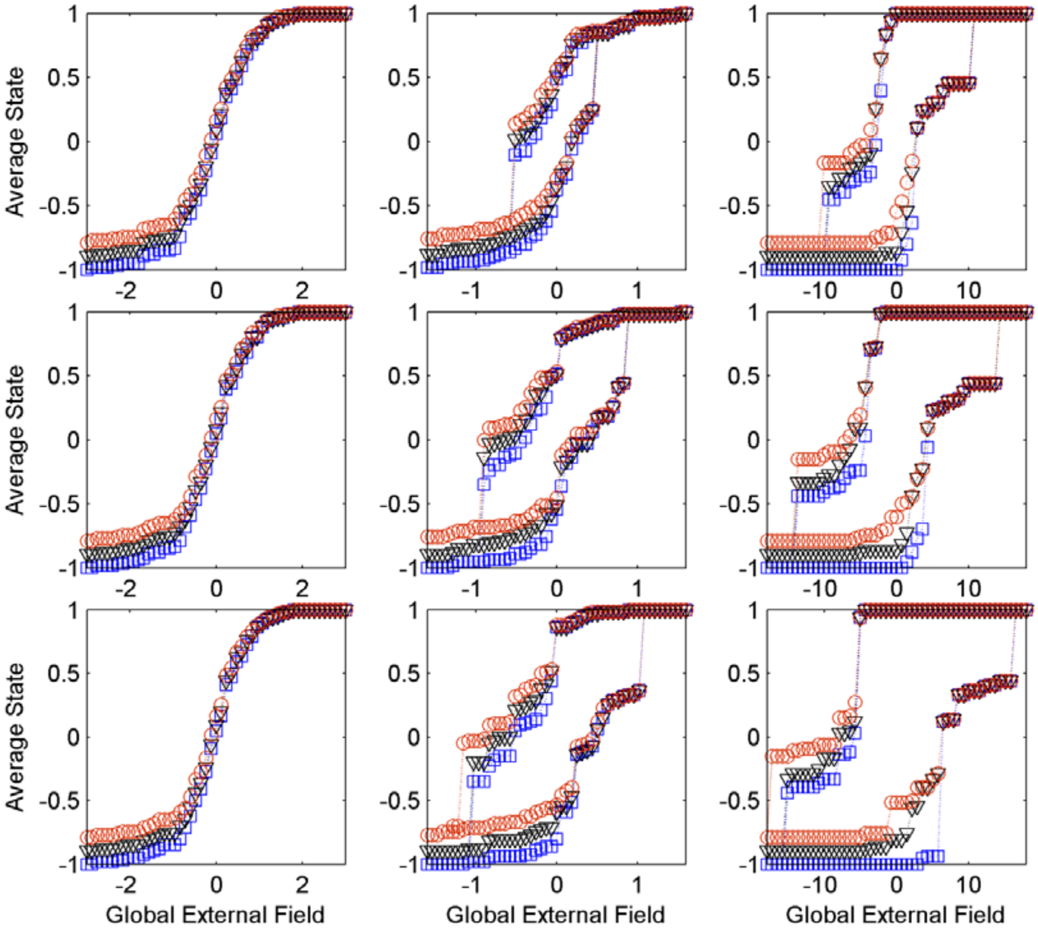

Figure 14.6. Model state behaviour with the estimated topology with a heavy local external loading affecting nodes outside the node cluster. The average state $\langle s \rangle$ is shown as a function of h with three J values, 0.02 (left column), 0.2 (centre column), and 2 (right column), and each with three A' values, 8.76 (top row), 11.76 (middle row), and 14.76 (bottom row). Heavy local loading affects 0 nodes (squares), 7 nodes (triangles), and 14 nodes (circles). Calculation: results with each h are average node states in a single simulated ensemble. With each h, two ensembles are simulated, one starting from $\langle s \rangle = -1$ and the other from $\langle s \rangle = +1$.

cluster changing states first, to be followed by nodes within the node cluster simultaneously undergoing state changes. Consequently, nodes affected by the local loading do not affect the qualitative transition properties of the average network state, because nodes chosen outside the node cluster are only loosely connected to the rest of the nodes and thus affect the node cluster very little. For the same reason, the node cluster is hardly affected at all whether seven or 14 nodes are affected. However, the number of heavily loaded nodes has its impact in that in the loosely connected part nodes change states the earlier the larger the group of affected nodes. The impact on transitions from $+1$ to -1 is mostly similar, but in some cases a node cluster undergoes a transition somewhat later because of the slowing effect of the affected nodes.

Figure 14.7 shows results in two local loading cases when the MTN logical topology is used with $A = 11.76$. As shown in Figure 12.8, the structure of the MTN logical topology is only slightly clustered. However, since the nodes affected by local loading are the same as before, i.e., chosen according to the estimated topology, they are not necessarily neighbours on the MTN logical topology. Yet again, as the local loading is now brought into the network, in some cases the other-

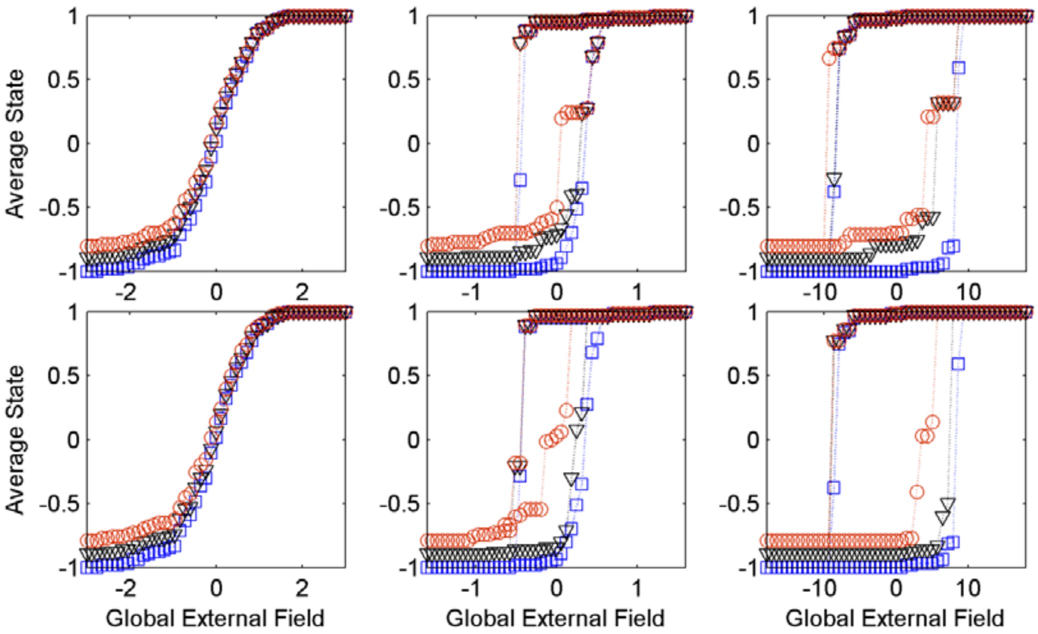

Figure 14.7. Model state behaviour with the MTN logical topology with a heavy local external loading affecting nodes inside (top row) and outside (bottom row) the node cluster. The average state $\langle s \rangle$ is shown as a function of h with three J values, 0.02 (left column), 0.2 (centre column), and 2 (right column), and each with $A = 11.76$. Heavy local loading affects 0 nodes (squares), 7 nodes (triangles), and 14 nodes (circles). Calculation: results with each h are average node states in a single simulated ensemble. With each h, two ensembles are simulated, one starting from $\langle s \rangle = -1$ and the other from $\langle s \rangle = +1$.

wise discontinuous transitions become more like two-phased transitions. Because of similarities between the estimated and MTN logical topology, results also differ whether in the estimated topology affected nodes are chosen inside or outside the node cluster.

Figure 14.8 shows simulation results when both estimated graph structures and estimated model parameters are applied with and without local loading ($h_{local} = 60$) effects. Because significant variations in the Ising model parameter estimates between varying discretisations were found in Chapter 13, the simulations are studied here with three discretisation thresholds, 0.3, 0.45, and 0.6, and with $A' = A = 11.76$. The estimated model parameters in these cases are given in Table 14.2. Because of negative H' at discretisation values 0.45 and 0.6, changes in $\langle s \rangle$ are in the opposite direction than in the previous cases above. However, with discretisation value 0.3, changes in $\langle s \rangle$ are in accordance with the previous cases, because with this discretisation H' is positive in sign. In addition, in all the three discretisation cases, changes in $\langle s \rangle$ are mostly continuous, but also some discontinuities occur in $\langle s \rangle$, partially because of rather large gaps between changes in global loading. Some cases also display modest hysteresis and the hysteresis properties seem to be somewhat affected by the heavy local loading. Otherwise, the heavy local loading have only a small effect on $\langle s \rangle$. We conclude that owing to the small data set available, relative to the network size, joint estimation of topology and parameters is rather tricky with the 132-node network. Consequently, between varying discretisation thresholds, the Ising model parameters identified result in diverse network state behaviour as a function of global loading.

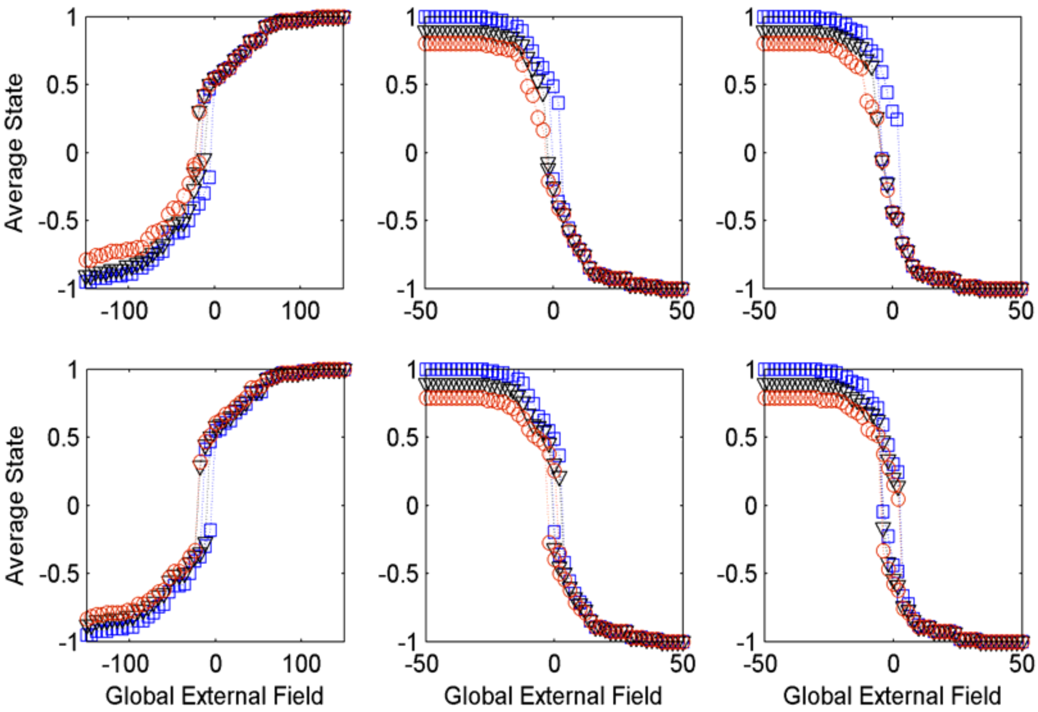

Figure 14.8. Model state behaviour with the estimated topology and estimated parameters with a heavy local external loading affecting nodes inside (top row) and outside (bottom row) the node cluster. The average state $\langle s \rangle$ is shown as a function of h with three discretisation threshold values, 0.3 (left column), 0.45 (centre column), and 0.6 (right column), and each with $A = 11.76$. Heavy local loading affects 0 nodes (squares), 7 nodes (triangles), and 14 nodes (circles). Calculation: results with each h are average node states in a single simulated ensemble. With each h, two ensembles are simulated, one starting from $\langle s \rangle = -1$ and the other from $\langle s \rangle = +1$.

14.3 MCMC Dynamics

Transient dynamics and state fluctuations are studied with the MTN of 132 nodes as with the synthetic networks in Section 10.3. Here both an estimated topology corresponding to discretisation threshold value of 0.6 and the MTN logical topology are applied in simulations. The three J values are again used, and the global uniform loading assumes the same values as the synthetic network in Section 10.3, i.e., $[-3, -2, -1, -0.5, 0, 0.5, 1, 2, 3]$ with $J = 0.02$, $[-1, -0.5, 0, 0.3, 0.5, 0.6, 0.7, 1, 1.5]$ with $J = 0.2$, and $[-1, 1, 3, 5, 7, 9, 11, 13, 15]$ with $J = 2$. Again, after each load change, the model is simulated 10^4 MCMC steps before another load change, and each simulation is started from the previous stationary state. The first simulation is started from $\langle s \rangle = -1$.

Table 14.2. Ising model parameter estimates J', H', and h_0' with discretisation thresholds 0.3, 0.45, and 0.6.

Discr. Value	J'	H'	h_0'
0.3	0.0603	0.0140	-12.3
0.45	0.0720	-0.0494	1.12
0.6	0.0855	-0.0641	-1.32

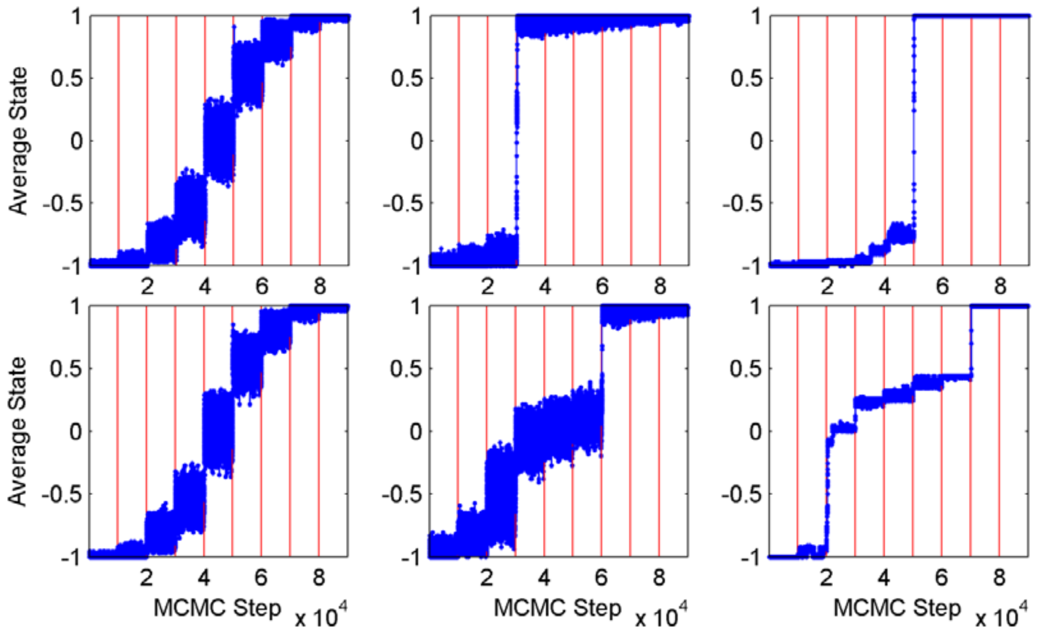

Figure 14.9. Model state fluctuations and transient dynamics under varying global external loading with the MTN logical topology (top row) and estimated topology (bottom row). The average state $\langle s \rangle$ is shown as a function of MCMC simulation step (dotted lines) for the three J values, 0.02 (left column), 0.2 (centre column), and 2 (right column), and each with $A = 11.76$. The vertical lines mark the spots were h changes. Calculation: results with each MCMC step are average node states in a single simulated ensemble. The first MCMC simulation step is started from $\langle s \rangle = -1$, and all the rest always from the previous simulated state.

Figure 14.9 shows $\langle s \rangle$ as a function of MCMC step in each case. With the MTN logical topology, changes in $\langle s \rangle$ are similar to those obtained in the synthetic case in Chapter 10. With the estimated topology, a two-phased transition appears again in $\langle s \rangle$, because the transition being a property of the topology of a central cluster and outside nodes. With both topologies, state fluctuations are larger than with the synthetic topology in Section 10.3. With the estimated topology, fluctuations are particularly large at $J = 0.2$ when only a part of the network has undergone a transition.

Convergence to a stationary probability distribution is again very fast, requiring usually no more than a few MCMC steps. However, as with the synthetic network, at the two largest J values, more discontinuous changes in $\langle s \rangle$ take several more MCMC steps. The discontinuous parts in $\langle s \rangle$ are shown in detail in Figure 14.10 as zoom-ins of the average node state curves of Figure 14.9. The figure shows with the estimated topology the step from $\langle s \rangle \approx 0$ to $\langle s \rangle \approx +1$ with $J = 0.2$ and the step from $\langle s \rangle \approx -1$ to $\langle s \rangle \approx 0$ with $J = 2$. With the MTN logical topology, discontinuous changes appear in $\langle s \rangle$ in about two steps with convergence being clearly faster with $J = 2$ because of greater interaction. With $J = 0.2$, transition takes about 300 MCMC steps, whereas with $J = 2$, it takes only about 30 steps. With the estimated topology with $J = 0.2$, transition happens fast after a lengthy period of fluctuations at around $\langle s \rangle \approx 0.2$. With $J = 2$, it appears in a few small steps.

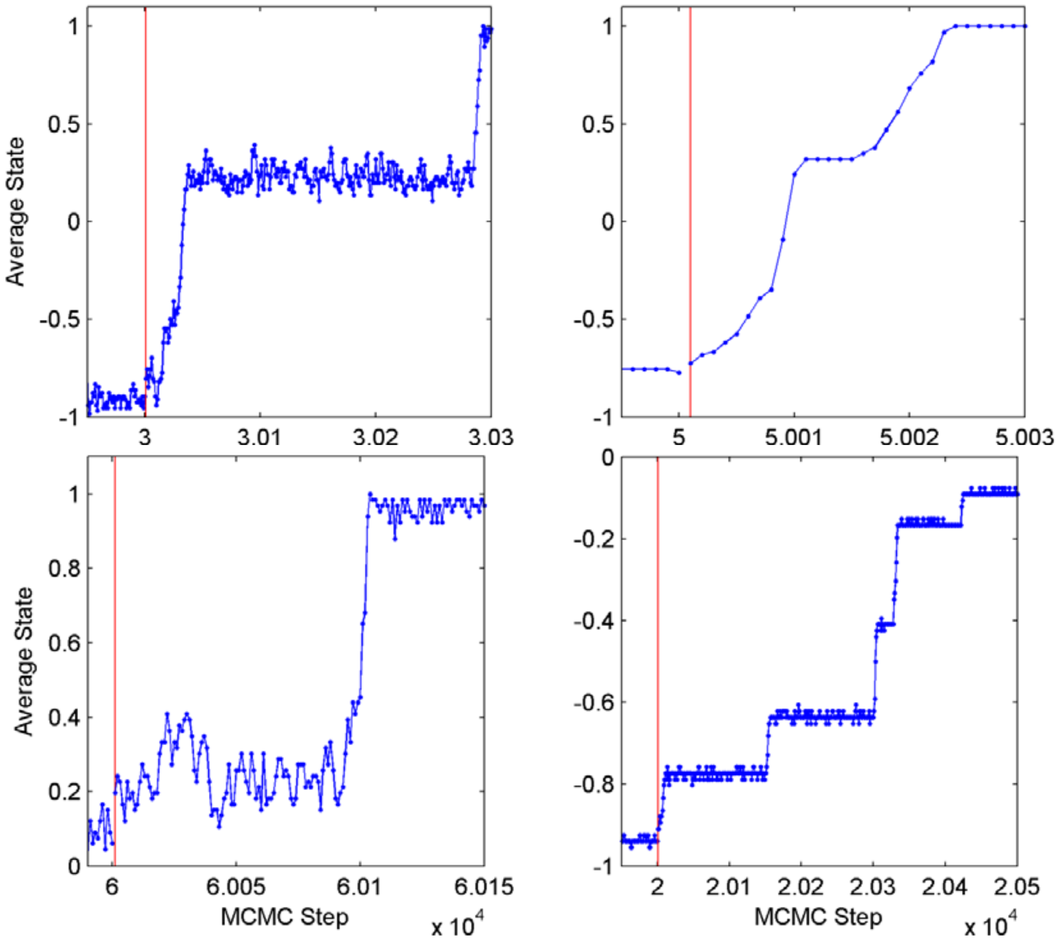

Figure 14.10. Zoom-in plots of the transient dynamics of changes in $\langle s \rangle$ shown in Figure 14.9. The two right-most columns at the top of Figure 14.9 correspond here to the top row, whereas the two right-most columns at the bottom of Figure 14.9 correspond here to the bottom row. Original plots are zoomed here to show the ranges of MCMC steps with most dramatic changes.

15. Conclusions and Discussion

Communications networks have traditionally been studied mostly by analysing the performance and optimisation of single network nodes with less emphasis on analysis of overall network performance. However, qualitative properties, which cannot be predicted or understood by inspecting only the properties of single network nodes, may emerge as a consequence of systemic collaborative network behaviour. Diverse networked systems are known to share similar, universal, complex qualitative network properties, even though the properties of their individual nodes may vary greatly. Therefore, analysis methods developed to study the overall system behaviour of one such system may be extended to cover other similar systems as well.

The Ising model is an example of a simple statistical collaborative network model originally developed to analyse ferromagnetic phenomena in physics, but as some qualitative properties the Ising model describes are universal it has since been applied widely to several diverse systems. Though the model describes single network nodes in an extremely simple fashion, it is yet capable of describing complex network phenomena, such as discontinuous phase transitions and hysteresis; hence its generality and wide applicability. Because essentially similar qualitative phenomena, such as the collapse of a coherently acting network under heavy loading, may take place also in mobile telecommunications networks, the Ising model is well suited for modelling such systems as well.

It is, in general, extremely difficult to model accurately the exact dynamical behaviour of complex networked systems, where, e.g., in specific circumstances with a heavy external loading affecting the nodes, even the smallest loading changes, e.g., due to random fluctuations, may drastically derail the network into a different state. In addition, if the network exhibits hysteresis, the recovery of the network to its original state may be substantially difficult and costly. Having a statistical description for a system, instead of trying to describe the exact system state under specific conditions, a joint probability is adjusted to the network node states. The network state and all the uncertainties related to it can then be assessed through the statistical model. Thus, for any real network it becomes of interest to define a statistical model, such as the Ising model, and to study its qualitative properties as a function of external conditions affecting the node states.

By exploiting statistical models, the uncertainties, and thus the risks, related to the behaviour of a networked system under certain conditions and external loadings become explicit for the network operator, who can then use the information in network planning and maintenance. A network's state behaviour can be studied by simulating the statistical model from a given initial state to a stationary state. Because the model is probabilistic, to make it yield a probability presentation of the final system state, a simulation must be run a number of times. Though no dynamics is essentially involved in the statistical model, and would be extremely difficult to assess, both transient dynamics (how a system transfers from one state into another) and adiabatic dynamics (how a system evolves under slowly changing conditions) can be attempted by using MCMC model simulations.

It is essential to be able to identify a statistical model from network data to obtain an appropriate model description of the true system, particularly when in a practical case the true system topology and model parameters are unknown. This thesis focused extensively on the identification of statistical models, in particular of the Ising model, by developing a topology identification method (abbreviated here as the MGMN method) and by exploiting an existing parameter estimation method, the pseudolikelihood method. Altogether, tests with model-generated synthetic network data suggest that the Ising model can be successfully identified when the network assumes neither too small nor too large coherence values—two extreme cases corresponding to nearly independent nodes, and a highly dependent node cluster, respectively. The model identification is thus useful in practical network cases when both node interactions and external loadings are contributing factors. In MCMC model simulations the practical relevance of the Ising model was demonstrated as the model being able to exhibit complex qualitative properties typical for networked systems such as discontinuous changes and hysteresis in network state under varying global external loading.

It is more difficult to test the performance of model identification methods with real network data because of lack of information about the true topology and true parameter values. The data available here was also considerably limited for any identification of large networks. For the same reason, a separate test data set was not used to evaluate the method, instead, all data was used in the identification phase. However, tests with existing data suggest that the identification methods, particularly the MGMN topology identification method, are relevant at least for mobile telecommunications networks and other similar systems, which assume an underlying spatial node configuration. Moreover, the MGMN method is expected to be relevant in general for MRF graph structure identification. In conclusion, at best, the model identification methods together with model MCMC simulations will prove themselves useful tools to help and support network operators in their decision making and in planning more reliable and efficient mobile telecommunications networks.

16. References

[1] Abdallah, S., (2002), *Towards Music Perception by Redundancy Reduction and Unsupervised Learning in Probabilistic Models*, Doc. Thesis, Department of Electronic Engineering, King's College, London.

[2] Abdallah, S., and Plumbley, M. (2004), “Application of geometric dependency analysis to the separation of convolved mixtures,” *Proc. ICA'04*, 540–547.

[3] Abellán, J., Gómez-Olmedo, M., and Moral, S. (2006), “Some variations on the PC algorithm,” *Proc of the Third European Workshop on Probabilistic Graphical Models.*

[4] Acharyya, M. (1999), “Nonequilibrium phase transition in the kinetic Ising model: Existence of a tricritical point and stochastic resonance,” *Phys. Rev. E*, 59(1), 218–221.

[5] Acharyya, M., Chakrabarti, B.K., and Stinchcombe, R.B. (1994), “Hysteresis in Ising model in transverse field,” *J. Phys. A: Math. Gen.*, 27, 1533-1540.

[6] Acharyya, M., Chakrabarti, B.K. (1995), “Response of Ising systems to oscillating and pulsed fields: Hysteresis, ac, and pulse susceptibility,” *Phys. Rev. B.*, 52(9), 6550–6568.

[7] Agrafiotis, D., Rassokhin, D.N., and Lobanov, V.S. (2001), “Multidimensional scaling and visualization of large molecular similarity tables,” *Journal of Computational Chemistry*, 22(5), 488–500.

[8] Ahsan, A., Rudnick, J., and Bruinsma, R (1998), “Elasticity theory of the B-DNA to S-DNA transition,” *Biophys J*, 74, 132–137.

[9] Albert, R., and Barabási, A.-L. (2002), “Statistical mechanics of complex networks,” *Rev. Modern Phys.*, 74, 47–97.

[10] Albert, R., Jeong, H., and Barabási, A.-L. (2000), “Error and attack tolerance of complex networks,” *Nature*, 406, 378.

[11] Anghel, M., Werley, K.A., and Motter, A.E. (2007), “Stochastic model for power grid dynamics,” *Proc. of the 40th Hawaii International Conference on System Sciences.*

[12] Arndt, C. (2004), *Information Measures: Information and Its Description in Science and Engineering*, Springer-Verlag.

[13] Barnett, G., A. (2001), “A longitudinal analysis of the international telecommunication network, 1978-1996,” *American Behavioral Scientist* 44(10), 1638–1655.

[14] Bar-Yam, Y. (1997), *Dynamics of Complex Systems*, Westview Press.

[15] Basalaj, W. (1999), “Incremental multidimensional scaling method for database visualization,” *Proc. SPIE'99*, 149–158.

[16] Besag, J.E. (1974), “Spatial interaction and the statistical analysis of lattice systems,” *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 36, 192–225.

[17] Besag, J.E. (1975), “Statistical analysis of non-lattice systems,” *The Statistician*, 24, 179–195.

[18] Bishop, C.M. (2006), *Pattern Recognition and Machine Learning*, Springer, 383–393.

[19] Boucherie , R.J., and Van Dijk, N.M. (2000), “On a queueing network model for cellular mobile telecommunications networks,” *Operations Research*, 48(1), 38–49

[20] Breitbart, Y., Garofalakis, M., Jai, B., Martin, C., Rastogi, R., and Silbershatz, A. (2004), “Topology discovery in heterogeneous IP networks: the NetInventory system,” *IEEE/ACM Transactions on Networking*, 12(3), 401–414.

[21] Brown, P., Cocke, J., Della Pietra, S., Della Pietra, V, Jelinek, F., Mercer, R, and Roossin, P. (1988), “A statistical approach to language translation,” *COLING-88*, 1, 71–76.

[22] Brush, S.G. (1967), “History of the Lenz-Ising model,” *Rev. Mod. Phys.*, 39(4), 883–893.

[23] Bromberg, F., Margaritis, D., and Honavar, V. (2006), “Efficient Markov network structure discovery using independence tests,” *Proceedings of the 2006 SIAM International Conference on Data Mining*, 141–152.

[24] Butte, A.J., and Kohane, I.S. (2000), “Mutual information relevance networks: Functional genomic clustering using pairwise entropy measurements,” *Pacific Symposium on Biocomputing 5*, 15–426.

[25] Callaway, D.S, Newman, M.E.J., Strogatz, S.H., and Watts, D.J. (2000), "Network robustness and fragility: Percolation on random graphs", *Phys. Rev. Lett.*, 85(25), 5468–5471.

[26] Castillo, C. (2004), *Effective Web Crawling*, Doc. Thesis, University of Chile, Chile.

[27] Chakrabarti, B.K., and Acharyya, M. (1999), "Dynamic transitions and hysteresis", *Reviews of Modern Physics*, 71(3), 847–859.

[28] Chen, H., and Varshney, P. (2003), "Mutual information-based CT-MR brain image registration using generalized partial volume joint histogram estimation," *IEEE Trans. on Medical Imaging*, 22(9), 1111–1119.

[29] Chen, H., and Varshney, P., and Arora, M.K. (2003), "Performance of mutual information similarity measure for registration of multitemporal remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, 41(11), 2445–2454.

[30] Cover, T.M., and Thomas, J.A. (2006), *Elements of Information Theory*, 2nd ed., John Wiley & Sons.

[31] Cheng, J., Bell, D., and Liu, W. (1998), "Learning Bayesian networks from data: An efficient approach based on information theory," Technical Report, Department of Computer Science, University of Alberta.

[32] Cheng, J., Greiner, R., Kelly, J., Bell, D., and Liu, W. (2002), "Learning Bayesian networks from data: An information-theory based approach," *Artificial Intelligence*, 137(1–2), 43–90.

[33] Cressie, N.A.C. (1993), *Statistics for Spatial Data*, John Wiley & Sons, 383–573.

[34] Crucitti, P, Latora, V., Marchiori, M., and Rapisarda, A. (2003), "Efficiency of scale-free networks: Error and attack tolerance," *Physica A: Statistical Mechanics and its Applications*, 320, 622–642.

[35] Dahmen, K.A., Sethna, J.P., Kuntz, M.C., Perkovic, O. (2001), "Hysteresis and avalanches: Phase transitions and critical phenomena in driven disordered systems,"*Journal of Magnetism and Magnetic Materials*, 1287–1292.

[36] 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.

[37] Dhar, D., Shukla, P., and Sethna, J.P. (1997), "Zero-temperature hysteresis in the random field Ising model on a Bethe lattice," *J. Phys. A,* 30, 5259.

[38] Diestel, R. (2000), *Graph Theory*, Springer-Verlag.

[39] Dixit, A. (1989), "Hysteresis, import penetration, and exchange rate pass-through," *The Quarterly Journal of Economics*, 104(2), 205–228.

[40] Dobson, I., Carreras, B.A., Lynch, V.E., and Newman, D.E. (2007), "Complex systems analysis of series of blackouts: Cascading failure, critical points, and self-organization," *Chaos*, 17.

[41] Dorogovtsev, S.N. (2004) "Clustering of correlated networks," *Phys. Rev. E*, 69(2), 027104, 1–4.

[42] Dorogovtsev, S.N., Goltsev, A.V., and Mendes, J.F.F. (2007), "Critical phenomena in complex networks," *arXiv.org*, Cornell University Library.

[43] Dorogovtsev, S.N., Goltsev, A.V., and Mendes, J.F.F. (2002), "Ising model on networks with an arbitrary distribution of connections," *Phys. Rev. E*, 66(1), 016104.

[44] Dorogovtsev, S.N., and Mendes, J.F.F. (2002), "Evolution of networks," *Advances in Physics*, 51(4), 1079–1187.

[45] Drees, B.L., Thorsson, V., Carter, G.W., Rives, A.W., Raymond, M.Z., Avila-Campillo, I., Shannon, P., and Galitski, T. (2005), "Derivation of genetic interaction networks from quantitative phenotype data," *Genome Biology*, 6:R38.

[46] Duda, R.O., Hart, P.E., Stork, D.G. (2001), *Pattern Classification*, 2nd ed., John Wiley & Sons, Inc.

[47] Erdõs, P., and Rényi, A. (1959), "On random graphs," *Publ. Math. Debrecen*, 6, 290–297.

[48] Erdõs, P., and Rényi, A. (1960), "On the evolution of random graphs," *Publ. Math. Inst. Hung. Acad. Sci.*, 5, 17.

[49] Esposito, F., Scarabino, T., Hyvärinen, A., Himberg, J., Formisano, E., Comani, S., Tedeschi, G., Goebel, R., Seifritz, E., and Di Salle, F. (2005), "Independent component analysis of fMRI group studies by self-organizing clustering," *NeuroImage*, 25(1), 193–205.

[50] Everitt, B.S., and Rabe-Hesketh, S. (1997), *Kendall's Library of Statistics 4: The Analysis of Proximity Data*, Arnold, 11–68.

[51] Friedman, N., and Koller, D. (2003), "Being Bayesian about network structure: A Bayesian approach to structure discovery in Bayesian networks," *Machine Learning*, 50, 95–126.

[52] Friston, K.J., Frith, C.D., Fletcher, P., Liddle, P.F., and Frackowiak, R.S. (1996) "Functional topography: multidimensional scaling and functional connectivity in the brain," *Cereb. Cortex*, 6, 156–164.

[53] Gaens, T., Maes, F., Vandermeulen, D., and Suetens, P. (1998), "Non-rigid Multimodal Image Registration Using Mutual Information," W.M. Wells *et al.* (Eds.): *MICCAI'98, LNCS 1496*, 1099–1106.

[54] Gandhi, P., Bromberg, F., and Margaritis, D. (2008), "Learning markov network structure using few independence tests," *Proceedings of the 2008 SIAM International Conference on Data Mining*, 680–691.

[55] Gilks, W.R, Richardson, S., and Spiegelhalter, D.J (1996), *Markov Chain Monte Carlo in Practice*, Chapman & Hall/CRC.

[56] Goebel, B.D.Z., Dawy, Z., Hagenauer, J., and Mueller, J.C. (2005), "An approximation to the distribution of finite sample size mutual information estimates," *Proc. IEEE International Conference on Communications*, 2, 1102–1106.

[57] Gower, J.C. (1975), "Generalized procrustes analysis," *Psychometrika*, 40(1), 33–51.

[58] Grimmett, G., and Stirzaker, D. (2004), *Probability and Random Processes*, 3rd ed., Oxford University Press.

[59] Hartman, L. (2007), *Spatial Statistics and Ancestral Recombination Graphs*, Doc. Thesis, Lund University, Sweden.

[60] Hayes, J.F., and Ganesh Babu, T.V.J. (2004), *Modeling and Analysis of Telecommunications Networks*, John Wiley & Sons, Inc.

[61] Hellebrandt, M., Mathar, R., and Scheibenbogen, M. (1997), "Estimating position and velocity of mobiles in a cellular radio network," *IEEE Trans. Vehicular Technology*, 46(1), 65–71.

[62] Hovorka, O., and Frieman, G. (2007), "Non-converging hysteresis cycles in random spin networks," *arXiv:cond-mat/0703525v2*.

[63] Hutter, M. (2002), "Distribution of mutual information" In: Dietterich, T.G., Becker, S., Ghahramani, Z. (Eds.), *Advances in Neural Information Processing Systems*, 14, MIT Press, 399–406.

[64] Hutter, M, and Zaffalon, M. (2005), "Distribution of mutual information from complete and incomplete data," *Computational Statistics & Data Analysis*, 48(3), 633–657.

[65] Imry, Y. and Ma, S.-K. (1975), "Random-field instability of the ordered state of continuous symmetry," *Phys. Rev. Lett.*, 35(21), 1399.

[66] Ising, E. (1925), "Beitrag zur Theorie des Ferromagnetismus," *Z. Physik.*, 31, 253–258.

[67] Ji, X., and Zha, H. (2004), "Sensor positioning in wireless ad-hoc sensor networks using multidimensional scaling," *INFOCOM 4*, 2652–2661.

[68] Kalisch, M., and Bühlmann, P. (2007), "Robustification of the PC-algorithm for directed acyclic graphs," To appear in the *Journal of Computational and Graphical Statistics*.

[69] Kendall, M.G. (1938), "A new measure of rank correlation," *Biometrika*, 30, 81–93.

[70] Kindermann, R., and Snell, J.L. (1980), *Markov Random Fields and Their Applications*, American Mathematical Society, Providence, RI.

[71] Kishino, H., and Waddell, P.J. (2000), "Correspondence analysis of genes and tissue types and finding genetic links from microarray data," *Genome Informatics*, 11, 83–95.

[72] Krejci P., and Sprekels J. (1997), "On a system of nonlinear PDE's with temperature-dependent hysteresis in one-dimensional thermoplasticity," *J. Math. Anal. Appl.*, 209, 25–46.

[73] Krisnamachari, B., Wicker, S.B., and Béjar, R. (2001), "Phase transition phenomena in wireless ad hoc networks," *GLOBECOM'01*.

[74] Kruskal, J.B. (1964), "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psychometrika*, 29, 1–27.

[75] Kruskal, J.B. (1964), "Nonmetric multidimensional scaling: A numerical method," *Psychometrika*, 29, 115–129.

[76] Kullback, S., and Leibler, R.A. (1951), "On information and sufficiency," *Ann Math. Statist.*, 22(1), 79–86.

[77] Kullback, S. (1959), *Information Theory and Statistics*. John Wiley (1959); reprinted by Dover Publications, Inc. (1997), 155–188.

[78] Kumpulainen, P., and Kylväjä, M. (2008), ”Local anomaly detection for mobile network monitoring,” *Information Sciences*, 178, 3840–3859.

[79] Kylväjä, M., Kumpulainen, P., and Hätönen, K. (2005), “Information summarization for network performance management,” *Proc. IMEKO TC10*, 167–172.

[80] Laiho, J. (2002), *Radio Network Planning and Optimisation for WCDMA*. Doc. Thesis, Helsinki University of Technology, Finland.

[81] Lehtimäki, P. (2008), *Data Analysis Methods for Cellular Network Performance Optimization*, Doc. Thesis, Helsinki University of Technology, Finland.

[82] Li, F. (2007), *Structure Learning with Large Sparse Undirected Graphs and Its Applications*, Doc. Thesis, Carnegie Mellon University, USA.

[83] Li, W., and Chao, X. (2004), “Modeling and Performance Evaluation of a Cellular Mobile Network,” IEEE/*ACM Transactions on Networking*, 12(1), 131–145.

[84] Liang, S., Fuhrman, S., and Somogyi, R. (1998), “REVEAL, a general reverse engineering algorithm for inference of genetic network architectures,” *Pacific Symposium on Biocomputing 3*, 18–29.

[85] Liao, H., Apt, J., and Talukdar, S. (2004) “Phase transitions in the probability of cascading failures,” *Proc. Electricity Transmission in Deregulated Markets*.

[86] Lifshitz, E.M., and Landau, L.D. (1980), *Statistical Physics: Course of Theoretical Physics, Volume 5*, 3rd ed., Pergamon Press Ltd.

[87] Lin, J. (1991), “Divergence measures based on the Shannon entropy,” *IEEE T Inform Theory*, 37(1), 145–151.

[88] Lee, S.-I., Ganapahthi, V., and Koller, D. (2007), ”Efficient structure learning of Markov networks using L_1-regularization,” *Advances in Neural Information Processing Systems*.

[89] Lenz, W. (1920), “Beitrag zum Verständnis der magnetishen Erscheinungen in festen Körpern,” *Z. Physik*, 21, 613–615.

[90] Lo, W.S., and Pelcovits, R.A. (1990), “Ising model in a time-dependent magnetic field,” *Phys. Rev. A*, 42(12), 7471–7474.

[91] MacKay, D. (2003), *Information Theory, Inference and Learning Algorithms*, Cambridge University Press, 357–421.

[92] Majewski, J., Li, H., and Ott, J. (2001), “The Ising model in physics and statistical genetics,” *Am. J. Hum. Genet.*, 69(8), 53–862.

[93] Malarz, K.W., Antosiewicz, J., Karpinska, K., Kulakowski, and Tadic, B. (2007), “Avalanches in complex spin networks,” *Physica A*, 373, 785–795.

[94] Margaritis, D. (2003), *Learning Bayesian Network Model Structure from Data*, Doc Thesis, Carnegie Mellon University, Pittsburgh, USA.

[95] Margaritis, D., and Thrun, S. (1999), “Bayesian network induction via local neighbourhoods,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*.

[96] Martín-Merino, M., and Muñoz, A. (2004), “A new MDS algorithm for textual data analysis,” *Proc. ICONIP’04, LNCS 3316*, 860–867.

[97] Mathiassen, J., Skavhaug, A., and Bø, K. (2002), “Texture similarity measure using Kullback-Leibler divergence between Gamma distributions,” *In ECCV 2002 Part III, LNCS 2352* (eds. A. Heyden *et al.*), 133–147.

[98] Mayergoyz, I.D. (1986), “Mathematical models of hysteresis,” *Phys. Rev. Lett.*, 56(15), 1518–1521.

[99] Meyer, C.D. (2000), *Matrix Analysis and Applied Linear Algebra*, SIAM.

[100] Mihalkova, L., and Mooney, R.J. (2007), “Bottom-up learning of Markov logic network structure,” *Proceedings of the* 24th *International Conference on Machine Learning*, 625–632.

[101] Molloy M., and Reed, B., (1995), “A critical point for random graphs with a given degree sequence,” *Random Structures and Algorithms*, 6, 161.

[102] Molloy M., and Reed, B., (1998), "The size of the giant component of a random graph with a given degree sequence," *Combinatorics, Probability and Computing*, 7, 295.

[103] Muñoz, A., and Martín-Merino, M. (2002), "New asymmetric iterative scaling models for the generation of textual word maps," *Proc. of JADT 2002.*

[104] Nedic, D.P., Dobson, I., Kirschen, D.S., Carreras, B.A., and Lynch, V.E. (2005), "Criticality in a cascading failure blackout model," *15th Power Systems Computation Conference.*

[105] Neopolitan, R.E. (2004), *Learning Bayesian Networks*, Pearson Prentice Hall.

[106] Newman, M.E.J. (2003), "The structure and function of complex networks," *SIAM Rev.*, 45(2), 167–256.

[107] Newman, M.E.J., Strogatz, S.H., and Watts, D.J. (2001), "Random graphs with arbitrary degree distributions and their applications," *Phys. Rev. E*, 64(2).

[108] Niss, M. (2005), "History of the Lenz-Ising model 1920–1950: From ferromagnetic to cooperative phenomena," *Arch. Hist. Exact Sci.*, 59, 267–318.

[109] Niu, C., and Grimson, E. (2006), "Recovering non-overlapping network topology using far-field vehicle tracking data," *Proc. ICPR'06*, 944–949.

[110] Ohira, T., and Sawatari, R. (1998), "Phase transition in a computer network traffic model," *Physical Review E*, 58(1), 193–195.

[111] Perkovic, O., Dahmen, K., and Sethnal, J. P. (1995), "Avalanches, Barkhausen noise, and plain old criticality," *Phys. Rev. Lett.*, 75, 4528.

[112] Poston, T., and Stewart, I. (1978), *Catastrophe Theory and Its Applications*, Pitman, London.

[113] Potts, R.B. (1952), "Some generalized order-disorder transformations," *Proc. Camb. Phil. Soc.*, 48, 106–109.

[114] Press W., Flannery, B., Teukolsky, S., and Vetterling, W. (1986), *Numerical Recipes: The Art of Scientific Computation.* Cambridge University Press, 476–481.

[115] Rajala, M., and Ritala, R. (2005), "Identification of diagnostics statistical state model for networked systems," *Proc. IMEKO TC10*, 161–166.

[116] Rajala, M., and Ritala, R. (2006), "Mutual information and multidimensional scaling as means to reconstruct network topology," *Proc. SICE-ICCAS'06*, 1398–1403.

[117] Rajala, M., and Ritala, R. (2006), "Statistical model describing networked systems phenomena," *Proc. ISCC'06*, 647–654.

[118] Rajala, M., and Ritala, R. (2007), "A method to estimate the graph structure for a large MRF model," In *ICANN 2007 Part II, LNCS 4669* (eds. J. Marques de Sá *et al.*), 836–849.

[119] Rajala, M. (2008), "Heavy Loading Effects in Networked Systems," *Proc. SMCia'08*, 119–124.

[120] Rangarajan, A., Chui, H., and Duncan, J.S. (1999), "Rigid point feature registration using mutual information," *Medical Image Analysis*, 3(4), 425–440.

[121] Robert, C.P. (2007), *The Bayesian Choice: From Decision Theoretic Foundations to Computational Implementation*, 2nd ed., Springer Science + Business Media, LLC.

[122] Rue, H., and Held, L. (2005), *Gaussian Markov Random Fields: Theory and Applications*, Chapman & Hall/CRC.

[123] Sabhapandit, S. (2002), *Hysteresis and Avalances in the Random Field Ising Model*, Doc. Thesis, University of Mumbai, India.

[124] Sabhapandit, S., Dhar, D., and Shukla, P. (2002), "Hysteresis in the random field Ising model and bootstrap percolation," *Physical Review Letters*, 88, 197202.

[125] Sachtjen, M.L., Carreras, B.A., and Lynch, V.E. (2000), "Disturbances in a power transmission system," *Phys. Rev. E*, 61, 4877–4882.

[126] Schroeder, D.V. (1999), *An Introduction to Thermal Physics*, Addison-Wesley.

[127] Seber, G. (1984), *Multivariate Observations*, John Wiley, 235–256.

[128] Sethna, J.P., Dahmen, K., Kartha, S., Krumhansl, J.A., Roberts, B.W., and Shore, J.D. (1993), "Hysteresis and hierarchies: Dynamics of disorder-driven first-order phase transformations," *Phys. Rev. Lett.*, 70, 3347.

[129] Sethna, J.P., Dahmen, K.A., and Myers, C.R. (2001), "Crackling noise," *Nature*, 410, 242.

[130] Sethna, J.P., Dahmen, K.A., and Perkovic, O. (2005), "Random-field Ising models of hysteresis," arXiv:cond-mat/0406320.

[131] Shannon, C.E. (1948), "A mathematical theory of communication," *The Bell System Technical Journal*, 27, 379–423, 623–656.

[132] Shih, M.-F., and Hero III, A. (2004), "Network topology discovery using finite mixture models," *Proc. ICASSP'04*, 433–436.

[133] Solomonoff, R. and Rapoport, A. (1951), "Connectivity of random nets", *Bull. Math. Biophys.*, 13, 107–117.

[134] Sornette, D. (2002), *Why Stock Markets Crash: Critical Events in Complex Financial Systems*, Princeton University Press.

[135] Sornette, D., and Zhou, W.-X. (2006), "Importance of positive feedbacks and overconfidence in a self-fulfilling Ising model of financial markets", *Physica A: Statistical Mechanics and Its Applications*, 370(2), 704–726.

[136] Spearman, C. (1904), "The proof and measurement of association between two things," *American Journal of Psychology*, 15, 72–101.

[137] Spirtes, P., Glymour, C., and Scheines, R. (2000), *Causation, Prediction, and Search.* The MIT Press, Cambridge, Massachusetts.

[138] Sporns, O., and Tononi, G. (2002), "Classes of network connectivity and dynamics," *Complexity*, 7(1), 28–38.

[139] Sprott, C. (2003), *Chaos and Time Series Analysis*, Oxford University Press.

[140] Stanley, H.E., Stauffer, D., Kertész, J., and Herrmann, H.J. (1987), "Dynamics of spreading phenomena in two-dimensional Ising models," *Phys. Rev. Lett.*, 59(20), 2326–2328.

[141] Steuer, R., Kurths, J., Daub, C.O., Weise, J., and Selbig, J. (2002), "The mutual information: Detecting and evaluating dependencies between variables," *Bioinformatics*, 18(2), 231–240.

[142] Steyvers, M. (2002), "Multidimensional scaling," In *Encyclopedia of Cognitive Science*, Macmillan, London.

[143] Strogatz, S.H. (2001), "Exploring complex networks," *Nature*, 410, 268–276.

[144] Stuart, A., Ord, K., and Arnold, S. (1999), *Kendall's Advanced Theory of Statistics, Vol. 2A: Classical Inference & the Linear Model.* 6th Ed., Oxford University Press, 46–116.

[145] Suutarinen, J. (1994), *Performance Measurements of GSM Base Station System.* Thesis (Lic. Tech.), Tampere University of Technology, Finland.

[146] Szabó, G., and Kádár G. (1998), "Magnetic hysteresis in an Ising-like dipole-dipole model," *Phys. Rev. B*, 58(9), 5584–5587.

[147] Tadic B., Malarz, K., and Kulakowski, K. (2005), "Magnetization reversal in spin patterns with complex geometry," *Phys. Rev. Lett.*, 94, 137204.

[148] Tagaris, G.A., Richter, W., Kim, S.-G., Pellizzer, G., Andersen, P., Ugurbil, K., and Georgopuolos, A.P (1998), "Functional magnetic resonance imaging of mental rotation and memory scanning: a multidimensional scaling analysis of brain activation patterns," *Brain Research Review*, 26, 106–112.

[149] Thomas, J., Ramakrishnan, N., and Bailey-Kellogg, C. (2007), "Graphical models of residue coupling in protein families," Accepted to *IEEE/ACM Transactions on Computational Biology and Bioinformatics.*

[150] Thomsen, J.M., and Manggaard, R. (2003), *Analysis of GSM Handover Using Coloured Petri Nets*, MSc Thesis, University of Aarhus, Denmark.

[151] Tieu, K., Dalley, E., and Grimson, W. (2005), "Inference of nonoverlapping camera network topology by measuring statistical dependence," *Proc. ICCV'05*, 2, 1842–1849.

[152] Torgerson, W.S. (1952), "Multidimensinal scaling: I. Theory and method," *Psychometrika*, 17(4), 401–419.

[153] Tretyakov, A.Y., Takayasu, H., and Takahasu, M. (1998), "Phase transition in a computer network model," *Physica A*, 253, 315–322.

[154] Vanderwalle, N., Ausloos M., Boveroux P., and Minguet A. (1998), "How the financial crash of October 1997 could have been predicted," *Eur. Phys. J. B*, 4, 139–141.

[155] Vasconcelos, N., and Lippman, A. (2000), "A unifying view of image similarity," *Proc. ICPR'00*, 38–41.

[156] Vehviläinen, P. (2004), *Data Mining for Managing Intrinsic Quality of Service in Digital Mobile Telecommunications Networks*. Doc. Thesis, Tampere University of Technology, Finland.

[157] Vehviläinen, P., Hätönen, K., and Kumpulainen, P. (2003), "Data mining in quality analysis of digital mobile telecommunications network," *Proc. XVII IMEKO World Congress*.

[158] Vives, E., and Planes, A. (2001), "Hysteresis and avalanches in the random anisotropy Ising model", *Physical Review B*, 63, 134431.

[159] Wachowiak, M. P., Smolíková, R., Tourassi, G., and Elmaghraby, S. (2002), "Generalised mutual information similarity metrics for multimodel biomedical image registration," *Proc. EMBS*, 23–26.

[160] Watts, D.J., and Strogatz, S.H. (1998), "Collective dynamics of 'small-world' networks," *Nature*, 393, 440–442.

[161] Winkler, G. (2003), *Image Analysis, Random Fields and Markov Chain Monte Carlo Methods*. 2nd ed., Springer-Verlag.

[162] Wu, H., and Huffer, F.W. (1997), "Modelling the distribution of plant species using the autologistic regression model," *Environmental and Ecological Statistics*, 4, 49–64.

[163] Wu, F.Y. (1982), "The Potts model," *Rev. Mod. Phys.*, 54(1), 235–268.

[164] Yang, C.N. (1952), "The spontaneous magnetization of a two-dimensional Ising model," *Phys. Rev.*, 85(5), 808–816.

[165] Yang, Z.R., and Zwolinski, M. (2001), "Mutual information theory for adaptive mixture models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(4), 396–403.

[166] Yeomans, J.M. (1992), *Statistical Mechanics of Phase Transitions*, Oxford University Press.

[167] Zhu, Y., Fu, X., Graham, B., Bettati, R., and Zhao, W. (2004), "On flow correlation attacks and countermeasures in mix networks," *Proc. PET'04*, 207–225.

[168] http://news.bbc.co.uk/1/hi/scotland/edinburgh_and_east/7431969.stm (2.3.2009). *Mobile Network Failure Reported.*

[169] http://news.bbc.co.uk/1/hi/technology/3305431.stm (2.3.2009), *Mobile Network Users Hit by Fault.*

[170] http://news.bbc.co.uk/1/hi/technology/4704359.stm (2.3.2009), *Mobile Networks Bear Blast Calls.*

[171] http://wikipedia.org/wiki/4G (2.3.2009), *4G.*

[172] http://www.3gpp.org/article/lte-advanced (2.3.2009), *LTE-Advanced.*